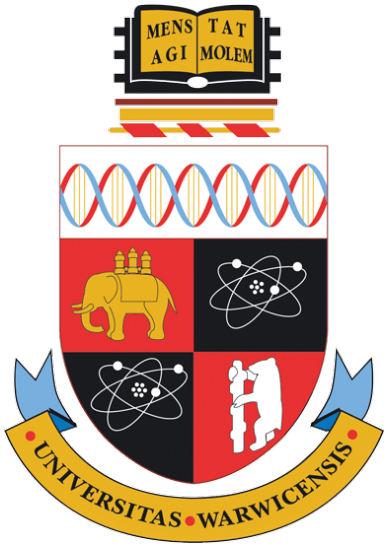


Structured Inference and Sequential Decision-Making with Gaussian Processes

by

Virginia Aglietti

Thesis

Submitted to the University of Warwick

for the degree of

Doctor of Philosophy

Department of Statistics

September 2021

Contents

List of Figures

List of Tables

Acknowledgments

First and foremost, I would like to thank my supervisor, Theodoros Damoulas. Theo's support through my Ph.D. has been crucial for my professional and personal development. Theo has been helpful, supportive, and patient in moments of success and moments of failure and for this, I am deeply grateful. He gave me a hard time (for good reasons) but I certainly would not be submitting this thesis were it not for his guidance. Thank you, Theo, you made me the researcher I am today, and I look forward to continuing our work.

During my Ph.D. journey, I moved around quite a lot and was privileged to have many excellent mentors and collaborators. I would like to thank Javier González for his enthusiasm, support, and careful feedback across our many collaborations. I am grateful for the opportunities he provided at Amazon and Microsoft Research and for teaching me a lot about doing research within the industry. Javier introduced me to the world of causality and this significantly changed my research path for the better. I am grateful to Edwin Bonilla for the patience he had with me during the first phase of my Ph.D. His expertise was crucial in developing the first part of this thesis and I will always remember the time I spent visiting and working with him in Sydney as one of the best periods of my life. Thanks to Sally Cripps, Roman Marchant, and the folks at the Centre for Translational Data Science for supporting my visit to Sydney. I am grateful for the enlightening discussions I had with many other students, scientists and faculty members at the University of Warwick, Oxford, and The Alan Turing Institute. I particularly thank David Firth for his invaluable support and guidance during a difficult phase of my Ph.D. journey.

My Ph.D. was cultivated in the stimulating research environment provided

by the OxWaSP programme. I am grateful to all my Warwick and Oxford friends: it has been an honour to have them in my life. In particular, I'd like to thank my Oxford companions Giulio, Petya, Chris, Emilka and Jack for the long bus trips back from Warwick, the hilarious project presentations, the college parties and the uncountable number of coffee (and beer) breaks. Thanks for making me feel at home right from the start and supporting me during my "master by research" phase. At Warwick, I met so many wonderful people that made these 3 last years pass by in the blink of an eye, despite the amazing city we were living in. Thanks to El Fuego de Warwick (Ale, Elia, Iota, Nayia, and Pier) for the endless laughs. Thanks to Silvia whose arrival in London made me feel a bit younger and light-hearted. Thanks to Vittoria for helping me in getting newfound confidence, for the amazing office experience of this last year, and for giving me the last big push when writing this thesis. An enormous thank goes to Chiara that always believed in me, supported me through very difficult times, and always found time to celebrate my successes. Finally, special thanks go to Nicola and Juan. Thanks to Nicola for helping me start this Ph.D. and to Juan for helping me finish it. Thank you Juan for the endless discussions on the most diverse topics, thanks for the time you were patiently sitting with me to check derivations or debugging the code. I have learned so much from you and despite the latest circumstances, I will always remember our walks in the park and our complaining sessions that made everything look better.

Finally, I'd like to thank my family. Your belief in education, but most importantly your belief in me, is what led me to this point. This journey would not have been possible without your endless love and support. Come sempre, siete voi la mia forza ed il mio orgoglio.

Declarations

This thesis is submitted to the University of Warwick in support of my application for the degree of Doctor of Philosophy. It has been composed by myself and has not been submitted in any previous application for any degree. The work presented was carried out by the author. Parts of this thesis have been previously published by the author as the lead contributor in the following top-ranked venues:

1. **Virginia Aglietti**, Theodoros Damoulas, and Edwin V Bonilla. Efficient Inference in Multi-task Cox Process Models. In *Artificial Intelligence and Statistics* (AISTATS), volume 89, pages 537–546. PMLR, 2019.

2. **Virginia Aglietti**, Edwin V Bonilla, Theodoros Damoulas, and Sally Cripps. Structured Variational Inference in Continuous Cox Process Models. In *Neural Information Processing Systems* (NeurIPS), volume 32, pages 12458–12468. PMLR, 2019.

3. **Virginia Aglietti**, Xiaoyu Lu, Andrei Paleyes, and Javier González. Causal Bayesian Optimization. In *Artificial Intelligence and Statistics* (AISTATS), volume 108, pages 3155–3164. PMLR, 2020b.

4. **Virginia Aglietti**, Theodoros Damoulas, Mauricio Álvarez, and Javier González. Multi-task Causal Learning with Gaussian Processes. In *Neural Information Processing Systems* (NeurIPS), volume 33, pages 6293–6304. PMLR, 2020a.

5. **Virginia Aglietti**, Neil Dhir, Javier González, and Theodoros Damoulas. Dynamic Causal Bayesian Optimization. Submitted to *Neural Information Processing Systems* (NeurIPS) in May 2021.

Additional research that does not form part of this thesis was performed in collaboration as a joint first (denoted by a star) or second author:

1. Nicola Branchini, **Virginia Aglietti***, Neil Dhir and Theodoros Damoulas. Causal Entropy Optimization: Joint Optimization and Structure Learning. In preparation for *Artificial Intelligence and Statistics* (AISTATS) 2022.

2. Shanaka Perera, **Virginia Aglietti** and Theodoros Damoulas. A variational Bayesian spatial interaction model for estimating revenue and demand at business facilities. Passed the editorial review in the *Journal of the Royal Statistical Society: Series A*. arXiv preprint arXiv:2108.02594.

3. Shanaka Perera, **Virginia Aglietti** and Theodoros Damoulas. Optimal store planning via a Bayesian Spatial Interaction Model. In preparation for the *Journal of the Royal Statistical Society: Series A*.

Abstract

Sequential decision-making is a central ability of intelligent agents interacting with an environment, including humans, animals, and animats. When those agents operate in complex systems, they need to be endowed with automatic decision-making frameworks quantifying the system uncertainty and the utility of different actions while allowing them to *sequentially* update their beliefs about the environment. When agents also aim at *manipulating* a system, they need to understand the *data-generating mechanism*. This requires accounting for causality which allows evaluating counterfactual scenarios while increasing interpretability and generalizability of an algorithm. Sequential causal decision-making algorithms require an accurate surrogate model for the causal system and an acquisition function that based on its properties allows selecting actions. In this thesis, I tackle both components through the Bayesian framework which enables probabilistic reasoning while handling uncertainty in a principled manner. I consider Gaussian process (GP) models for both inference and causal decision-making as they provide a flexible framework capable of capturing a variety of data distributions.

I first focus on developing scalable GP models incorporating structure in the likelihood and accounting for complex dependencies in the posteriors. These are indeed crucial properties of surrogate models used within decision-making algorithms. Particularly, I investigate models for point data as many real-world problems involve events and they present significant computational and methodological challenges. I then study how such models can incorporate causal structure and can be used to select actions based on cause-effect relationships. I focus on multi-task GP models, Bayesian Optimization, and Active Learning and show how they can be generalized to capture causality.

Acronyms

AL Active Learning.

BO Adaptive Bayesian Optimization.

BO Bayesian Optimization.

BVI Black-box Variational Inference.

CBO Causal Bayesian Optimization.

CGO Causal Global Optimization.

CDF Cumulative Distribution Function.

CI Credible Intervals.

CP Cox Process.

CPU Central Processing Unit.

DAG Directed Acyclic Graph.

DBN Dynamic Bayesian Network.

DCBO Dynamic Causal Bayesian Optimization.

DCGO Dynamic Causal Global Optimization.

EC Empirical Coverage.

EI Expected Improvement.

ELBO Evidence Lower Bound.

ELL Expected Log Likelihood.

ES Exploration Set.

GP Gaussian Process.

i.i.d. Independent Identically distributed.

ICM Intrinsic Coregionalisation Model.

KL Kullback-Leibler.

LCM Linear Coregionalisation Model.

LGCP Log Gaussian Cox Process.

MAB Multi-armed bandits.

MCMC Markov Chain Monte Carlo.

MCPM Multi-task Cox Process Model.

MGF Moment Generating Function.

MIS Minimal Intervention Set.

MLGCP Multivariate Log Gaussian Cox Process.

MTL Multi-Task Learning.

NLPL Negative Log Predicted Likelihood.

PDF Probability Density Function.

PI Probability of Improvement.

PO Potential Outcome.

POMIS Possibly-Optimal Minimal Intervention Set.

PPP Poisson Point Process.

RBF Gaussian Radial Basis Function kernel.

RCT Randomized Controlled Trial.

RL Reinforcement Learning.

RMSE Root Mean Square Error.

SCM Structural Causal Model.

SGCP Sigmoidal Gaussian Cox Process.

SLFM Semiparametric Latent Factor Msodel.

STVB Structured Variational Bayes.

SVI Structured Variational Inference.

VI Variational Inference.

VI-MF Variational Inference - Mean Field.

Notation

The first subgroup of the notation includes the symbols used throughout the complete thesis. The notation included in **Part 1** is used in Chapter 4 and Chapter 5 and is specific to the Poisson point processes models and variational inference scheme developed in those chapters. Finally, the subgroup denoted by **Part 2** includes the notation adopted in Chapter 6, Chapter 7, and Chapter 8 and is specific to sequential causal decision-making algorithms.

$\mathbf{y}$	Generic output vector
$\mathbf{X}$	Generic input matrix
$\mathbf{X}^\star$	Generic matrix of test input
$\mathcal{X}$	Input space
$\mathcal{Y}$	Output space
D	Input dimensionality
P	Output dimensionality\Number of tasks
$\mathcal{D}$	Dataset
N	Number of training data points
$\mathcal{GP}$	Gaussian Process
$\mathcal{N}$	Gaussian distribution
$\mathcal{N}_T$	Truncated Gaussian distribution
$f(\cdot)$	Generic function mapping from $\mathbf{X}$ to $\mathbf{y}$
$\mathbf{f}$	Vector of function values for a set of inputs
ϵ	Likelihood noise
$m(\cdot)$	Prior mean function
$k(\cdot)$	Prior kernel function
ℓ	Kernel length-scale
σ_f^2	Kernel signal variance
σ^2	Variance of the distribution on ϵ.
K_ν	Modified Bessel function
$\mathbf{K}_{xx}$	Kernel function evaluated at $\mathbf{X}$
$\mathbf{K}_{x^\star x}$	Kernel function evaluated at $\mathbf{X}^\star$ and $\mathbf{X}$
$\mathbf{K}_{xx^\star}$	Kernel function evaluated at $\mathbf{X}$ and $\mathbf{X}^\star$
$\mathbf{K}_{x^\star x^\star}$	Kernel function evaluated at $\mathbf{X}^\star$
$\mathbf{K}_{xz}$	Kernel function evaluated at $\mathbf{X}$ and $\mathbf{Z}$

$\mathbf{K}_{zx}$	Kernel function evaluated at $\mathbf{Z}$ and $\mathbf{X}$
$\mathbf{K}_{zz}$	Kernel function evaluated at $\mathbf{Z}$
$m(\cdot \mid \mathcal{D})$	Posterior mean function
$k(\cdot \mid \mathcal{D})$	Posterior kernel function
$\sigma^2(\cdot \mid \mathcal{D})$	Posterior variance
Q	Number of latent functions
$\boldsymbol{\theta}$	Vector of hyperparameters
$\mathbf{Z}$	Inducing inputs
$\mathbf{u}$	Inducing variables
M	Number of inducing points
$q(\cdot)$	Variational distribution
$\boldsymbol{\nu}$	Variational parameters
$\boldsymbol{\nu}_f$	Variational parameters for the latent processes
$\boldsymbol{\nu}_u$	Variational parameters for the inducing processes
$\mathcal{L}_{\text{elbo}(\cdot)}$	Evidence Lower Bound term
$\mathcal{L}_{\text{ell}(\cdot)}$	Expected Log-Likelihood term
$\mathcal{L}_{\text{kl}(\cdot)}$	KL-divergence term
S_p	Training data for the p-th task
N_p	Cardinality of S_p
$f_p(\cdot)$	Generic function mapping for the p-th task
$\mathbf{B}_q$	q-th coregionalization matrix
k_q	q-th kernel matrix
$G_{p,q}$	q-th smoothing kernel for the p-th task
$\mathcal{S}$	Input space for a Poisson point process
ξ	Realisation of a Poisson point process
λ	Intensity function
Λ	Intensity measure
$N(B)$	Number of events in region B
$\alpha(\cdot)$	Acquisition function
$\mathcal{G}$	Causal graph
$\mathbf{U}$	Exogenous variables in a SCM
$\mathbf{V}$	Endogenous variables in a SCM
F	Set of functions in a SCM
$\text{Pa}(\mathbf{X})$	Parents for the variables in $\mathbf{X}$
$\text{do}(X = x)$	Intervention on X at x

Part 1

f_q	q-th latent function
$\mathbf{f}_{\bullet q}$	Latent function values for q-th process
$\mathbf{f}_{n\bullet}$	Latent function values at $\mathbf{x}_n$
$\mathbf{W}$	Matrix of stochastic weights

$\mathbf{W}_{\bullet q}$	Weights for the q-th latent function
$\mathbf{W}_{p\bullet}$	Weights for the p-th task
$\mathbf{K}_{xx}^{q}$	Kernel function for $\mathbf{f}_{\bullet q}$
$\mathbf{K}_{w}^{q}$	Kernel function for $\mathbf{W}_{\bullet q}$
$\boldsymbol{\theta}_{f}^{q}$	Vector of hyperparameters for the q-th latent process
$\boldsymbol{\theta}_{w}^{q}$	Vector of hyperparameters for the q-th mixing process
H_p	p-th task descriptors
D'	Dimensionality for H_p
y_{np}	Number of events at $\mathbf{x}_n$ for the p-th task
$\mathbf{x}_{n_p}$	Location of the n-th event for the p-th task
$\mathbf{x}_n$	Location of the n-th event
ϕ_p	Task specific offset
λ_p	Intensity function for the p-th task
λ_{np}	Intensity function for the p-th task at $\mathbf{x}_n$
$\lambda^{\star}$	Upper bound of the intensity function
τ	Observation domain for a Poisson point process
$\mathbf{Z}_q$	Inducing inputs for the q-th latent function
$\mathbf{u}_{\bullet q}$	Inducing variables for the q-th latent function
$\mathbf{K}_{zz}^{q}$	q-th kernel function evaluated at $\mathbf{Z}_q$
$\mathbf{K}_{xz}^{q}$	q-th kernel function evaluated at $\mathbf{X}$ and $\mathbf{Z}_q$
$\mathbf{K}_{zx}^{q}$	q-th kernel function evaluated at $\mathbf{Z}_q$ and $\mathbf{X}$
$\boldsymbol{\nu}_w$	Variational parameters for the mixing processes
$\mathbf{m}_q$	Mean of the variational distribution on $\mathbf{u}_{\bullet q}$
$\mathbf{S}_q$	Covariance of the variational distribution on $\mathbf{u}_{\bullet q}$
ω_q	Mean of the variational distribution on $\mathbf{W}_{\bullet q}$
$\boldsymbol{\Omega}_q$	Covariance of the variational distribution on $\mathbf{W}_{\bullet q}$
$\mathcal{L}_{\text{ent}(\cdot)}$	Entropy term
$\mathcal{L}_{\text{cross}(\cdot)}$	Negative Cross-Entropy term
$N^{\star}$	Predicted counts
$\sigma(\cdot)$	Logistic sigmoid function
$\mathcal{L}(\cdot)$	Likelihood function
K	Number of latent thinned events
$\mathbf{y}_k$	Location of the k-th thinned event
$\mathbf{m}$	Mean of the variational distribution on $\mathbf{u}$
$\mathbf{S}$	Covariance of the variational distribution on $\mathbf{u}$
μ_s	Mean of the s-th component of $q(\{\mathbf{y}_k\}_{k=1}^{K}\|K)$
σ_s^2	Variance of the s-th component of $q(\{\mathbf{y}_k\}_{k=1}^{K}\|K)$
π_s	Weight of the s-th component of $q(\{\mathbf{y}_k\}_{k=1}^{K}\|K)$
α	Shape of the variational distribution on $\lambda^{\star}$
β	Rate of the variational distribution on $\lambda^{\star}$
η	Parameter of the variational distribution on K

$\gamma(\cdot)$	Digamma function
l_2	Euclidean norm
ℓ_{test}	Test log likelihood

Part 2

$\mathbf{C}$	Non manipulative variables in a SCM
Y	Target variable in a SCM
$\mathbf{X}$	Manipulative variables in a SCM
$\mathcal{D}^O$	Observational dataset
$\mathcal{D}^I$	Interventional dataset
$\mathbf{X}_s^\star$	Optimal intervention set
$\mathbf{x}_s^\star$	Optimal intervention level
$\mathbf{X}_s$	s-th intervention set
$\mathbf{x}_s$	Generic intervention level for the s-th intervention set
$D(\mathbf{X})$	Interventional domain of variable set $\mathbf{X}$
$Co(\cdot,\cdot)$	Cost function
H	Number of interventions (budget) the agent can perform
$\mathbb{M}_{\mathcal{G},Y}^{\mathbf{C}}$	MISs sets
$\mathbb{P}_{\mathcal{G},Y}^{\mathbf{C}}$	POMISs sets
$\mathbb{B}_{\mathcal{G},Y}^{\mathbf{C}}$	Intervention set explored by Bayesian Optimization
f_s	Objective function corresponding to the intervention set $\mathbf{X}_s$
$m_s(\cdot)$	Prior mean function of GP on f_s
$k_s(\cdot)$	Prior kernel function of GP on f_s
$\mathcal{C}(\cdot)$	Convex hull
N_s^I	Number of interventional data points for $\mathbf{X}_s$
$\mathbf{T}$	Full set of intervention functions in a DAG
$t_s(\cdot)$	Intervention function in $\mathbf{T}$ corresponding to $\mathbf{X}_s$
L_s	Causal operator for $t_s(\cdot)$
$\mathbf{L}$	Set of variables directly confounded with Y in a DAG
$\mathbf{L}^N$	Variables in $\mathbf{L}$ that are not colliders
$\mathbf{I}$	Set of parents of the target node
$\mathbf{b}$	Value for the variables in the base set
$m_{\mathbf{T}}(\cdot)$	Prior mean function of the joint distribution on $\mathbf{T}$
$K_{\mathbf{T}}(\cdot)$	Prior covariance function of the joint distribution on $\mathbf{T}$
$m_{\mathbf{T}\|\mathcal{D}^I}(\cdot)$	Posterior mean function of the joint distribution on $\mathbf{T}$
$K_{\mathbf{T}\|\mathcal{D}^I}(\cdot)$	Posterior covariance function of the joint distribution on $\mathbf{T}$
$H(\cdot)$	Entropy function
$A^\star$	Optimal set of observations collected by active learning
$\mathcal{G}_{0:T}$	Causal graph including all variables from 0 to T
M_t	Sub SCM at time t
$\mathcal{G}_t$	Sub graph associated to M_t

Y_t	Target variable at time t
$\mathbf{X}_t$	Manipulative variables in a SCM at time t
$\mathbf{C}_t$	Non manipulative variables in a SCM at time t
$\mathbf{U}_t$	Exogenous variables in a SCM at time t
$\mathbf{V}_t$	Endogenous variables in a SCM at time t
$\mathbf{X}^\star_{s,t}$	Optimal intervention set at time t
$\mathbf{x}^\star_{s,t}$	Optimal intervention level at time t
$\mathbf{X}_{s,t}$	s-th intervention set at time t
$\mathbf{x}_{s,t}$	Generic intervention level for the s-th intervention set at time t
$I_{0:t-1}$	Decision Interventions at time step 0 to $t-1$
$I^V_{0:t-1}$	Intervention sets at time step 0 to $t-1$
$I^L_{0:t-1}$	Intervention levels at time step 0 to $t-1$
$f_{s,t}$	Objective function corresponding to the intervention set $\mathbf{X}_{s,t}$
$\mathbb{M}_t$	MISs sets at time t
$m_{s,t}(\cdot)$	Prior mean function of GP on $f_{s,t}$
$k_{s,t}(\cdot)$	Prior kernel function of GP on $f_{s,t}$
$\mathcal{D}^I_{s,t}$	Interventional dataset for the intervention set $\mathbf{X}_{s,t}$
$\mathbf{X}^I$	Vector of interventional values in $\mathcal{D}^I_{s,t}$
$\mathbf{Y}^I_{s,t}$	Vector of target values corresponding to $\mathbf{X}^I$
$m_{s,t}(\cdot \mid \mathcal{D})$	Posterior mean function of GP on $f_{s,t}$
$k_{s,t}(\cdot \mid \mathcal{D})$	Posterior kernel function of GP on $f_{s,t}$
G_t	Modified gap metric at time t

Chapter 1

Introduction

1.1 Motivation

Sequential decision-making is a central ability of intelligent agents interacting with an environment, including humans, animals, and animats. Indeed, many important real-world problems such as systems design, medical treatment selection, or gene targeting involve deciding under uncertainty while *sequentially* updating the beliefs about the environment. For instance, doctors recommend a sequence of treatments weighting up the benefits, burdens, and risks of the various options while accounting for uncertainty in the treatment effectiveness and the patient's reaction to it. When designing a system, engineers need to account not only for the business context they operate in but also the environmental context. This includes e.g. the state of the economy or the consumers' preferences which are uncertain and changing over time. Finally, when studying human genes, researchers employ biological systems that approximate their functions as closely as possible and decide which genes to remove or alter in order to study the effects of specific variants. Therefore, in a wide variety of domains, agents need to be endowed with automatic decision-making frameworks quantifying the system uncertainty and the utility of different actions. When agents also aim at *manipulating* a system, they need to understand the underlying *data-generating mechanism* which requires accounting for *causality* and incorporating it within the decision process. This thesis tackles this issue by developing probabilistic methods for structured inference and sequential selection of actions in a causal system.

Causality has been discussed by philosophers since the time of Hume [Hume, 2003] and Kant [Kant and Guyer, 1996] and has been studied by researchers in a variety of different fields e.g. social science [Hedström and Ylikoski, 2010], psychology [Michotte, 2017] or physics [Frisch, 2014]. Indeed, causal reasoning has been recognised as a distinctive feature of human beings [Buchsbaum et al.,

2012; Penn and Povinelli, 2007; Sloman and Lagnado, 2015] and, as recently discussed by Judea Pearl [Pearl and Mackenzie, 2018], from "the discovery that certain things cause other things [...] came organized societies, then towns and cities, and eventually the science and technology-based civilization we enjoy today. All because we asked a simple question: Why?". The first formal mathematical treatment of causal inference in observational studies appeared in the 1930s in the field of econometrics and can be traced back to Wright [Wright, 1934] and Haavelmo [Haavelmo, 1943]. Over the last thirty years, research on causal inference in statistics and computer science has thrived. Two main frameworks, namely the Potential Outcome (PO) framework and the work on Directed Acyclic Graphs (DAGs), have been adopted to investigate causality. The PO framework [Rubin, 2005] is associated with the work by Donald Rubin and builds on the research of Ronald Fisher [Fisher, 1936] and Jerzey Neyman [Splawa-Neyman et al., 1990] on randomized controlled trials. The approach based on Structural Causal Models (SCM) and DAGs [Pearl, 1995] is instead associated with the work by Judea Pearl and his collaborators.

While Pearl [2009a] has shown how every assumption in a SCM framework can be translated to its counterpart in the PO framework, we can identify some major differences between the two approaches that led us to adopt Pearl's framework. First of all, the object of analysis in the PO framework is the unit-based response variable that is the counterfactual quantity representing the value that an outcome variable would obtain in a specific experimental unit had the treatment been set to a certain value. Given the focus on the unit, in the PO framework the causal effects of the variables other than the treatment and the special variables e.g. instrumental variable are not defined. While this limits the prior knowledge required by the PO framework, in this thesis we are interested in evaluating and comparing various treatment variables. In addition, the methods developed within the PO literature have mainly focused on estimating the average effects of binary treatments. This thesis focuses on settings where the variables are mainly continuous thus we need a framework that allows us to model and compare not only the causal effects across different variables but also across a range of interventional levels. Furthermore, the SCM framework uses DAGs to give clear graphical representations of the assumptions behind a causal model. Using the graphical representation is particularly useful when studying the correlation structure among causal effects, as we shall see in Chapter 7, but also when comparing different populations which might be associated with partially different graph structures. Finally, the methodologies developed within the DAG literature allow answering causal queries in complex models characterized by a large number of variables and where scalability might represent an issue[1].

[1]See Imbens [2020] and Chapter 3 for a comparative discussion of the two approaches.

Given the centrality of causation in many aspects of human reasoning, automated decision-making algorithms should encode and reason in terms of cause-effect relationships, especially when an agent aims at understanding the data generating mechanism and potentially manipulate it. This would allow them to evaluate multiple counterfactual scenarios while increasing the interpretability and generalizability of decision-making algorithms. Various sequential acausal decision-making algorithms, such as Bayesian Optimization (BO) [Shahriari et al., 2015], Multi-Armed Bandits (MAB) [Slivkin, 2019] and Reinforcement Learning algorithms (RL) [Kaelbling et al., 1996] have been proposed in the literature with the shared goal of taking decisions based on a belief state, that is a probabilistic representation of our knowledge about the system[2]. All these algorithms are *sequential* that is decisions are selected over a sequence of time steps, in RL and MAB, or a sequence of function evaluations in BO and Active Learning (AL). There is thus a notion of time encoded in sequential decision-making algorithms, which is generally treated explicitly through a time index in RL and MAB algorithm and, apart from some exceptions [Nyikosa et al., 2018], is treated implicitly in BO and AL where we generally speak about trials. Time is also a crucial aspect of causality. Indeed it is sometimes said that causality can only be discussed when taking into account the notion of time. Causes are temporally prior to their effects, which is a concept known as causal asymmetry [Aalen et al., 2012; Peters et al., 2017; Wunsch et al., 2020].

While all these decision-making algorithms are sequential, they deal with different assumptions in terms of action set, reward function, and the way in which decisions affect the state of the environment. In MAB and RL the action set is generally discrete and at every step, the agent takes a decision by maximizing a cumulative reward. In BO the action space is instead continuous and there are infinitely many arms the agent can select. The surrogate model is thus a continuous function defined on the action space, modelling the reward associated with each action and whose correlation structure determines the correlation structure across the rewards. While in BO and MAB the decisions only affect the rewards, in RL every action also influences the state of the environment which evolves over time. BO [Nyikosa et al., 2018] and MAB extensions [Besbes et al., 2014; Wu et al., 2018] have been proposed to deal with dynamic acausal settings. In addition, causal MAB [Bareinboim et al., 2015; Lattimore et al., 2016; Lee and Bareinboim, 2018] and causal RL [Gershman, 2017; Zhang, 2020; Zhang and Bareinboim, 2016] algorithms have been recently proposed with the goal of incorporating causal knowledge into the decision process.

[2]See [Toussaint, 2014] for a discussion on optimal search policies and how they can be seen as trajectories through belief space.

Among sequential decision-making frameworks, Bayesian Optimization or Active Learning have so far lacked a causal counterpart. However, selecting the action(s) optimizing a target variable or allowing to accurately learn a set of functions requires considering causal information. As discussed above for acausal settings, existing causal MAB and causal RL algorithms cannot be straightforwardly applied to solve the problems considered by causal BO or AL. Indeed, as in standard RL, causal RL algorithms focus on finding a policy, that is a mapping between states and actions, with the final goal of minimizing the cumulative regret. Apart from some exceptions (e.g. Lattimore et al. [2016]), cumulative regrets are also considered by causal MAB. More importanlty, the actions space in both causal MAB and causal RL is generally discrete. In those cases agents have to select the intervention variables to manipulate but not the intervention level.

In general, sequential causal decision-making algorithms can be developed by considering two different building blocks:

(i) An accurate *surrogate model* representing the causal system we are interested in and integrating all available sources of information thus quantifying existing uncertainty. We refer here to *epistemic* uncertainty that is the uncertainty representing our current knowledge of the environment i.e. our belief state.

(ii) An *acquisition function* that, based on the properties of the surrogate model e.g. its uncertainty, balances the use of our resources and enables the selection of actions.

These two building blocks are highly interconnected. An accurate surrogate model enables the acquisition function to correctly quantify the benefit of selecting each specific action thus driving the exploration to promising regions. Vice versa, the actions selected sequentially via the acquisition function lead to new observations that are used to update the surrogate model decreasing its uncertainty and leading to more accurate predictions.

This thesis addresses the problem of *developing an integrated framework for accurate estimation and selection of actions in a causal system.* I tackle this problem through the Bayesian framework that allows probabilistic reasoning while handling uncertainty in a principled manner. More specifically, I consider Gaussian process (GP) models for both inference and causal decision-making [3].

[3]Note that resorting to Bayesian approaches for causal inference presents its flaws. For instance, Hahn et al. [2018] shows how using shrinkage priors for linear regression coefficients in the context of causal effect estimation might lead to "regularization-induced confounding". re

GPs are a Bayesian non-parametric approach[4] well suited for building complex probabilistic models as they are capable of expressing a wide range of modelling assumptions. Aside from expressivity, GPs allow quantifying uncertainty which is vital for both predictions and decision-making. Finally, GP models can be used to incorporate complex prior beliefs about a system which might come in different forms e.g. a causal graph or a multi-task structure. As we will see later, structured prior distributions enable the integration of different data sources and are crucial when dealing with complex causal systems where only a few interventions can be performed. While adopting a Bayesian non-parametric framework is advantageous in terms of model flexibility, uncertainty estimation, and prior knowledge incorporation, it often involves computations that are costly or impossible to solve exactly. Incorporating information from observed data, that is computing the posterior distribution, is often not possible in closed-form. This led to the development of many approximation methods like Markov Chain Monte Carlo algorithms [Brooks et al., 2011] or variational methods [Beal, 2003] which will be the focus of this thesis.

This thesis investigates two specific research questions: *how to develop scalable probabilistic models for point data that incorporate structure in the likelihood and posterior distribution and could be thus used as surrogate models*; *why and how to incorporate causality into sequential decision-making algorithms so as to enable the selection of actions.*

I first focus on how to construct flexible and meaningful representations of the system we are analysing. I develop accurate and scalable GP models that incorporate complex dependencies in the likelihood and the posterior distributions. *Structure* in this context refers to two different aspects. On the one hand, capturing complex phenomena often requires structured inference as the posterior distributions for different model variables are highly dependent and standard mean-field factorisations would not suffice. On the other hand, structure might need to be incorporated in the likelihood function to account for cross-correlation across different processes we wish to jointly model. Indeed, multi-task models lead to improved prediction accuracy, especially in the context of missing data, and better uncertainty quantification. These are important features for both inference, as we shall see in Chapter 4, and decision-making, as discussed in Chapter 7. Particularly, I investigate models for point data as many real-life observations in fields as diverse as epidemiology, social sciences, or geology are represented by events. In addition, similarly to what happens for binary data or outlying observations, models for point data present significant methodological and computational challenges. Indeed, when non-Gaussian likelihoods are adopted in the context of GP models, posterior distribution can

[4]See Wasserman [2006] for an introduction to non-parametric statistics.

not be computed in closed form and approximation schemes need to be used, potentially slowing down inference and compromising accuracy.

I then study decision-making algorithms and investigate how, based on complex surrogate models such as those developed in the first part of the thesis, an agent can act in a causal system with the final goal of maximizing a reward. More specifically, this thesis generalises Bayesian Optimization, Active Learning, and multi-task GP models to deal with causal information. We show why sequential decision-making algorithms should be equipped with causal knowledge and how one can develop such frameworks integrating different types of data. In these settings, we incorporate *structure* in the surrogate models by encoding causal information which might come in the form of a causal graph or as observed interventional data.

1.2 Thesis Structure

The work in this thesis is related to two main areas of research: Gaussian process models and causal decision-making algorithms. I thus provide two background chapters and divide the notation into subgroups to aid the reader. In Chapter 2 I introduce GP regression, GP modulated Poisson point processes and the connected scalable inference schemes. Furthermore, I discuss the advantages and disadvantages of using such probabilistic models. Chapter 3 provides an introduction on causality and the sequential decision-making framework we mainly focus on in this thesis that is Bayesian Optimization (BO).

The first part of the thesis includes Chapter 4 and Chapter 5 and focuses on developing models incorporating correlation structure in the likelihood function (Chapter 4) or in the posterior approximation (Chapter 5). These are indeed important features of probabilistic models used within decision-making algorithms. By properly quantifying uncertainty, these models allow the acquisition function constructed based on their properties to efficiently explore different actions, correctly balancing exploration and exploitation. In addition, the efficient inference schemes we develop in these chapters enable fast updating of the posterior distributions and can be thus used when actions are selected sequentially and posterior updates are not available in closed form.

In the second part of the thesis, I study how probabilistic models, such as those developed in the first two chapters, can be combined with an acquisition function to obtain sequential decision-making algorithms. In particular, I analyse how a causation structure rather than a correlation structure can be incorporated in GP surrogate models allowing to select actions based on cause-effect relationships. Chapter 6 and Chapter 8 generalise Bayesian Optimization

to incorporate causal information both in static and dynamic settings. Chapter 7 links correlation and causation through a causal multi-task formulation that, as done in Chapter 4, captures the correlation structure across a set of functions but where each function represents a causal quantity. In turn, this causal multi-task formulation leads to complex structured posterior distributions, such as those seen in Chapter 5, thereby significantly improving the performance of decision-making algorithms such as Bayesian Optimization and Active Learning when used as a surrogate. Chapter 9 concludes the thesis, summarising the primary contributions and outlining open research questions and challenges.

1.3 Contributions

I outline the high-level contributions in each core chapter and how they relate to the central theme of the thesis. More detailed statements on the contributions can be found in each chapter.

Chapter 4. Efficient Inference in Multi-task Cox Process Models
Chapter 4 considers the problem of accurately modelling point data and generalizes the log Gaussian Cox process (LGCP) framework to deal with the existence of multiple correlated processes we wish to *jointly* model. In this chapter, we develop a framework in which observations are treated as realizations of multiple LGCPs, whose log intensities are given by linear combinations of latent functions drawn from GP priors. The combination coefficients are also drawn from GP and can incorporate additional dependencies. To ensure scalability, we derive closed-form expressions for the moments of the intensity functions and propose an efficient variational inference algorithm that is orders of magnitude faster than competing deterministic and stochastic approximations. We show how the proposed approach outperforms the benchmarks in multiple problems, offering the current state of the art in modelling multivariate point processes.

The work in this chapter appeared as: Aglietti, V., Damoulas, T. & Bonilla, E.V. Efficient inference in multi-task cox process models. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics.* PMLR, 2019. The acceptance rate for this conference was 32.4% (360 accepted papers out of 1111 submissions).

Chapter 5. Structured Variational Inference in Continuous Cox Process Models
Still focusing on point data, this chapter proposes a scalable framework for accurate inference in a single-task inhomogeneous Poisson process model. Differently, from Chapter 4, we model the data with a *continuous* sigmoidal Cox

process where the intensity function is given by a GP prior transformed with a scaled logistic sigmoid function. We present a tractable representation of the likelihood function through augmentation with a superposition of Poisson processes. This view enables a structured variational approximation scheme capturing dependencies across variables in the model. The proposed framework avoids discretization of the domain, does not require accurate numerical integration over the input space, and is not limited to GPs with squared exponential kernels. We demonstrate the benefits of this approach on different synthetic and real-world settings with increasing input dimensionality.

The approach presented in this chapter was published as: Aglietti, V., Bonilla, E. V., Damoulas, T., & Cripps, S. Structured variational inference in continuous cox process models. In *Proceedings of the 33rd International Conference on Neural Information Processing System.* 2019. The acceptance rate for this conference was 21.1% (1428 accepted papers out of 6743 submissions).

Chapter 6. Causal Bayesian Optimization
This chapter builds on non-parametric methods and advances the thesis into causal decision-making. I study the problem of optimizing a variable that is part of a causal model in which a sequence of interventions can be performed. We develop an approach which we call Causal Bayesian Optimization (CBO) and generalizes Bayesian optimization to scenarios where causal information is available. We combine ideas from causal inference, uncertainty quantification, and sequential decision making to improve the ability to reason about optimal decision-making strategies decreasing the optimization cost while avoiding suboptimal solutions. We develop a GP surrogate model incorporating different types of data and knowledge of the causal graph via a structured prior. We discuss how CBO automatically balances two trade-offs: the classical exploration-exploitation and the new observation-intervention, which emerges when combining real interventional data with the estimated intervention effects computed via *do*-calculus. We demonstrate the practical benefits of this method in a synthetic setting and in two real-world applications.

This work has been introduced in: Aglietti, V., Xiaoyu, L., Paleyes, A. & González, J. Causal Bayesian Optimization. In *Proceedings of the 23nd International Conference on Artificial Intelligence and Statistics.* PMLR, 2020. The acceptance rate for this conference was 30.2% (423 accepted papers out of 1400 submissions).

Chapter 7. Multi-task Causal Learning with Gaussian Processes
Causal information is useful not only when optimizing a target variable but,

more generally, when the goal is to learn a set of functions defined on a causal graph. In this chapter, we tackle this problem by first studying the correlation structure of a set of intervention functions. Based on this, we then propose the first multi-task causal GP model, which we call DAG-GP. Constructing a structured prior based on the causal graph topology, DAG-GP allows for information sharing *across* continuous interventions and *across* experiments on different variables. DAG-GP accommodates different assumptions in terms of data availability and captures the correlation between functions lying in input spaces of different dimensionality via a well-defined integral operator. We give theoretical results detailing *when* and *how* the DAG-GP model can be formulated depending on the causal graph. We test both the quality of its predictions and its calibrated uncertainties. We show how, compared to single-task models, DAG-GP achieves the best fitting performance in a variety of real and synthetic settings. In addition, it helps to select optimal interventions faster than competing approaches when used within sequential decision-making frameworks, like active learning and the CBO framework introduced in Chapter 6.

The work in this chapter appeared as: Aglietti, V., Damoulas, T., Álvarez, M. & González. Multi-task Causal Learning with Gaussian Processes. In *Proceedings of the 34rd International Conference on Neural Information Processing System*. PMLR, 2020. The acceptance rate for this conference was 20.1% (1900 accepted papers out of 9454 submissions).

Chapter 8. Dynamic Causal Bayesian Optimization

While in Chapter 6 we deal with static settings, in various real-world applications the goal is to identify a *sequence* of optimal interventions in a causal dynamical system where both the target variable of interest and the inputs evolve over time. In this chapter we generalise CBO to deal with these problems and propose Dynamic Causal Bayesian Optimization (DCBO), a framework bringing together ideas from causal inference, GP emulation, and dynamic Bayesian networks. DCBO is useful in scenarios where causal effects in a graph are changing over time and the agents need to track the optimum over time. Using a structured GP surrogate model and the acquisition function proposed in Chapter 6, DCBO identifies a local optimal intervention at every time step by integrating both observational and past interventional data collected from the system. We give theoretical results detailing how one can transfer interventional information across time steps and define a dynamic causal GP model which can be used to quantify uncertainty and find optimal interventions in practice. We demonstrate how DCBO identifies optimal interventions faster than competing approaches in multiple settings and applications.

The DCBO framework was submitted to the 34rd International Conference on Neural Information Processing System as: Aglietti, V., Dhir, N., González, J., & Damoulas, T. Dynamic Causal Bayesian Optimization.

Chapter 2

Background Part A: Inference

In this chapter, we introduce Gaussian process (GP) models and Poisson point processes (PPP). In Section 2.1 we discuss Gaussian processes, the sparse GP approximations, which are often used due to computational reasons, and introduce variational inference as an approximate inference technique for sparse GPs. In Section 2.3 we introduce Poisson point processes focusing in particular on Cox processes. We then discuss how GPs can be used to model the intensity function of PPP to obtain a GP modulated PPP model. We conclude with a discussion of the main advantages and disadvantages of GPs based models.

2.1 Gaussian Processes

Many problems in machine learning can be reduced to learning a mapping from a space of inputs $\mathcal{X}$ to a space of outputs $\mathcal{Y}$. In classification, $\mathcal{Y}$ is a set of discrete values, while in regression it is continuous. When adopting a Bayesian framework, the mapping is represented by a random variable, with our current state of knowledge represented as its distribution. Inference proceeds by first defining a prior distribution consistent with our beliefs and then updating it with observed data using Bayes' rule. Gaussian processes (GPs) are a class of distributions over functions that can be used for representing prior and posterior beliefs over mappings [Williams and Rasmussen, 2006]. In the next few sections, we will review how to manipulate GPs priors and posteriors. An in-depth discussion can be found in Williams and Rasmussen [2006].

A GP is a generalization of the multivariate Gaussian distribution to an infinite number of dimensions or random variables. More formally:

Definition 2.1. (Gaussian process) A Gaussian process is a collection of variables, any finite number of which have a joint Gaussian distribution.

A GP is thus a stochastic process that is fully specified by its mean and covariance functions, $m : \mathcal{X} \to \mathbb{R}$ and $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ respectively, which can

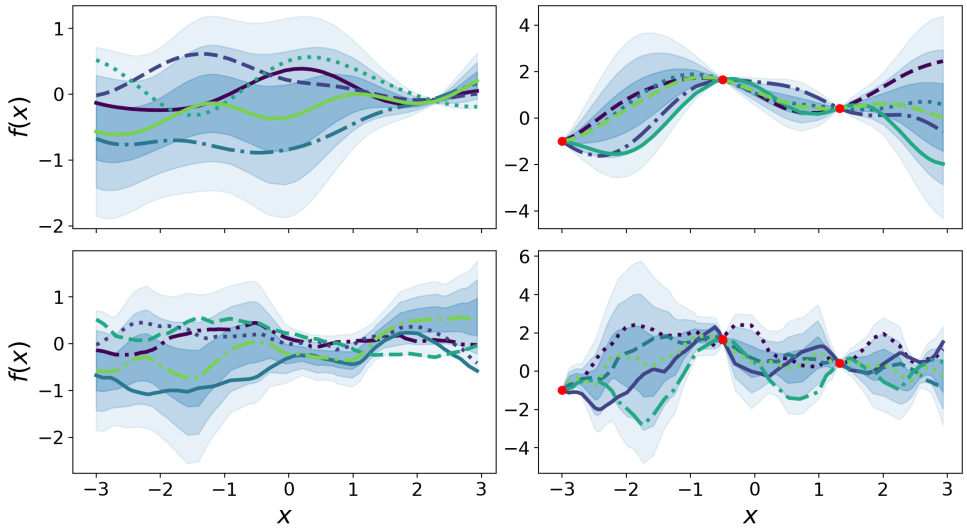


Figure 2.1: A visual representation of a GP model in a one-dimensional input space. Shaded areas give different levels of standard deviations of the predictive distribution at each input location. Red dots represent observed data points. *Left plots*: Samples from the GP prior distribution with $m(\mathbf{x}) = 0$ and an RBF kernel (top) or a Matérn 3/2 kernel (bottom). *Right plots*: Samples from the GP posterior distribution with RBF kernel (top) or a Matérn 3/2 kernel (bottom).

be evaluated at any position of an infinite input domain, e.g. $\mathcal{X} = \mathbb{R}$. We write:

$$\begin{aligned} f &\sim \mathcal{GP}(\mu(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')) \\ m(\mathbf{x}) &= \mathbb{E}[f(\mathbf{x})] \\ k(\mathbf{x}, \mathbf{x}') &= \mathbb{E}\big[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))\big]. \end{aligned}$$

Different assumptions in terms of $m(\mathbf{x})$ and $k(\mathbf{x}, \mathbf{x}')$ can be made depending on the setting considered and the assumptions we want to encode in terms of differentiability and smoothness of the function we are modelling. In this thesis we assume $m(\mathbf{x}) = 0$ unless otherwise stated which is a typical choice in many modelling scenarios. This is not a limitation as the mean of the posterior distribution is not confined to be zero. In addition, when assuming a deterministic prior mean function different from zero, one can apply the usual zero mean GP model to the difference between the observations and the fixed mean function. Yet there are several reasons why one might wish to explicitly model a mean function, including model interpretability or to incorporate additional prior knowledge. We will see some examples in the second part of this thesis when discussing causal decision-making algorithms. When $m(\mathbf{x}) = 0$, the GP is described solely by the covariance function for which different parametric form $k(\mathbf{x}, \mathbf{x}') = k_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{x}')$, where $\boldsymbol{\theta}$ denotes the kernel *hyperparameters*, can be assumed. A frequent choice for this function is represented by the Gaussian Radial Basis Function kernel (RBF) defined as

$k_{\text{RBF}}(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp(-\frac{r^2}{2\ell^2})$ with $r = ||\mathbf{x} - \mathbf{x}'||_2$ being the Euclidean distance and $\boldsymbol{\theta} = (\sigma_f^2, \ell)$ where ℓ is called characteristic length-scale and σ_f^2 is the signal variance. This covariance function is infinitely differentiable, which means that a GP with an RBF kernel has mean square derivatives of all orders, and is thus very smooth. Alternatively, one can choose a Matérn kernel defined as $k_{\text{MATÉRN}}(\mathbf{x}, \mathbf{x}') = \frac{2^{\nu-1}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}r}{\ell} \right)^{\nu} K_\nu \left(\frac{\sqrt{2\nu}r}{\ell} \right)$ with positive hyper-parameters $\theta = (\nu, \ell)$ and where K_ν is a modified Bessel function. The hyperparameters ν controls the smoothness of the resulting function. The smaller ν, the less smooth the function is. As $\nu \to \infty$, the Matérn kernel becomes equivalent to the RBF kernel. When $\nu = 1/2$, the Matérn kernel becomes identical to the absolute exponential kernel. Important intermediate values are $\nu = 3/2$ for which the function is once differentiable and $\nu = 5/2$ for which the functions is twice differentiable. For an overview of useful kernels in machine learning and GP modelling see Williams and Rasmussen [2006] and Schölkopf et al. [2002]. As an illustration, consider a GP with mean function $m(\mathbf{x}) = 0$ and an RBF kernel with $(\sigma_f^2, \ell) = (1, 1)$. We can draw different realizations from this GP *prior distribution* on a grid of points $X := \{x_1, \ldots, x_n\}$ (Fig. 2.1, left column, top plot). The same can be repeated for a Matérn kernel with $(\nu, \ell) = (3/2, 1)$ (Fig. 2.1, left column, bottom plot).

In a Bayesian framework, we are usually not primarily interested in drawing random functions from the prior distribution but aim at incorporating the knowledge that a set of training data $\mathcal{D} = (\mathbf{X}, \mathbf{y})$ with $\mathbf{X} \in \mathbb{R}^{N \times D}$ and $\mathbf{y} \in \mathbb{R}^N$ provides about the function. In real-world applications, it is typical to assume that we can only observe noisy measurements $\mathbf{y}$ of the true function values $\mathbf{f} = f(\mathbf{X})$, where each output value is generated according to $y = f(x) + \epsilon$ with $\epsilon \sim \mathcal{N}(0, \sigma^2)$. This induces the following *likelihood function*:

$$p(\mathbf{y}|\mathbf{f}, \mathbf{X}) = \mathcal{N}(f(\mathbf{X}), \sigma^2 \mathbf{I}). \tag{2.1}$$

Assuming additive and independently identically distributed (i.i.d.) Gaussian noise ϵ across target values implies a prior covariance for the noisy observations given by $\text{Cov}(y_p, y_q) = k(x_p, x_q) + \sigma^2 \delta_{pq}$ with δ_{pq} representing a Kronecker delta which is one *if and only if* $p = q$. For the vector $\mathbf{y}$ we can write $\text{Cov}(\mathbf{y}) = \mathbf{K}_{xx} + \sigma^2 \mathbf{I}$ where $\mathbf{K}_{xx} = K(\mathbf{X}, \mathbf{X})$ denotes the $N \times N$ matrix obtained evaluating the covariance function at all pairs of training point in $\mathbf{X}$.

Inference in a Bayesian framework requires computing the *posterior distribution* over the latent function $\mathbf{f}$ by Bayes' rule:

$$p(\mathbf{f}|\mathbf{X}, \mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{f}, \mathbf{X}) p(\mathbf{f}|\mathbf{X})}{p(\mathbf{y}|\mathbf{X})}. \tag{2.2}$$

Given the posterior distribution, we can compute the predictive distribution

for the latent function at a test point $\mathbf{x}^\star$ as:

$$p(f^\star|\mathbf{x}^\star, \mathbf{X}, \mathbf{y}) = \int p(f^\star|\mathbf{x}^\star, \mathbf{f}) p(\mathbf{f}|\mathbf{X}, \mathbf{y}) \mathrm{d}\mathbf{f}$$

and subsequently use this distribution to produce a probabilistic prediction for the corresponding output $y^\star$. When considering a Gaussian likelihood function (Eq. (2.1)) together with a GP prior, the posterior and predictive distributions can be computed in closed form as the prior and likelihood form a conjugate pair. Consider a set of test inputs $\mathbf{X}^\star$ for which the corresponding output values $\mathbf{y}^\star$ are not observed. We can write the joint distribution of the observed target values and the function values at the test locations $\mathbf{f}^\star = f(\mathbf{X}^\star)$ under the prior as:

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}^\star \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{K}_{xx} + \sigma^2\mathbf{I} & \mathbf{K}_{xx^\star} \\ \mathbf{K}_{x^\star x} & \mathbf{K}_{x^\star x^\star} \end{bmatrix}\right)$$

with $\mathbf{K}_{xx^\star} = K(\mathbf{X}, \mathbf{X}^\star)$, $\mathbf{K}_{x^\star x} = K(\mathbf{X}^\star, \mathbf{X})$, $\mathbf{K}_{x^\star x^\star} = K(\mathbf{X}^\star, \mathbf{X}^\star)$ denoting the covariance matrices induced by evaluating the covariance function at all pairwise rows of the training inputs $\mathbf{X}$ and test inputs $\mathbf{X}^\star$. Deriving the conditional distribution of $\mathbf{f}^\star|\mathbf{y}, \mathbf{X}, \mathbf{X}^\star$ we obtain the key predictive equation for the Gaussian process regression, see e.g. Williams and Rasmussen [2006]:

$$\mathbf{f}^\star|\mathbf{y}, \mathbf{X}, \mathbf{X}^\star \sim \mathcal{N}\left(\bar{\mathbf{f}}^\star, \mathrm{Cov}(\mathbf{f}^\star)\right) \quad \text{where} \tag{2.3}$$

$$\bar{\mathbf{f}}^\star = \mathbf{K}_{x^\star x}[\mathbf{K}_{xx} + \sigma^2\mathbf{I}]^{-1}\mathbf{y}, \tag{2.4}$$

$$\mathrm{Cov}(\mathbf{f}^\star) = \mathbf{K}_{x^\star x^\star} - \mathbf{K}_{x^\star x}[\mathbf{K}_{xx} + \sigma^2\mathbf{I}]^{-1}\mathbf{K}_{xx^\star}. \tag{2.5}$$

Notice how the variance is the difference between two terms. The first term $\mathbf{K}_{x^\star x^\star}$ is simply the prior covariance computed at the test points. The second term represents the information that the observations give us about the function thus reducing its uncertainty. We can compute the predictive distribution for the test outputs $\mathbf{y}^\star$ by adding $\sigma^2\mathbf{I}$ to the expression for $\mathrm{Cov}(\mathbf{f}^\star)$. Computing $\bar{\mathbf{f}}^\star$ and $\mathrm{Cov}(\mathbf{f}^\star)$ is computationally expensive when the size of $\mathbf{X}^\star$ is large as it involves Cholesky decompositions requiring $\mathcal{O}(N^3)$ time to compute. In the following section, we will introduce the inducing point approximations [Quinonero-Candela and Rasmussen, 2005; Titsias, 2009a] which can be used to scale up the training of a GP model and the computation of the predictive mean. More recently, several approaches have been proposed to speed up the computation of the predictive uncertainties and the sampling from predictive distributions, e.g. Pleiss et al. [2018]; Wilson et al. [2020]. The right column in Fig. 2.1 shows the sample paths drawn from the posterior GP distributions obtained with different priors (Fig. 2.1, left column) and a Gaussian likelihood when a set of training points are observed.

In GP regression the marginal likelihood $p(\mathbf{y}|\mathbf{X})$ of the observed outputs given the inputs is available in closed form and it is given by:

$$p(\mathbf{y}|\mathbf{X}) = \int p(\mathbf{y}|\mathbf{f}, \mathbf{X})p(\mathbf{f}|\mathbf{X})\mathrm{d}\mathbf{f} = \mathcal{N}(\mathbf{0}, \mathbf{K}_{xx} + \sigma^2\mathbf{I}). \tag{2.6}$$

The term marginal likelihood refers to the marginalization over the function values $\mathbf{f}$. Following the standard maximum likelihood procedure one can optimize Eq. (2.6) to find the values of the kernel hyperparameters $\boldsymbol{\theta}$ and σ^2 improving the fit to the data.

In the regression case, the computation of posterior and predictive distributions is straightforward as the relevant integrals can be computed analytically. This is not the case when dealing with non-Gaussian likelihoods. For instance, in binary classification, the likelihood is given by a product of Bernoulli random variables which makes the computation of Eq. (2.2) analytically intractable. The same happens when GPs are used to model count data and Poisson distributions appear in the likelihood functions. In these cases we need to resort to approximate Bayesian methods such as Monte Carlo sampling [Filippone et al., 2013; Havasi et al., 2018; Neal, 1997; Samo and Roberts, 2016] or Variational Bayes [Blei et al., 2017; Fox and Roberts, 2012; Frigola et al., 2014; Tran et al., 2016]. Another well-known problem with GPs is scalability. As mentioned above, computing the posterior distribution requires the inversion of a $N \times N$ covariance matrix $\mathbf{K}_{xx}$ which implies a computational complexity of $\mathcal{O}(N^3)$. However, a variety of sparse approximations have been recently introduced to deal with the memory and computational limitations of GPs. These two aspects, namely sparse approximations and inference in GP models with non-Gaussian likelihoods, will be the focus of the next sections.

2.1.1 Sparse Gaussian Processes

Although GPs have many desirable properties from a modelling point of view, they become computationally intractable to manipulate for even moderately sized datasets. Indeed, posterior inference involves matrix operations costing $\mathcal{O}(N^3)$ where N is the number of observations. To overcome this limitation, several methods have been proposed in the literature, see Liu et al. [2020] for a review. Scalable GPs can be classified into two main categorises. On the one hand we have approaches that approximate the kernel matrix either by considering a subset of the training data [Keerthi and Chu, 2006; Lawrence et al., 2003; Seeger, 2003], or via sparse kernels [Buhmann, 2001; Gneiting, 2002; Melkumyan and Ramos, 2009] or by resorting to sparse approximations [Hensman et al., 2013, 2017; Lázaro-Gredilla and Figueiras-Vidal, 2009; Quinonero-Candela and Rasmussen, 2005; Rossi et al., 2021; Seeger et al., 2003; Smola and Bartlett, 2001; Snelson and Ghahramani, 2005; Titsias, 2009a; Williams and Seeger, 2001;

Wilson and Nickisch, 2015].

On the other hand several approches focus on local approximations and divide the data for subspace learning [Gramacy, 2016; Liu et al., 2018; Rasmussen and Ghahramani, 2002; Samo and Roberts, 2016; Yuksel et al., 2012]. While in terms of scalability, most of the sparse approximations have the same training complexity, they can be further sped up through parallel and distributed computing [Dai et al., 2014; Gal et al., 2014b; Gramacy, 2016]. In addition, the complexity can be futher reduced by exploiting Toeplitz and Kronecker matrix structure for fast matrix-vector multiplications which in turns require regularly spaced inducing points [Cunningham et al., 2008; Saatçi, 2012; Wilson and Nickisch, 2015]. In this thesis, we will focus on the inducing point approximations which are based on the work by Quinonero-Candela and Rasmussen [2005] and Titsias [2009a].

The key idea of the inducing point approximation is to learn the function values at a certain number $M \ll N$ of input locations that are highly informative of what the posterior GP is more globally. The additional auxiliary input-output pairs of variables are denoted by $\mathbf{Z}$ and $\mathbf{u}$ respectively. $\mathbf{Z}$ is a $M \times D$ matrix of *inducing inputs* while $\mathbf{u} \in \mathbb{R}^M$ gives the vector of corresponding function values called *inducing variables*. The original covariance matrix $\mathbf{K}_{xx}$ is replaced with a low-rank approximation that requires the inversion of the $M \times M$ covariance matrix computed in the inducing inputs. The key problem of this approach is the selection of the inducing inputs. These can be constrained to be on a regular grid covering the input space or to be a subset of the training inputs. A common approach is to allow the inducing inputs to lie anywhere in the input domain and determine their location with some form of optimization [Snelson and Ghahramani, 2005]. Here we consider the case where the inducing inputs are related to the outputs with the same GP prior as the training inputs and write an *augmented* joint prior distribution for $\mathbf{f}$ and $\mathbf{u}$ as:

$$\begin{bmatrix}\mathbf{f}\\ \mathbf{u}\end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix}\mathbf{K}_{xx} & \mathbf{K}_{xz}\\ \mathbf{K}_{zx} & \mathbf{K}_{zz}\end{bmatrix}\right)$$

where $\mathbf{K}_{zz}$ is built by evaluating the covariance function on the inducing inputs $\mathbf{Z}$ while $\mathbf{K}_{xz}$ and $\mathbf{K}_{zx}$ represent the cross-covariance matrices between $\mathbf{X}$ and $\mathbf{Z}$. The marginal distribution for the inducing variables can be written as $p(\mathbf{u}|\mathbf{Z}) = \mathcal{N}(\mathbf{0}, \mathbf{K}_{zz})$ which implies the following conditional GP prior:

$$\begin{aligned} p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z}) &= \mathcal{N}\left(\mathbf{K}_{xz}(\mathbf{K}_{zz})^{-1}\mathbf{u}, \mathbf{K}_{xx} - \mathbf{A}\mathbf{K}_{zx}\right) \\ \text{with} \quad \mathbf{A} &= \mathbf{K}_{xz}\mathbf{K}_{zz}^{-1} \end{aligned}$$

where $\mathbf{A}\mathbf{K}_{zx}$ represents the Nyström approximation of the true covariance $\mathbf{K}_{xx}$.

Writing this conditional prior only involves inversion of matrices of dimension $M \times M$ and integrating out $\mathbf{u}$ from the augmented joint prior distribution we can recover the initial prior distribution for $\mathbf{f}$ exactly. While this naive formulation does not bring any computational benefits [Williams and Seeger, 2001], applying the Woodbury formula we obtain the so called Nyström approximation [Williams and Seeger, 2001] which reduces the computational complexity to $\mathcal{O}(NM^2)$. Alternatively, in order to induce sparsity, one can consider an approximation $q(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})$ to the true conditional $p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})$. As noted by Quinonero-Candela and Rasmussen [2005], different sparse approximations correspond to different assumptions on the covariance term of $q(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})$ while maintaining the exact likelihood $p(\mathbf{y}|\mathbf{f})$ and prior $p(\mathbf{u})$. Let's assume $q(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z}) = \mathcal{N}(\mathbf{K}_{xz}(\mathbf{K}_{zz})^{-1}\mathbf{u}, \tilde{Q})$ with $\tilde{Q} \neq \mathbf{K}_{xx} - \mathbf{A}\mathbf{K}_{zx}$ then the marginal prior distribution for $\mathbf{f}$ is given by $q(\mathbf{f}|\mathbf{X}) = \int q(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})p(\mathbf{u}|\mathbf{Z})\mathrm{d}\mathbf{u} = \mathcal{N}(\mathbf{0}, \tilde{Q} + \mathbf{A}\mathbf{K}_{zx})$. Specific forms of $\tilde{Q}$ allow to use the Woodbury matrix identity and the matrix determinant lemma[1] to obtain expressions that depend on the inversion of a matrix of size $M \times M$. Examples are given by the deterministic training conditional (DTC) approximation [Csató and Opper, 2002; Seeger et al., 2003] where $\tilde{Q} = 0$ and the "fully independent training conditional (FITC)" [Snelson and Ghahramani, 2005] method where $\tilde{Q} = \mathbf{A}\mathbf{K}_{zx} - \mathrm{diag}(\mathbf{A}\mathbf{K}_{zx} - \mathbf{K}_{xx})$.

Apart from the methods based on inducing points, recent works [Hartikainen and Särkkä, 2010; Sarkka and Hartikainen, 2012; Särkkä et al., 2013] have further scaled spatio-temporal GPs by reformulating regression problems as Kalman filtering and smoothing problems. Interestingly, the formulation proposed by Hartikainen and Särkkä [2010] is exact for the Matérn class of covariance functions. Exact solutions not constrained to specific kernel classes have also been proposed by Wang et al. [2019] and Cutajar et al. [2016]. The former exploits the Blackbox Matrix-Matrix multiplication [Gardner et al., 2018] while the latter applies different kernel matrix approximations as preconditioners and develops a method that can be applied to any likelihood factorizing over the data points, thus tackling both regression and classification problems.

[1] Consider an invertible matrix $\mathbf{A}$ of size $N \times N$, two matrices U and V of size $N \times M$ and W an invertible matrix of size $M \times M$. We have two following two identities:

- **The Woodbury matrix identity:**

$$(\mathbf{A} + UWV^T)^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}U(W^{-1} + V^T\mathbf{A}^{-1}U^{-1})V^T\mathbf{A}^{-1}.$$

- **The matrix determinant lemma:**

$$|\mathbf{A} + UWV^T| = |W^{-1} + V^T\mathbf{A}^{-1}U||W||\mathbf{A}|.$$

2.1.2 Variational Sparse Gaussian Processes

Focusing an approximation methods based on inducing points, notice how approximating $p(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})$ with $q(\mathbf{f}|\mathbf{u}, \mathbf{X}, \mathbf{Z})$ corresponds to modifying the GP prior [Quinonero-Candela and Rasmussen, 2005; Titsias, 2009a]. The approaches mentioned above turn the inducing inputs into additional kernel hyperparameters and, while this can increase flexibility when we fit the data, it can also lead to overfitting when jointly optimizing over all unknown hyperparameters. To overcome this limitation, Titsias [2009a] proposed a variational method that jointly selects the inducing inputs and the hyperparameters by maximizing a lower bound to the exact marginal likelihood. Rather than modifying the exact GP model, this approach minimizes a distance between the exact GP posterior and a variational approximation thus turning the inducing inputs $\mathbf{Z}$ into variational parameters. Several works have shown how the inducing point approximation avoids the undesirable behaviours observed for the FITC method i.e. overestimation of the marginal likelihood, underestimation of the noise variance parameter, and inability to recover the true posterior in a large number of settings [Bauer et al., 2016; Matthews et al., 2016; Matthews, 2017]. In particular, it correctly identifies good solutions, always improves with additional inducing inputs, and recovers the true posterior when possible. Before detailing the variational inducing point approximation, which will be adopted throughout this thesis for the reasons given above, we introduce variational inference as an approximate inference method that can be used when the posterior GP distribution is not available in closed form.

Variational Bayes Given a GP prior distribution (Eq. (4.1)) and a non-Gaussian likelihood, posterior inference is analytically intractable. Therefore, we need to resort to approximate inference methods to either get samples from it or obtain an approximate form. Variational inference (VI) methods [Jordan et al., 1999] find an approximate posterior distribution by positing a tractable family of distributions and finding the member of the family that is "closest" to the true posterior in terms of their Kullback-Leibler divergence. In GP models, VI seeks to approximate the true posterior $p(\mathbf{f}|\mathcal{D})$ with a variational distribution $q(\mathbf{f}|\boldsymbol{\nu}_\mathbf{f})$ where $\boldsymbol{\nu}_\mathbf{f}$ represents the variational parameters and is obtained by minimizing :

$$KL(q(\mathbf{f}|\boldsymbol{\nu}_\mathbf{f})||p(\mathbf{f}|\mathcal{D})) = \int q(\mathbf{f}|\boldsymbol{\nu}_\mathbf{f}) \log \frac{p(\mathbf{f}|\mathcal{D})}{q(\mathbf{f}|\boldsymbol{\nu}_\mathbf{f})} \mathrm{d}\mathbf{f}.$$

In the following derivations we omit $\boldsymbol{\nu}_\mathbf{f}$ to avoid clutter. It is possible to show that minimizing the Kullback-Leibler (KL) divergence between the approximate posterior and the true posterior is equivalent to maximizing the log-evidence lower bound (ELBO), which is composed of a KL-term and an expected log-

likelihood term (ELL). We can write:

$$\begin{aligned}\log p(\mathbf{y}) &= \log \int p(\mathbf{y}, \mathbf{f}) \mathrm{d}'\mathbf{f} = \log \int p(\mathbf{y}, \mathbf{f}) \frac{q(\mathbf{f})}{q(\mathbf{f})} \mathrm{d}'\mathbf{f} \\ &= \log \mathbb{E}_{q(\mathbf{f})}\left[\frac{p(\mathbf{y}, \mathbf{f})}{q(\mathbf{f})}\right] \geq \mathbb{E}_{q(\mathbf{f})}\left[\log \frac{p(\mathbf{y}, \mathbf{f})}{q(\mathbf{f})}\right]\end{aligned}$$

where $p(\mathbf{y})$ is the marginal likelihood or *evidence* and the last equation is derived by applying Jensen's inequality. The term $\mathcal{L}_{\text{elbo}} = \mathbb{E}_{q(\mathbf{f})}\left[\log \frac{p(\mathbf{y},\mathbf{f})}{q(\mathbf{f})}\right]$ is thus a lower bound on the evidence. In addition we can write:

$$\begin{aligned}KL(q(\mathbf{f})||p(\mathbf{f}|\mathbf{y})) &= \mathbb{E}_{q(\mathbf{f})}\left[\log \frac{q(\mathbf{f})}{p(\mathbf{f}|\mathbf{y})}\right] \\ &= \mathbb{E}_{q(\mathbf{f})}[\log q(\mathbf{f})] - \mathbb{E}_{q(\mathbf{f})}\left[\log \frac{p(\mathbf{f}, \mathbf{y})}{p(\mathbf{y})}\right] \\ &= \mathbb{E}_{q(\mathbf{f})}[\log q(\mathbf{f})] - \mathbb{E}_{q(\mathbf{f})}[\log p(\mathbf{f}, \mathbf{y})] + \mathbb{E}_{q(\mathbf{f})}[\log p(\mathbf{y})] \\ &= \mathbb{E}_{q(\mathbf{f})}[\log p(\mathbf{y})] - \mathbb{E}_{q(\mathbf{f})}\left[\log \frac{p(\mathbf{y}, \mathbf{f})}{q(\mathbf{f})}\right] \\ &= \log p(\mathbf{y}) - \mathcal{L}_{\text{elbo}}. \end{aligned} \tag{2.7}$$

Therefore, minimizing the KL divergence is equivalent to maximizing the ELBO which is a lower bound on the marginal likelihood. This can be done in closed form for the conditionally conjugate exponential family (see Blei et al. [2017] for an example in which the ELBO is computed analytically for a mixture of Gaussians). In this case, coordinate ascent can be used to iteratively update the variational distribution [Ghahramani and Beal, 2000] for each variable until convergence. For generic models and arbitrary variational families, there is no closed-form solution. When the ELBO can be evaluated analytically one can resort to gradient descent methods. However, in complex models, also computing the required expectations in the ELBO expression becomes intractable. In these settings, one needs to resort to model-specific algorithms [Braun and McAuliffe, 2010; Jaakkola and Jordan, 1997] or generic algorithms that require model specific computations [Knowles and Minka, 2011; Paisley et al., 2012].

Alternatively, black-box variational inference (BVI) [Ranganath et al., 2014] has been proposed as a *generic* variational inference algorithm for which only the generative process of the data has to be specified. The main idea of BVI is to represent the gradient of the ELBO as an expectation and to use Monte Carlo techniques to estimate them. One can thus obtain an unbiased gradient estimator by sampling from the variational distribution without having to compute the ELBO analytically. While BVI and stochastic gradient descent make VI applicable to a range of complicated models, they lead to high variance in the gradient estimators which can, in turn, prevent the convergence of the algorithm. Reducing these variances is essential for fast convergence of the VI

scheme and several strategies have been proposed to address this issue, e.g. control variates [Boyle, 1977], re-parametrization tricks [Rezende et al., 2014] and Rao-Blackwellization based approaches [Ranganath et al., 2014].

Mean field approximation A key aspect of variational inference is the chosen family of variational distributions. There exists a trade-off in choosing $q(\mathbf{f}|\boldsymbol{\nu}_\mathbf{f})$ expressive enough to approximate the posterior well, and simple enough to lead to a tractable approximation. When dealing with several latent variables, a common choice is a fully factorized variational distribution, also called mean-field distribution. A mean-field approximation assumes that all latent variables are independent a posteriori, which simplifies derivations. However, this independence assumption might lead to less accurate results as it ignores dependencies. This is especially true when the posterior variables are highly dependent such as in models with hierarchical structure and in point process models, as we will see in Chapter 5. To avoid this issue, structured variational distribution can be used to increase expressiveness at the price of a higher computational cost. Allowing a structured variational distribution to capture dependencies between latent variables is a modelling choice; different dependencies may be more or less relevant and depend on the model under consideration. For example, structured variational inference for Latent Dirichlet Allocation [Hoffman and Blei, 2015] shows that maintaining global structure is vital, while structured variational inference for the Beta Bernoulli Process [Shah et al., 2015] shows that maintaining local structure is more important.

A variational lower bound for the inducing points As mentioned above, the variational inducing point approach [Titsias, 2009a] turns the inducing inputs into variational parameters and, instead of modifying the prior on $\mathbf{f}$, considers a free posterior $q(\mathbf{u}|\boldsymbol{\nu}_\mathbf{u})$ whose variational parameters $\boldsymbol{\nu}_\mathbf{u}$ are then optimized. The approximation is thus made with respect to the true posterior. This will be the approach adopted throughout this thesis. We consider a joint approximate posterior distribution defined as:

$$q(\mathbf{f}, \mathbf{u}) = p(\mathbf{f}|\mathbf{u})q(\mathbf{u}|\boldsymbol{\nu}_\mathbf{u}) \quad \text{with} \quad q(\mathbf{u}|\boldsymbol{\nu}_\mathbf{u}) = \mathcal{N}(\mathbf{m}, \mathbf{S}) \tag{2.8}$$

where $\boldsymbol{\nu}_\mathbf{u} = \{\mathbf{m}, \mathbf{S}, \mathbf{Z}\}$ and $p(\mathbf{f}|\mathbf{u}) = \mathcal{N}(\mathbf{K}_{xz}\mathbf{K}_{zz}^{-1}\mathbf{u}, \mathbf{K}_{xx} - \mathbf{K}_{xz}\mathbf{K}_{zz}^{-1}\mathbf{K}_{zx})$. We can thus rewrite the variational bound in Eq. (2.7) as:

$$
\begin{aligned}
KL(q(\mathbf{f},\mathbf{u}|\boldsymbol{\nu}_\mathbf{u})||p(\mathbf{f},\mathbf{u}|\mathbf{y})) &= KL(q(\mathbf{f}|\mathbf{u})q(\mathbf{u}|\boldsymbol{\nu}_\mathbf{u})||p(\mathbf{f},\mathbf{u}|\mathbf{y})) \\
&= \log p(\mathbf{y}) - \mathbb{E}_{q(\mathbf{f},\mathbf{u}|\boldsymbol{\nu}_\mathbf{u})}\left[\log \frac{p(\mathbf{y},\mathbf{f},\mathbf{u})}{q(\mathbf{f},\mathbf{u}|\boldsymbol{\nu}_\mathbf{u})}\right] \\
&= \log p(\mathbf{y}) - \mathbb{E}_{q(\mathbf{f},\mathbf{u}|\boldsymbol{\nu}_\mathbf{u})}\left[\log \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{u})p(\mathbf{u})}{p(\mathbf{f}|\mathbf{u})q(\mathbf{u}|\boldsymbol{\nu}_\mathbf{u})}\right] \\
&= \log p(\mathbf{y}) - \mathbb{E}_{q(\mathbf{f},\mathbf{u}|\boldsymbol{\nu}_\mathbf{u})}\left[\log \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{u})}{q(\mathbf{u}|\boldsymbol{\nu}_\mathbf{u})}\right].
\end{aligned}
$$

The evidence lower bound is thus written as $\mathcal{L}_{\text{elbo}} = \mathbb{E}_{q(\mathbf{f},\mathbf{u}|\boldsymbol{\nu}_\mathbf{u})}\left[\log \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{u})}{q(\mathbf{u}|\boldsymbol{\nu}_\mathbf{u})}\right]$ and can be decomposed as the sum of two terms, one giving the expected log likelihood and one representing the KL divergence between M dimensional distributions on the inducing inputs $\mathbf{u}$:

$$
\begin{aligned}
\mathcal{L}_{\text{elbo}} &= \mathbb{E}_{q(\mathbf{f},\mathbf{u}|\boldsymbol{\nu}_\mathbf{u})}\left[\log \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{u})}{q(\mathbf{u}|\boldsymbol{\nu}_\mathbf{u})}\right] \\
&= \mathbb{E}_{q(\mathbf{f},\mathbf{u}|\boldsymbol{\nu}_\mathbf{u})}[\log p(\mathbf{y}|\mathbf{f})] - KL(q(\mathbf{u}|\boldsymbol{\nu}_\mathbf{u})||p(\mathbf{u})) \\
&= \underbrace{\mathbb{E}_{q(\mathbf{f})}[\log p(\mathbf{y}|\mathbf{f})]}_{\mathcal{L}_{\text{ell}}(\boldsymbol{\nu}_\mathbf{u})} - \underbrace{KL(q(\mathbf{u}|\boldsymbol{\nu}_\mathbf{u})||p(\mathbf{u}))}_{\mathcal{L}_{\text{kl}}(\boldsymbol{\nu}_\mathbf{u})}. \qquad (2.9)
\end{aligned}
$$

Notice how the N dimensional distributions cancel out in the ELBO therefore reducing the computational complexity to $\mathcal{O}(NM^2)$. In addition, when the likelihood factorizes across the N data points we can write the log likelihood term as $\mathbb{E}_{q(\mathbf{f},\mathbf{u}|\boldsymbol{\nu}_\mathbf{u})}[\log p(\mathbf{y}|\mathbf{f})] = \sum_{i=1}^{N} \mathbb{E}_{q(f(\mathbf{x}_n))}[\log p(y_n|f(\mathbf{x}_n))]$ where $q(\mathbf{f})$ can be derived in closed form from Eq. (2.8) and it is given by $q(\mathbf{f}) = \mathcal{N}(\mathbf{K}_{zx}\mathbf{K}_{zz}^{-1}\mathbf{m}, \mathbf{K}_{xx} - \mathbf{K}_{xz}\mathbf{K}_{zz}^{-1}\mathbf{K}_{zx} + \mathbf{K}_{zx}\mathbf{K}_{zz}^{-1}\mathbf{S}(\mathbf{K}_{zx}\mathbf{K}_{zz}^{-1})^T)$. The expected log likelihood term lends itself to stochastic optimization using mini-batches by sub-sampling the sum over N data points [Hensman et al., 2013]. This makes the algorithm independent of N and dominated by algebraic operations that are $\mathcal{O}(M^3)$ in time, where M is the number of inducing points. Inference is thus converted into a maximizing problem of the ELBO in Eq. (2.9) with respect to $\boldsymbol{\nu}_\mathbf{u}$ using gradient-based optimization methods.

2.2 Multi-task Gaussian Processes

We have so far considered single-task learning settings where, given a set of input-output training points $\mathcal{D}$, we wish to learn a mapping $f : \mathbb{R}^D \to \mathbb{R}$ with $D \in \mathbb{N}$ and make predictions on an unseen data-point $\mathbf{x}^\star$. However, the output variable does not need to be one dimensional and multiple outputs or *tasks* could be considered jointly. Multi-task learning (MTL) is an active research area in machine learning and has received a lot of attention over the past few years,

see Caruana [1998] for an early reference and Zhang and Yang [2021] for an overview. The key idea of MTL is that learning tasks simultaneously and sharing information across them can lead to improved performance in comparison to learning the same tasks individually. As we shall see in Chapter 7, MTL is also useful for decision-making algorithms as it allows exploiting all available information across correlated processes thus leading to correct uncertainty quantification and, as a consequence, efficient exploration of available actions. In GP models we consider the mappings $f_p(\mathbf{x})$ for $p = 1, \ldots, P$ where P gives the number of tasks and, for each function f_p, we might have a different training set $S_p = (\mathbf{X}_p, \mathbf{Y}_p) = (\mathbf{x}_{p,1}, y_{p,1}), \ldots, (\mathbf{x}_{p,N_p}, y_{p,N_p})$ with task-specific cardinality N_p.

A variety of multi-task GP models have been proposed in the literature. Among the most commonly used we find the intrinsic coregionalization model (ICM) [Goovaerts et al., 1997] and the linear model of coregionalization (LCM) [Goovaerts et al., 1997; Journel and Huijbregts, 1976] that were developed in the context of geostatistics. In LCM each output is expressed as a linear combination of independent random functions that are shared across tasks:

$$f_p(\mathbf{x}) = \sum_{q=1}^{Q} a_{p,q} u_q(\mathbf{x})$$

where $a_{p,q}$ are scalar coefficients and the latent function $u_q(\mathbf{x})$ has zero mean and covariance function $k_q(\mathbf{x}, \mathbf{x}')$. This ensures that the resulting covariance function expressed jointly over all the outputs, that is $K(\mathbf{x}, \mathbf{x}') = \sum_{q=1}^{Q} \mathbf{B}_q k_q(\mathbf{x}, \mathbf{x}')$ where each $\mathbf{B}_q$ is known as *coregionalization matrix*, is a valid positive definite function. The ICM is a simplified version of LCM where all latent functions are assumed to have the same covariance function. Conversely, in the semi-parametric latent factor model (SLFM) proposed by Teh et al. [2005a] all functions are assumed to have a difference covariance structure.

Apart from different assumptions on the latent function covariance structure, all these models involve "*instantaneous mixing*" [Álvarez et al., 2012] through a linear weighted sum of independent processes to construct correlated processes. This means that the output function f_p evaluated at $\mathbf{x}$ only depends on the values of the latent functions also evaluated at the same input $\mathbf{x}$. This leads to an overall kernel function across outputs that has a separable form. An alternative way to mix the latent functions and induces more complex covariance structures is given by convolutions. In convolution processes [Álvarez and Lawrence, 2011] each function is written as the convolution of base processes with a smoothing

kernel:

$$f_p(\mathbf{x}) = \sum_{q=1}^{Q} \int_{\mathcal{X}} G_{p,q}(\mathbf{x} - \mathbf{z}) u_q(\mathbf{z}) d\mathbf{z} + w_p(\mathbf{x})$$

where $\{w_p(\mathbf{x})\}_{p=1}^{P}$ and $\{u_q(\mathbf{x})\}_{q=1}^{Q}$ are independent GPs and each kernel $G_{p,q}(\mathbf{x})$ is a continuous function, also known as *smoothing kernel*, with compact support [Hörmander, 2007] or square-integrable [Higdon, 2002; Ver Hoef and Barry, 1998]. In the latent force model proposed by Álvarez et al. [2009], each smoothing kernel corresponds to a Green's function arising from a second order ordinary differential equation. As we will see in Chapter 7, this formulation is important when sharing information in a causal setting. Indeed, a multi-task model for functions defined on a causal graph can be developed by considering a convolution process with smoothing kernels that can be interpreted as Green's functions capturing the graph topology. Notice that the convolution of a GP is also a GP therefore convolutions have been used to construct a variety of more complex covariance functions, see Álvarez et al. [2012] for a review on convolution process and more generally on multi-output GP models.

Finally notice that, while all these methods have been originally developed in the context of Gaussian likelihood, they have been used as building blocks of complex probabilistic models for non-Gaussian likelihood [Chai, 2012; Dezfouli and Bonilla, 2015; Moreno-Muñoz et al., 2018; Skolidis and Sanguinetti, 2011]. In Chapter 4 we will focus on count data and show how multi-output GPs can be used to jointly model different point processes in a spatio-temporal region.

2.3 Poisson Point Processes

Poisson point processes (PPP) are stochastic processes used to model the distribution of random occurrences of points in a multidimensional space. In PPP, both the number of points and their locations are modelled as random variables. This means that a realization of a PPP in a state space $\mathcal{S}$, which is generally the Euclidean space $\mathcal{S} = \mathbb{R}^m$ with $m \geq 1$ or some subset thereof, comprises the number $N \geq 0$ and the locations $\mathbf{x}_1, \ldots, \mathbf{x}_n$ of the points in $\mathcal{S}$. The realization is denoted by the ordered pair $\xi = (N, \{\mathbf{x}_1, \ldots, \mathbf{x}_n\})$. Every PPP is parametrized by a quantity called the intensity which takes different forms depending on whether the state space $\mathcal{S}$ is continuous, discrete, or discrete-continuous. In this thesis we focus on continuous state spaces and define the intensity as a non-negative function $\lambda(s) : \mathcal{S} \to [0, \infty)$. If $\lambda(s) = \alpha$ for some constant $\alpha \geq 0$ the PPP is said to be *homogeneous*. Otherwise we have a *non-homogeneous* PPP and speak about intensity function which describes the expected number of points found in any bounded region of some arbitrary

domain. While the intensity function $\lambda(s)$ does not need to be continuous, it needs to satisfy $0 \leq \int_B \lambda(s)ds \leq \infty$ for all bounded subsets $B \subseteq \mathcal{S}$. We call $\Lambda(B) = \int_B \lambda(s)ds$ the intensity measure. More formally, a PPP is defined as:

Definition 2.2. (Poisson point process) A point process X on $\mathcal{S}$ is a Poisson point process with intensity function λ and intensity measure Λ if the following two properties hold:

- for any $B \subseteq \mathcal{S}$ such that $\Lambda(B) < \infty$, $N(B) \sim \text{Poisson}(\Lambda(B))$ that is the Poisson distribution with mean $\Lambda(B)$.
- For any $n \in \mathbb{N}$ and $B \subseteq \mathcal{S}$ such that $0 < \Lambda(B) < \infty$ we have

$$X_B | N(B) = n \sim \text{Binomial}\,(B, n, \lambda(s)/\Lambda(B))$$

  where Binomial$(\cdot)$ denotes a Binomial Point process [278].

We write $X \sim \text{Poisson}(\mathcal{S}, \lambda)$. For any bounded $B \subseteq \mathcal{S}$, the intensity function determines the expected number of points in B that is $\mathbb{E}[N(B)] = \Lambda(B)$. There are two basic operations for a PPP that will be exploited in this thesis: thinning and superposition. These are defined as:

Definition 2.3. (Superposition) A disjoint union $\bigcup_{i=1}^{\infty} X_i$ of point processes $X_1, X_2, ...$ is called a superposition.

Definition 2.4. (Thinning) Let $p : \mathcal{S} \to [0, 1]$ be a function and X a point process on $\mathcal{S}$. The point process $X_{thin} \subseteq X$ obtained by including $s \in X$ in X_{this} with probability $p(s)$, where the point are included or excluded independently on each other, is said to be an independent thinning of X with retention probabilities $p(s)$.

2.3.1 Cox processes

The intensity function of a PPP is generally unknown and another stochastic process is typically used to model it. In this case the process is called Cox process (CP) [Cox, 1955] and is characterized by the following properties:

- CP1: $\Lambda = \{\lambda(\mathbf{x}) : \mathbf{x} \in \mathcal{S}\}$ is a non-negative-valued stochastic process;
- CP2: conditional on the realization $\lambda(\mathbf{x}) : \mathbf{x} \in \mathcal{S}$ the point process is a non-homogeneous Poisson process with intensity $\lambda(\mathbf{x})$.

A CP is also called *doubly stochastic Poisson process* and is defined as:

Definition 2.5. (Cox Process) Denote by $Z = \{Z(s) : s \in \mathcal{S}\}$ a non-negative random field such that with probability one, $Z(s)$ is a locally integrable function. If $X|Z \sim \text{Poisson}(\mathcal{S}, Z)$ then X is said to be a Cox process driven by Z.

Note that independent thinning of a Cox process results in a new Cox process. In addition, the moment properties of a Cox process are inherited from those of the intensity process and are easily derived by conditioning on the random intensity and exploiting the properties of the Poisson process $X|Z$. For example, the intensity of X is equal to the expectation of Z and the covariance density of the Cox process is equal to the covariance function of Z.

2.3.2 GP modulated Cox processes

Among the variety of approaches adopted in the literature, GPs have been successfully used to model the intensity function of a CP [Adams et al., 2009; Fernandez et al., 2016; Gunter et al., 2014; López-Lopera et al., 2019; Møller et al., 1998; Rao and Teh, 2011] as they provide a flexible non-parametric Bayesian framework. In GP modulated Poisson processes, a GP prior is placed on a latent function $f(\mathbf{x})$ that is related to the intensity function of the non-homogeneous Poisson process through a link function $g(\cdot)$. This is usually taken to be the exponential transformation which results in the Log Gaussian Cox process [Møller et al., 1998]:

Definition 2.6. (Log Gaussian Cox Process) Let X be a Cox process on $\mathcal{S}$ driven by the intensity $\lambda(\cdot) = \exp(f(\cdot))$ where $f(\cdot)$ is a GP. Then X is said to be a Log Gaussian Cox Process (LGCP).

Other link functions frequently considered in the literature are the sigmoidal transformation which leads to the so-called sigmoidal Gaussian Cox Process (see, e.g., [Adams et al., 2009]) and the square transformation which leads to the Permanental Cox Process (see, e.g., [Lloyd et al., 2015]) These models are computationally challenging as they are doubly intractable [Murray et al., 2006]. Indeed, the posterior distribution on the latent function f can be computed as:

$$p(f|\mathcal{D}) = \frac{p(f)\exp(-\int_{\mathcal{X}} g(f(\mathbf{x}))\mathrm{d}\mathbf{x})\prod_{i=1}^{n} g(f(\mathbf{x}_i))}{\int p(f)\exp(-\int_{\mathcal{X}} g(f(\mathbf{x}))\mathrm{d}\mathbf{x})\prod_{i=1}^{n} g(f(\mathbf{x}_i))\mathrm{d}f}. \tag{2.10}$$

The likelihood function in the numerator involves an integral of the process over the spatio-temporal domain, which in general cannot be computed analytically. Computing $p(f|\mathcal{D})$ also requires computing the marginal likelihood in the denominator which in turn involves a double, generally intractable, integral. In Chapter 4 and Chapter 5 we will see how variational inference can be used to derive a scalable inference scheme for GP modulated PPP.

2.4 Why GPs?

All the work to follow, both in terms of modelling approaches and sequential decision-making frameworks, is build upon the power of GPs. We will in

particular spend Chapter 4 and Chapter 5 developing models and associated inference techniques that showcase the flexibility of GPs in capturing the behaviour of PPP. Chapter 6, 7 and 8 will instead demonstrate how GPs allow to properly quantify uncertainty in a variety of decision-making settings. Before delving into the core material of this thesis we now discuss the main advantages and shortcomings of using GPs.

2.4.1 Advantages of GPs

GP models are well-suited for building complex probabilistic models and sequential decision making algorithms. Indeed, we can identify the following advantages:

- **Expressivity**. Through the choice of a covariance function, GPs can express a wide range of modeling assumptions. For instance, as mentioned in Section 2.1, the RBF kernel or the Matérn kernel can be used to encode different degrees of function smoothness. Traditional GP models have also been extended to more expressive variants, for example by considering sophisticated covariance functions [Durrande et al., 2011; Remes et al., 2017; Wang et al., 2020] or by embedding GPs in more complex probabilistic structures [Damianou and Lawrence, 2013; Snoek et al., 2014; Ton et al., 2018; Wilson et al., 2011b] able to learn more powerful representations of the data.

- **Tractability**. When dealing with Gaussian likelihoods, the posterior GP distributions and predictive distributions are available in closed form. This is a rare property for non-parametric models to have and it is particularly useful in the context of sequential decision-making.

- **Uncertainty quantification**. Being a Bayesian non-parametric technique, GPs are capable of quantifying uncertainty through Bayesian inference which is vital for improving regularization. Note that the notion of uncertainty discussed here is very different from that used in other contexts. We refer to *epistemic* uncertainty, that is uncertainty representing our personal lack of knowledge about a problem, rather than aleatoric uncertainty due to inherent stochasticity in the system. Through GPs we can encode prior beliefs about a functional form and propagate the prior uncertainty when computing posterior distributions. This ensures correctly large uncertainty estimates in regions with little data avoiding overconfidence if the model is faced with a prediction task from e.g. a different input distribution. In addition, uncertainty in the inputs [Titsias and Lawrence, 2010] and in the kernel hyperparameters [Lalchand and Rasmussen, 2020] can be accounted for in GP models.

2.4.2 Limitations of GPs

There are several issues which make GPs sometimes difficult to use:

- **Computational cost**. Computing posterior and predictive distributions require the inversion of $N \times N$ matrices taking $\mathcal{O}(N^3)$ time. Exact inference is thus prohibitively slow for more than a few thousand data points. However, as mentioned above, a variety of sparse GP approximations can be used to address this problem.

- **Approximate inference for non-Gaussian likelihoods**. When using Gaussian likelihood the predictive distribution of a standard GP model is Gaussian. In various settings, we might want to consider non-Gaussian likelihoods e.g. to perform classification or in order to be robust to outliers. This requires approximate inference schemes which might negatively impact inference accuracy and computational cost.

- **Kernel choice and hyperparameters optimization**. The expressiveness and flexibility of GP stems from the possibility to chose a kernel function that adapts to the problem at hand. Choosing a kernel is equivalent to learning a useful representation of the inputs and, until recently, expert elicitation was required to choose the right parametric form. More recently, automatic kernel selection schemes were proposed with the goal of automatizing its construction given a dataset and make kernel learning more generally applicable [Duvenaud et al., 2013; Lloyd et al., 2014]. In addition, given a certain kernel function, the classical approach for learning the hyperparameters entails maximizing the marginal likelihood to get fixed point estimates. Extending the Bayesian treatment to hyperparameters in a hierarchical framework leads to a posterior which is highly intractable. In turn, this also renders the predictive posterior intractable and requires an additional approximate inference scheme. This can once again slow down inference and reduce prediction accuracy.

Chapter 3

Background Part B: Sequential Decision-Making

Sequential decision-making is a central ability of any intelligent agent interacting with an environment and faced with the problem of choosing a sequence of actions delivering the highest reward while accounting for uncertainty in the system. Sequential decision-making has attracted a lot of attention in the past years and frameworks such as Reinforcement Learning (RL) [Sutton and Barto, 2018], Multi-Armed Bandits [Lattimore and Szepesvári, 2020], Bayesian Optimization (BO) [Mockus, 2012] and Active Learning (AL) [Settles, 2012] have been developed with the goal of driving the agent's exploration and exploitation of available actions. These methods are dealing with different assumptions in terms of action space features, environment characteristics, and reward structure. In this thesis, we will focus on BO and briefly touch on AL in Chapter 7. In this chapter, we review the standard BO framework and introduce the causal concepts that will be used in the following chapters to extend BO and multi-task GP models to incorporate causal information.

3.1 Bayesian Optimization

Bayesian optimization (BO) is a sequential decision-making algorithm that can be used for optimizing an unknown, usually multimodal and expensive to evaluate function $f : \mathcal{X} \to \mathcal{Y}$ over the input space $\mathcal{X}$:

$$\mathbf{x}^\star = \arg\min_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}) \tag{3.1}$$

where $\mathcal{X}$ is often a compact subset of $\mathbb{R}^D$ but more unusual search spaces that involve categorical or conditional inputs can be considered. We sometimes refer to $\mathcal{X}$ as *action space* where each action, e.g. $x = 1$, corresponds to a different function evaluation, e.g. $f(1)$, in a BO setting. The unknown function is a

black-box and it can only be observed via point evaluations which are often corrupted by noise. The key idea of BO is to use a *surrogate model* to carry out the optimization and define a *utility function* to collect new data points satisfying some optimality criterion. Using a probabilistic model as a surrogate allows us to correctly quantify uncertainty. The utility function represents our design goal which could be the exploration of the function values or could encourage exploitation of the optimal function values found throughout the optimization. While different models have being used as a surrogate for BO e.g. random forests [Hutter et al., 2011], t-processes [Shah et al., 2014] and neural networks [Snoek et al., 2015], here we will focus on GPs as they have been successfully used in a variety of applications including environmental monitoring [Marchant and Ramos, 2012], robotics [Martinez-Cantin et al., 2007] and experimental design [Azimi et al., 2012].

3.1.1 GP surrogate models

In BO, the surrogate model includes the prior distribution that captures our beliefs about the behaviour of the unknown objective function and an observation model that describes the data generating mechanism:

$$\begin{aligned} f(\mathbf{x}) &\sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')) \\ y &= f(\mathbf{x}) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma^2). \end{aligned}$$

This corresponds to the GP regression discussed in Section 2.1. Given a dataset $\mathcal{D}$, posterior mean $m(\mathbf{x}|\mathcal{D})$ and variance $\sigma^2(\mathbf{x}|\mathcal{D})$ can be computed explicitly similarly to Eq. (2.3) - (2.5). There are two main reasons why GPs are popular surrogate models for BO. Firstly, unlike other non-parametric probabilistic models, GPs require the specification of only a handful of model parameters, namely the kernel hyperparameters and the mean function parameters if this is different from zero, rather than the many thousands required for Bayesian neural networks. Despite that, as mentioned in Section 2.4, GPs are flexible models that can express a wide range of modelling assumptions through the choice of a covariance function[1]. Secondly, GPs provide well-calibrated uncertainty estimates. This, coupled with the closed-form predictive distributions that can be obtained with Gaussian likelihoods, allow the fast and easy evaluation of acquisition functions.

Note that a variety of kernel functions can be used for $k(\mathbf{x}, \mathbf{x}')$. In the

[1] Based on Ghosal and Roy [2006], it is possible to show that GPs based on universal kernels [Steinwart, 2001] have sample paths which are arbitrarily close to any continuous function. See van der Wilk et al. [2017] for an interesting discussion on the universal approximation property for GPs.

following discussion, an RBF kernel is used unless otherwise stated. As in GP regression, the choice of the kernel function should reflect our beliefs in terms of function differentiability and smoothness. Importantly, every kernel function comes with a set of hyperparameters that need to be either estimated or marginalised. A preponderance of literature on GPs addresses this problem through maximization of the marginal likelihood, ML-II [MacKay, 1999]. Indeed, once the point estimate hyperparameters have been selected, the posterior distribution over latent function values and hence predictions can be derived in closed form. However, this approach suffers from two main issues. On the one hand, the non-convexity of the marginal likelihood implies that local optima could be found during the optimization. On the other hand, using point estimates of hyperparameters yields overconfident predictions, by failing to account for hyperparameters' uncertainty. Extending the Bayesian treatment to hyperparameters in a hierarchical framework leads to an intractable posterior and thus requires resorting to approximate inference methods, see e.g. Lalchand and Rasmussen [2020] for an example of a fully bayesian GP regression. Therefore, in this thesis we estimate the kernel hyperparameters by maximizing the marginal likelihood via the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm.

3.1.2 Acquisition functions

The posterior parameters of the surrogate model are used by the acquisition function, whose purpose is to guide the search for the optimum. The acquisition function $\alpha : \mathcal{X} \to \mathbb{R}$ is defined on the action space, leverages the uncertainty in the posterior distribution to guide the agents' exploration, and trades off exploration and exploitation of actions. At each iteration, one sample is gathered from $f(\mathbf{x})$ at a location selected by maximizing $\alpha(\mathbf{x}; \mathcal{D})$, which is a simpler and faster optimization procedure compared to the original problem of optimizing $f(\mathbf{x})$. A variety of acquisition functions have been proposed in the literature (see [Shahriari et al., 2015] for a review) which are giving rise to distinct sampling behaviours. The three most popular choices are the Probability of Improvement (PI) [Kushner, 1964], the Upper Confidence Bound [Cox and John, 1992] and the Expected Improvement (EI) [Mockus et al., 1978]. The EI will be used in this thesis as it is simple and readily implementable while offering reasonable performance in practice[2]. The EI favours points that are likely to improve upon the best function value $y^\star$ observed and, differently from PI, incorporates the *amount* of improvement. It corresponds to the expected value of a utility function that is called the improvement function $I(\mathbf{x}) = (f(\mathbf{x}) - y^\star)\mathbb{I}(f(\mathbf{x}) > y^\star)$

[2]See Wang and de Freitas [2014] for a study of the convergence rate of the EI and a discussion of its properties.

and can be written as:

$$\alpha_{\text{EI}}(\mathbf{x}; \mathcal{D}) = \mathbb{E}[\max(f(\mathbf{x}) - y^\star, 0)]$$

where the expectation is taken with respect to the distribution of f which is given by the surrogate model. When the surrogate model is a GP the EI can be computed in closed form and it is equal to:

$$\alpha_{\text{EI}}(\mathbf{x}; \mathcal{D}) = \begin{cases} (m(\mathbf{x}; \mathcal{D}) - y^\star)\Phi(Z) + \sigma(\mathbf{x}; \mathcal{D})\phi(Z) & \text{if} \quad \sigma(\mathbf{x}; \mathcal{D}) > 0 \\ 0 & \text{if} \quad \sigma(\mathbf{x}; \mathcal{D}) = 0 \end{cases}$$

with $Z = \frac{m(\mathbf{x};\mathcal{D}) - y^\star}{\sigma(\mathbf{x};\mathcal{D})}$ and Φ and ϕ representing the CDF and PDF of the standard normal distribution, respectively. It is critical for the acquisition function to be quick and cheap to evaluate or approximate with respect to the black-box function f. Indeed, in BO the original problem in Eq. (3.1) is translated into the problem of optimizing the acquisition function. Clearly, the alternative optimization must incur a lower computational cost than the maximisation of the original objective function for BO to be a feasible optimisation strategy. At step t in the optimization, the next sampling point is determined by:

$$\mathbf{x}_t^\star = \operatorname*{argmax}_{\mathbf{x} \in \mathcal{X}} \alpha(\mathbf{x}; \mathcal{D}_{t-1})$$

where $\mathcal{D}_{t-1}$ denotes the dataset including the function observations collected during the first $t-1$ steps. Fig. 3.1 shows a posterior GP model (left plot) and the associated EI acquisition function (right plot) after having collected three data points. Notice how the EI is higher in areas where the uncertainty in the surrogate model is large (exploration) and/or where the model prediction is low (exploitation) therefore a minimum is expected. The next optimal value to collect, which corresponds to the red line in the right plot, is characterized by a high posterior variance and a low posterior mean therefore it balances exploration and exploitation.

BO has been extended to support a broad class of common high-cost optimization tasks. This includes multi-fidelity settings [McLeod et al., 2017; Song et al., 2019; Swersky et al., 2013], batch optimization [Alvi et al., 2019; González et al., 2016a], non-myopic optimization [González et al., 2016b; Jiang et al., 2020; Yue and Kontar, 2020], dynamic settings [Nyikosa et al., 2018], constrained problems [Gelbart et al., 2014] and multi-objective optimization [Wada and Hino, 2019]. Some works have focused on combining BO with more explorative searches [Ahmed et al., 2016; Falkner et al., 2017] thus yielding "safe" BO methods or on incorporating derivative information into the algorithm

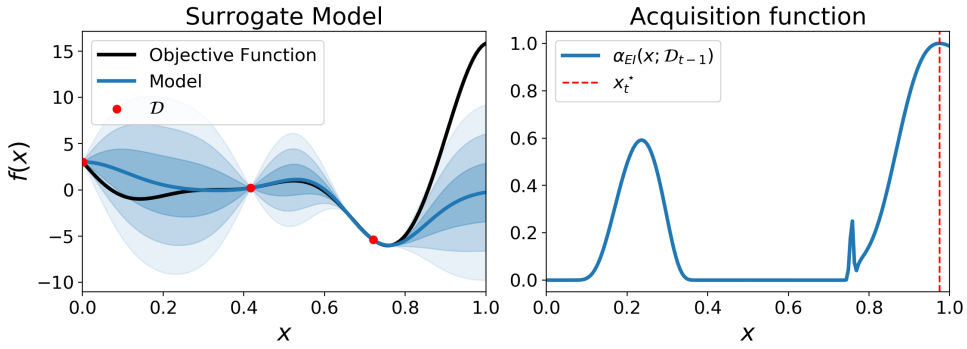


Figure 3.1: *Left plot:* Posterior GP surrogate model for a BO problem where three data points (red dots) are observed from the true underlying objective function (black line). The blue line gives the posterior mean while the shaded areas represent posterior uncertainty ($\pm$ 1, 2 and 3 standard deviations). *Right plot:* EI acquisition function computed based on the posterior parameters of the GP model on the left. At every step in the optimization, BO selects x by maximizing the acquisition function α_{EI}. Therefore, the next optimal observation to collect is highlighted in red and corresponds to $x = 1$.

[Ahmed et al., 2016; Wu et al., 2017].

In addition, various BO algorithms have been proposed to deal with discrete and highly structured input spaces [Garnett et al., 2010; Moss et al., 2020; Ru et al., 2020; Wan et al., 2021]. Despite the success of BO across different applications and problem settings, BO has been generally used to solve problems of moderate dimension. Several workshops on BO have identified its scaling to high dimensions as one of the main challenges. Indeed, to ensure that a global optimum is found, we require good coverage of the input space but, as the dimensionality increases, the number of evaluations needed to cover it increases exponentially. Different approaches have been proposed in the literature to tackle this issue [Chen et al., 2012; Eriksson et al., 2019; Moriconi et al., 2019; Wan et al., 2021; Wang et al., 2013]. Among these, Wang et al. [2016] develop an algorithm called BALD that uses random embeddings to reduce the problem dimensionality and can be used when the objective function has low intrinsic dimensionality. In the following discussion (Chapter 6) we will see how their idea can be formalized and made explicit by taking a causal perspective on the optimization problem.

In this thesis, we will extend BO to incorporate causal information. In particular, we will see how complex surrogates based on GP models, both single-task (Chapter 6) and multi-task (Chapter 7), can be developed in order to select interventions to perform in a causal system. Incorporating causal assumptions reduces the dimensionality of the optimization problem in a principled way and allows us to integrate different types of data thereby correctly quantifying

uncertainty. As we shall see in the second part of this thesis, this enables the development of efficient BO schemes that can be used to optimize target variables that are part of a causal graph, both in static and dynamic settings (Chapter 8). Before getting into the details of the causal extension, we will now review some causal inference concepts and ideas in structural equation models that will be used in the remainder of this thesis.

3.2 Causality and Decision-Making

As discussed in the introduction, causal reasoning has been recognised as a central feature of human beings, crucial in many aspects of their thought processes. Given its centrality, we would like to develop automated decision-making algorithms that encode and reason in terms of cause-effect relationships, especially when we aim at understanding a data generating mechanism and potentially manipulate it. Incorporating causal information into decision-making frameworks would allow us to (i) understand the causes of a certain outcome thus increasing interpretability; (ii) account for the existence of unobserved variables in the environment; (iii) compute counterfactual scenario and finally (iv) improve the generalization capabilities of the algorithm.

3.2.1 Two frameworks for causal inference

While the study of causality can be traced back hundreds of years and was discussed by philosophers such as Hume or Kant[3], two main frameworks have been adopted in the fields of statistics and machine learning. These are (i) the Potential Outcome (PO) framework, associated with the work by Donald Rubin [Rubin, 2005], building on the work on randomized controlled trials (RCT) from the 1920s by Ronald Fisher and Jerzey Neyman, and (ii) the work on Directed Acyclic Graphs (DAGs), much of it associated with work by Judea Pearl and his collaborators [Pearl, 2009b].

While Pearl [2009a] has shown how the two frameworks are equivalent, that is an assumption in one framework can be translated to its counterpart in the other, we can identify some major differences between the two approaches. First of all, the two methods differ in the use of graphical representations. In Pearl's framework, all assumptions are encoded in a structural causal model (SCM) and the related DAG which gives a clear visual representation. DAGs provide a graphical tool to represent the causal system underlying a research question making it easier to interpret and assess the overall model. In the PO framework, causal assumptions (stable unit treatment value, consistency,

[3]See Section 1.1 for a brief historical overview on causality.

and ignorability also called unconfoundedness) are expressed in the form of conditional independence relationships involving counterfactual variables and are thus difficult to articulate. In addition, while graphical representations give us a way to check identifiability of causal effects [Tian and Pearl, 2002], verifying whether the PO assumptions are complete, that is they are sufficient for deriving causal quantities, is challenging.

More importantly, the object of analysis in the PO framework is the unit-based response variable where unit stands for an individual experimental subject. This is the value that an outcome variable, say Y, would obtain in experimental unit u, had treatment X been x. Given the focus on the unit, in the PO framework the causal effects of the variables other than the treatment and the special variables e.g. instrumental variable are not defined. This is a strength of this framework as we can model the interesting causal effects without knowing the complete causal graph. In the DAGs approach, the focus is on the data-generating mechanism and all causal quantities of interests can be defined starting from the causal model, whose DAG is assumed or discovered from data [Glymour et al., 2019]. Once a causal model is defined, we can study the causal effect of any variable and compute any counterfactual scenario. Therefore, to learn causal relationships among an arbitrary set of variables, Pearl's framework is often preferred.

The two frameworks also differ in terms of interpretation of interventions. The DAGs approach views the intervention on a variable, e.g. Y, as an operation that changes its distribution while still keeping it in the SCM. The PO approach views the variable Y under an intervention to be a different variable, say $Y_{X=x}$, loosely connected to Y and remaining unobserved. The problem of inferring $Y_{X=x}$ becomes a missing data problem.

Finally, with respect to the DAGs literature, primarily concerned with identification, studies within the PO approach literature have focused on *estimating* average causal effects of binary treatments. They have thus addressed important problems regarding study design, estimation and inference leading to powerful methods such as instrumental variables, difference-in-differences, regression discontinuity designs and synthetic control methods [Abadie, 2005, 2021; Abadie and Imbens, 2006; Angrist et al., 1996; Heckman and Vytlacil, 2005; Imbens and Lemieux, 2008; Rubin, 1973]. The question of which, how, and when different causal frameworks should be adopted is still very much open, we refer the reader to Aliprantis [2015]; Guo et al. [2020]; Imbens [2020] and [Pearl, 2009a] for interesting discussions on this topic.

Given our interest in understanding data-generating mechanisms while comparing and evaluating various *continuous* treatment variables, we adopt Pearl's framework in this thesis and review it in the next sections. Indeed, the DAGs framework can be used to uncover the underlying data-generating

processes and compute all causal effects existing in a DAGs in order to identify optimal interventions. In addition, the methodologies developed within the DAGs literature allow answering causal queries in complex models characterized by a large number of variables and where scalability might represent an issue.

3.2.2 Structural Causal Models

In order to deal rigorously with questions of causality, we must have a way of formally setting down our assumptions about the causal relationships behind observed data. To do so, we introduce the concept of structural causal model (SCM) which is a way of describing the variables relevant to the problem we are analysing and how they interact with each other, in other words their data-generating mechanism. More formally we can define a SCM as follow:

Definition 3.1. (Structural Causal Model) [Pearl, 2009b]. A structural causal model M is a four-tuple $\langle \mathbf{U}, \mathbf{V}, F, P(\mathbf{U}) \rangle$ where:

- $\mathbf{U}$ is a set of background variables, also called *exogenous* variables, that are determined by factors outside of the model and are distributed according to the probability distribution $P(\mathbf{U})$.
- $\mathbf{V}$ is a set $\{V_1, V_2, \ldots, V_{|\mathbf{V}|}\}$ of observable variables, also called *endogenous* variables, that are determined by variables in the model (i.e., determined by variables in $\mathbf{U} \cup \mathbf{V}$).
- F is a set of functions $\{f_1, f_2, \ldots, f_n\}$ such that each f_i is a mapping from the respective domains of $U_i \cup \mathrm{Pa}(U_i)$ to V_i, where $U_i \subseteq \mathbf{U}$ and $\mathrm{Pa}(U_i) \subseteq \mathbf{V} \setminus V_i$ and the entire set F forms a mapping from $\mathbf{U}$ to $\mathbf{V}$. In other words, each $\{f_i \in v_i \leftarrow f_i(\mathrm{Pa}(u_i), u_i) \mid i = 1, \ldots, n\}$, assigns a value to V_i that depends on the values of the select set of variables $(U_i \cup \mathrm{Pa}(U_i))$.

Every SCM is associated with a graphical causal model denoted by $\mathcal{G}$, see Fig. 3.2 for some examples. A graphical causal model consists of a set of nodes representing the variables in $\mathbf{U}$ and $\mathbf{V}$, and a set of edges between the nodes representing the functions in F.

While causal inference methods have been developed for both graphical models with [Bongers et al., 2016; Hyttinen et al., 2012; Koster, 1996; Mooij et al., 2011, 2013a; Neal, 2000; Pearl and Dechter, 1996; Richardson, 2013; Rothenhäusler et al., 2015; Spirtes, 1995] and without cycles, here we focus on directed *acyclic* graphs (DAGs). DAGs are directed graphs which means that all edges are marked by a single arrowhead on the edge. In addition, they are acyclic thus they do not contain directed cycles representing mutual causation or feedback

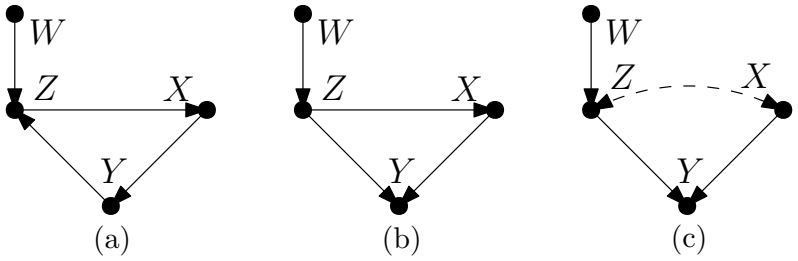


Figure 3.2: Examples of causal graphs. (a) Causal graph that is not a DAG as it contains a cycle between Z and Y. (b) Valid DAG where all variables are observed. (c) DAG with un unobserved confounder between Z and X represented by a dashed bidirected edge.

processes. For instance, the graph in Fig. 3.2(a) is not a DAG as it contains a cycle between Z and Y. While some of the basic definitions and properties given for acyclic graphs are also valid for cyclical models (e.g. causal effect definition given in Definition 3.2 and d-separation property mentioned in Definition 3.2) [Pearl and Dechter, 1996; Spirtes, 1995], diagrams involving directed cycles or feedback loops might present additional identification and inference issues. In particular, the computation of causal effects is generally harder in cyclic models. Notice however that focusing on DAGs is not a limitation as cyclic graphs can be converted in DAGs by explicitly accounting for time in the graph, similarly to what has been done to write systems of ordinary differential equations as structural causal model [Mooij et al., 2013a]. Extending the approaches presented in this thesis to directed *cyclic* graphs remains an open research direction.

We will use bidirected edges to denote the existence of unobserved common causes which are called *confounders*. These edges will be marked as dashed edges with two arrowheads (Fig. 3.2(c)). We call a *path* in $\mathcal{G}$ a sequence of edges (e.g. (W, Z), (Z, X), (X, Y) in Fig. 3.2(b)) such that each edge starts with the vertex where the preceding edge ends. If every edge in a path is an arrow that points from the first to the second vertex of the pair, we have a directed path. In the following discussion, we will make use of the terminology of kinship such as parents, children, descendants, ancestors to denote various relationships in $\mathcal{G}$ [Peters et al., 2017, Section 6.1]. In Fig. 3.2(b), for example, Y has two parents (X and Z), three ancestors (X, Z, and W), and no children, while X has no parents (hence, no ancestors) and one child (Y).

In a fully specified SCM with *no unobserved confounders*, we can represent the joint distributions of $|\mathbf{V}|$ variables with great efficiency. Indeed, for any model whose graph is acyclic, the joint distribution of the variables in the model is given by the product of conditional distributions of the form $P(\text{child}|\text{parents})$.

Formally, we write this rule as:

$$p(x_1, x_2, \ldots, x_N) = \prod_i p(x_i|\text{pa}_i) \tag{3.2}$$

where pa_i stands for the values of the parent variables for X_i, and the product $\prod_i$ runs over all $i = 1, \ldots, N$. The set pa_i gives the Markovian parents of X_i. A causal model that has Markovian parents and for which the factorization in Eq. (3.2) holds is called a Markovian model. For such models all causal effects are identifiable; that is they can be estimated consistently from non-experimental data. Non-Markovian models, such as those involving correlated errors which may result from unmeasured confounders, permit identification only under certain conditions [Tian and Pearl, 2002]. In this thesis we focus on DAGs for which causal effects are identifiable (see Tian and Pearl [2002] for identifiability conditions) and in the next section we discuss how, depending on the assumed causal graph, one can compute causal effects.

3.2.3 Causal Calculus

The ultimate aim of many statistical studies is to predict the effects of interventions. Randomized controlled trials (RCT) are considered the gold standard for assessing causal effects. Indeed, in RCT all factors influencing the outcome variable of interest Y are controlled and any change in Y must be due to the variables we are manipulating. In many settings, randomized controlled experiments cannot be performed and only observational studies can be conducted. In observational studies, variables are simply observed and not controlled. The difference between RCT and observational studies is reflected in the distributions they allow to estimate. On the one hand, observing a system allows us to collect *observational data* and estimate conditional distributions e.g. $P(Y|X = x)$ in Fig. 3.2(b). On the other hand, intervening in a system allows us to estimate the *interventional distribution* $P(Y|\text{do}\,(X = x))$ which denotes the distribution of Y when the variable X is intervened and fixed to x. These two might differ substantially depending on the structure of the causal graph. For instance, in Fig. 3.2(b), $P(Y = y|X = x)$ reflects the population distribution of Y among individuals whose X value is observed to be x. Note that X is caused by Z therefore observing a value of $X = x$ could be due to the value taken by Z which also affects Y directly. Looking at $P(Y = y|X = x)$ does not allow us to distinguish between the effect of X and the effect of Z (direct and indirect) on Y. This can be achieved by computing $P(Y = y|\text{do}\,(X = x))$ which represents the population distribution of Y if everyone in the population had their X value fixed at x independently of other variables in the causal graph. We similarly write $P(Y = y|\text{do}\,(X = x), Z = z)$ to denote the conditional probability of

$Y = y$, given $Z = z$, in the distribution created by the intervention $\text{do}(X = x)$. Causal effects are defined via interventional distributions:

Definition 3.2. (Causal effect) Given two disjoint sets of variables, X and Y, the causal effect of X on Y, denoted as $P(y|\text{do}(X = x))$ is a function from X to the space of probability distributions on Y. For each realization x of X, $P(y|\text{do}(X = x)$ gives the probability of $Y = y$ induced by deleting from the connected SCM all equations corresponding to variables in X and substituting $X = x$ in the remaining equations.

In order to compute causal effects, we need a set of rules that allows us to write *do*-distributions in terms of conditional distributions that can be estimated from observational data. These are the so-called rules of *do* calculus for which a proof is given in Pearl [1995]. Denote by $\mathcal{G}_{\overline{X}}$ the graph obtained by deleting from $\mathcal{G}$ all arrows pointing to nodes in X. Likewise, we denote by $\mathcal{G}_{\underline{X}}$ the graph obtained by deleting from $\mathcal{G}$ all arrows emerging from nodes in X. Finally, denote the deletion of both incoming and outgoing arrows by $\mathcal{G}_{\overline{X}\underline{X}}$.

Theorem 3.1. (Rules of *do* calculus) [Pearl, 1995] *Let $\mathcal{G}$ be the* DAG *associated with a causal model as defined in Definition 3.1 and let $P(\cdot)$ be the probability distribution induced by that model. For any disjoint subsets of variables X, Y, Z and W we have the following:*

- ***Rule 1*** *(insertion/deletion of observations):*

$$P(Y|do(X = x), Z, W) = P(Y|do(X = x), W)$$

  if $(Y \perp\!\!\!\perp Z)|X, W$ in $\mathcal{G}_{\overline{X}}$.

- ***Rule 2*** *(action/observation exchange):*

$$P(Y|do(X = x), do(Z = z), W) = P(Y|do(X = x), Z = z, W)$$

  if $(Y \perp\!\!\!\perp Z)|X, W$ in $\mathcal{G}_{\overline{X}\underline{Z}}$.

- ***Rule 3*** *(insertion/deletion of actions):*

$$P(Y|do(X = x), do(Z = z), W) = P(Y|do(X = x), W)$$

  if $(Y \perp\!\!\!\perp Z)|X, W$ in $\mathcal{G}_{\overline{X}\overline{Z(W)}}$ where $Z(W)$ is the set of Z-nodes that are not ancestors of any W-node in $\mathcal{G}_{\overline{X}}$.

See Pearl [1995] for the proof of Theorem 3.1. A causal effect such as e.g. $P(Y|\text{do}(X = x))$ is identifiable if there exists a finite sequence of transformations, each conforming to one of the inference rules in Theorem 3.1, that reduces

$P(Y|\text{do}\,(X = x))$ into a standard *do*-free probability expression involving observed quantities. The *do*-calculus has been shown to be complete [Huang and Valtorta, 2006; Shpitser and Pearl, 2006] that means that the three rules in Theorem 3.1 are sufficient for deriving all identifiable causal effects. Two specific cases of causal effects computation are given by scenario in which either the *back-door criterion* or the *front-door criterion* are fulfilled.

The back-door criterion Assume we are given a DAG together with observational data for the variables in $\mathcal{G}$ and we wish to estimate the effect on Y of an intervention on X, that is we seek to estimate $P(Y|\text{do}\,(X = x))$.

Theorem 3.2. (Back-door adjustment formula) *If a set of variables Z in $\mathcal{G}$ satisfies the* back-door criterion *relative to (X, Y) then the causal effect of X on Y is identifiable and is given by the formula:*

$$P(Y = y|do\,(X = x)) = \int_{\mathcal{Z}} P(Y = y|X = x, Z = z)P(Z = z)dz$$

where $\mathcal{Z}$ represents the domain of $P(Z)$.

A proof for Theorem 3.2 is given in [Pearl, 2009b]. The back-door criterion is a simple graphical test that can be applied directly to $\mathcal{G}$ in order identify the set Z:

Definition 3.3. (Back-door criterion) Given an ordered pair of variables (X, Y) in a DAG, a set of variables Z satisfies the backdoor criterion relative to (X, Y) if no node in Z is a descendant of X, and Z blocks every path between X and Y that contains an arrow into X.

The set Z represents a set of nodes blocking all spurious paths between X and Y while not modifying the directed paths between X and Y and not creating additional spurious paths. In other words X d-separates Z and Y where d-separation is defined as:

Definition 3.4. (d-separation) A path p is said to be d-separated (or blocked) by a set of nodes Z if and only if:

1. p contains a chain $i \to m \to j$ or a fork $i \leftarrow m \to j$ such that the middle node m is in Z, or

2. p contains an inverted fork (or collider) $i \to m \leftarrow j$ such that the middle node m is not in Z and such that no descendant of m is in Z.

A set Z is said to d-separate X from Y if and only if Z blocks every path from a node in X to a node in Y.

Note that $\text{Pa}(X)$ always satisfies the backdoor criterion in a DAG with no unobserved confounders. Indeed, every subset of nodes X in a causal graph is d-separated from the remaining nodes in the DAG by $\text{Pa}(X)$.

The front-door criterion An alternative criterion, "the front-door criterion", may be applied in cases where we cannot find observed covariates Z satisfying the back-door conditions. This is useful when the DAG contains unobserved confounders thus it is not possible to condition on their values to block the back-door paths.

Theorem 3.3. (Front-door adjustment formula) *If a set of variables Z in $\mathcal{G}$ satisfies the* front-door criterion *relative to (X, Y) then the causal effect of X on Y is identifiable and is given by the formula:*

$$P(Y = y|do\,(X = x)) = \int_{\mathcal{Z}} P(z|X = x) \int_{\mathcal{X}} P(Y = y|X = x', z)P(X = x')dz\,dx' \tag{3.3}$$

where $\mathcal{Z}$ represents the domain of $P(Z)$.

A proof for Theorem 3.3 was originally given in Pearl [1993]. The front-door criterion is defined as follow:

Definition 3.5. (Front-door criterion) A set of variables Z is said to satisfy the front-door criterion relative to an ordered pair of variables (X, Y) if:

- Z intercepts all directed paths from X to Y;
- There is no back-door path from X to Z;
- All back-door paths from Z to Y are blocked by X.

We conclude this section with a set of examples demonstrating how the back-door formula, the front-door formula, and more in general the Rules 1-3 of *do*-calculus can be used to derive causal effects for the DAG in Fig. 3.3(a). Here we are assuming all variables in the DAG to be continuous but similar derivations hold for discrete or dichotomous variables.

Computation of $\mathbb{E}[Y|\mathbf{do}\,(Z = z)]$ In order to compute this interventional distribution notice that X satisfies the back-door criteria for the pair (Z, Y) as it blocks all back-door paths from Z to Y. We can thus write:

$$\begin{aligned}\mathbb{E}[Y|\mathrm{do}\,(Z = z)] &= \int yp(Y = y|\mathrm{do}\,(Z = z))\mathrm{d}y \\ &= \int\int yp(Y = y|\mathrm{do}\,(Z = z)\,, x)p(X = x|\mathrm{do}\,(Z = z))\mathrm{d}y\mathrm{d}x.\end{aligned}$$

We now have to reduce the distributions exhibiting *do* operations to *do*-free expressions. Starting with $p(Y = y|\mathrm{do}\,(Z = z)\,, x)$, notice that we can apply Rule 2 of *do*-calculus. Indeed, $(Y \perp\!\!\!\perp Z)|X$ in $\mathcal{G}_{\underline{Z}}$ (Fig. 3.3(b)). We can thus exchange action with observation and write this distribution as $p(Y = y|Z =$

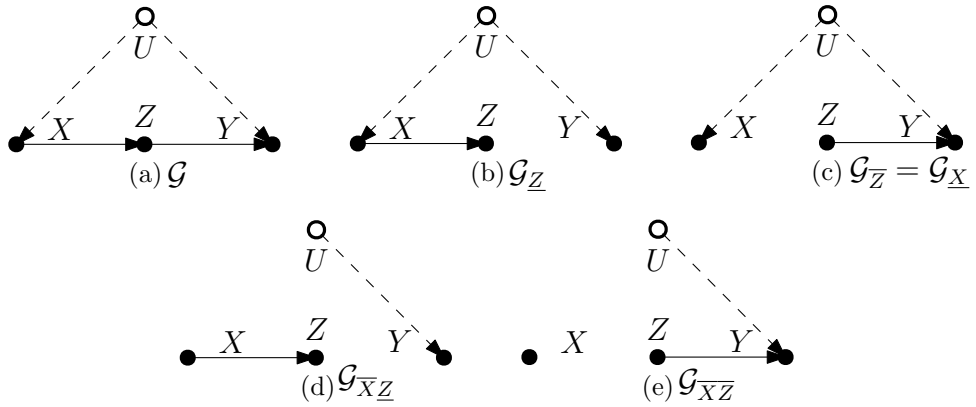


Figure 3.3: Example of a DAG (a) and the corresponding mutilated graphs used to derive various interventional distributions.

$z, x)$. Next, note that $p(X = x|\text{do}\,(Z = z))$ can be simplified resorting to Rule 3 of *do*-calculus as $(X \perp\!\!\!\perp Z)$ in $\mathcal{G}_{\overline{Z}}$ (Fig. 3.3(c)). We can thus write it as the marginal distribution $p(X = x)$. The complete expression becomes:

$$\mathbb{E}[Y|\text{do}\,(Z = z)] = \int\int yp(Y = y|Z = z, x)p(X = x)\text{d}y\text{d}x. \tag{3.4}$$

which correspond to the back-door adjustment formula where the variable to be manipulated is Z.

Computation of $\mathbb{E}[Y|\mathbf{do}\,(X = x)]$ In order to compute this interventional distribution notice that there exists a back-door path from X to Y that cannot be blocked by conditioning on observed variables. Indeed, the variable U is an unobserved confounder. We thus need to resort to the front-door adjustment formula and exploit the fact that the relationship between X and Z is not confounded. We start by adding Z to the expression:

$$\begin{aligned}\mathbb{E}[Y|\text{do}\,(X = x)] &= \int yp(Y = y|\text{do}\,(X = x))\text{d}y \\ &= \int\int yp(Y = y|\text{do}\,(X = x)\,, z)p(Z = z|\text{do}\,(X = x))\text{d}y\text{d}z.\end{aligned}$$

Notice that, exploiting Rule 2, we can write $p(Y = y|\text{do}\,(X = x)\,, z)$ with a double intervention on both X and Z that is $p(Y = y|\text{do}\,(X = x)\,, \text{do}\,(Z = z))$. Indeed $(Y \perp\!\!\!\perp Z)|X$ in $\mathcal{G}_{\overline{X}\underline{Z}}$ (Fig. 3.3(d)). In addition, by Rule 3 of *do* calculus, $p(Y = y|\text{do}\,(X = x)\,, \text{do}\,(Z = z)) = p(Y = y|\text{do}\,(Z = z))$ as $(Y \perp\!\!\!\perp X)|Z$ in $\mathcal{G}_{\overline{XZ}}$ (Fig. 3.3(e)). The expression for $p(Y = y|\text{do}\,(Z = z))$ is given in Eq. (3.4). Focusing now on $p(Z = z|\text{do}\,(X = x))$, notice that $(Z \perp\!\!\!\perp X)$ in $\mathcal{G}_{\underline{X}}$ (Fig. 3.3(c)) therefore action and observation can be exchanged again to write is a $p(Z = z|X = x)$. Plugging in these expression we can write the targeted interventional

distribution as:

$$\mathbb{E}[Y|\mathrm{do}\,(X=x)] = \int p(z|X=x) \int\int yp(Y=y|z,x')p(X=x')\mathrm{d}y\mathrm{d}x'\mathrm{d}z.$$

which corresponds to the front-door adjustment formula in Eq. (3.3).

Part I

Structured Inference of Gaussian Process Modulated Cox Processes

In the first part of this thesis, we will focus on developing scalable modelling frameworks for single-task and multi-task GP modulated PPP. In the second part, we will then see how these probabilistic models can be used as crucial building blocks of different decision-making algorithms. Specifically, in the next two chapters, we will first look at multi-task settings where the intensities of different point processes are jointly estimated and their likelihood function is discretized to allow for tractable inference. The *complex correlation structure* existing among the processes will be incorporated via a linear coregionalization type model where not only the function but also the mixing weights are Gaussian processes and capture tasks' similarities. We will then focus on the discretization issue and developed a continuous Cox process model that can be used for accurate predictions in high-dimensional input spaces. The complex dependencies in the posterior distribution will be retained by the *structured variational inference* scheme without compromising on the accuracy, scalability, and speed of the posterior approximation.

Chapter 4

Efficient Inference in Multi-task Cox Process Models

In this chapter we focus on the structured inference problem and consider the goal of jointly modelling different point processes happening in a given spatio-temporal domain. Indeed, many problems in urban science and geostatistics are characterized by count or point data observed in a spatio-temporal region. Crime events, traffic or human population dynamics are some examples. Furthermore, in many settings, these processes can be strongly correlated. For example, in a city such as New York (NYC), burglaries can be highly predictive of other crimes' occurrences such as robberies and larcenies. These settings are multi-task problems and our aim is to exploit such dependencies in order to improve the generalization capabilities of our learning algorithms and correctly quantify uncertainty.

Point processes in a spatio-temporal region can be modelled as non homogeneous processes where a space-time varying intensity determines event occurrences. As mentioned in Section 2.3, a popular modelling approach for non-homogeneous Poisson point processes (PPPs) is given by the log Gaussian Cox process (LGCP) where the intensity is driven by a Gaussian process (GP). The flexibility of LGCP comes at the cost of incredibly hard inference challenges due to its doubly-stochastic nature and the scalability issues of GP models. The computational problems are exacerbated when considering multiple correlated tasks and, therefore, the development of new approaches and scalable inference algorithms for LGCP models remains an active area of research [Coeurjolly et al., 2017; Cuevas-Pacheco and Møller, 2018; Diggle et al., 2013; Flaxman et al., 2015, 2019; Hessellund et al., 2020; Johnson et al., 2019; Leininger et al., 2017; Nasirzadeh et al., 2021; Shirota and Banerjee, 2019; Shirota and Gelfand, 2017; Simpson et al., 2016a; Taylor et al., 2015].

From a modelling perspective, existing multivariate LGCPs or linear core-

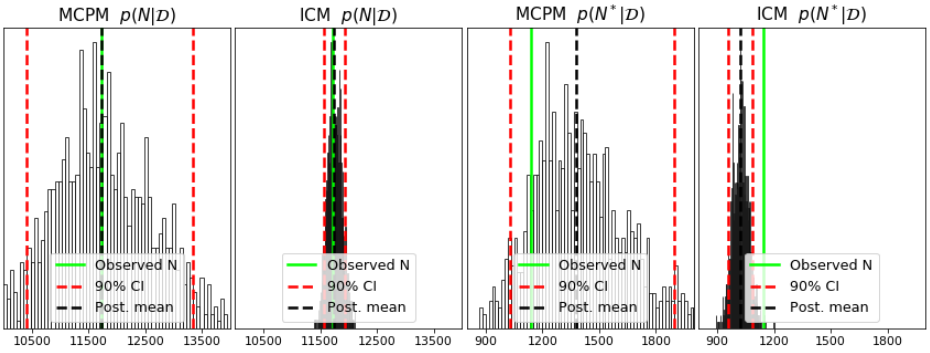


Figure 4.1: Posterior and predictive distributions, $p(N|\mathcal{D})$ and $p(N^*|\mathcal{D})$ respectively, of the number of burglary events in NYC using a similar analysis as in Leininger et al. [2017] on the CRIME dataset (Section 4.4.2) for our model (MCPM) and ICM. The solid line shows the ground truth. Details on the CI construction are given in Section 4.4.

gionalization model (LCM) variants for point processes have intensities given by *deterministic* combinations of latent GPs [Diggle et al., 2013; Taylor et al., 2015; Álvarez et al., 2012]. These approaches fail to propagate uncertainty in the weights of the linear combination, leading to statistical deficiencies that we aim at addressing in this chapter. For instance, Fig. 4.1 shows how, by propagating uncertainty, the approach developed in this chapter, henceforth MCPM, provides a predictive distribution that contains the counts' ground truth in its 90% credible interval (CI). This is not observed for the standard intrinsic coregionalization model (ICM). From an inference point of view, sampling approaches have been proposed [Diggle et al., 2013; Taylor et al., 2015] and variational inference algorithms for models with GP priors and 'black-box' likelihoods have been used [see e.g. Dezfouli and Bonilla, 2015; Matthews et al., 2017]. While sampling approaches have prohibitive computational cost [Shirota and Gelfand, 2016] and mixing issues [Diggle et al., 2013], generic methods based on variational inference do not exploit the LGCP likelihood details and, relying upon Monte Carlo estimates for computing expectations during optimization, can exhibit slow convergence. In this chapter we address the modelling and inference limitations of current approaches. More specifically, we make the following contributions.

Stochastic mixing weights We propose a model, henceforth referred to as MCPM, that considers correlated count data as realizations of multiple LGCPs, where the log intensities are linear combinations of latent GPs and the combination coefficients are also GPs. This provides additional model flexibility and the ability to propagate uncertainty in a principled way.

Efficient inference We carry out posterior estimation over both the latent and the mixing processes using variational inference. Our method is orders of magnitude faster than competing approaches.

Closed-form expectations in the variational objective We express the required expectations in the variational inference objective, the so called evidence lower bound, in terms of moment generating functions (MGFs) of the log intensities, for which we provide analytical expressions. We thus avoid Monte Carlo estimates, which are commonplace in modern variational inference methods and might slow down the convergence of the algorithm.

Experimental comparison and state-of-the-art performance We provide an experimental comparison between existing multi-task point process methods. This is important as there are currently two dominant approaches (based on the LGCP or on the Permanental process), for which there is little insight on which one performs better and under what settings. Furthermore, we show that our method provides the best predictive performance on two large-scale multi-task point process problems with very different spatial cross-correlation structures.

4.1 The MCPM model

Recall that the LGCP model is an inhomogeneous PPP with a stochastic intensity function [see e.g. Cox, 1955], where the logarithm of the intensity surface is a GP. Given a GP $f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), \kappa(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta}))$, the intensity function of a LGCP model can be written as $\lambda(\mathbf{x}) = \exp\{f(\mathbf{x})\}$. Conditioned on the realization of the intensity function, the number of points in an area, say A, is given by $y_A|\lambda(\mathbf{x}) \sim \text{Poisson}\left(\int_{\mathbf{x}\in A} \lambda(\mathbf{x}) d\mathbf{x}\right)$. Based on the LGCP model, in the next section we will introduce a multi-task framework that deals with multiple correlated point processes happening in the same spatio-temporal region.

4.1.1 Model formulation

We consider learning problems where we are given a dataset $\mathcal{D} = \{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^N$ where $\mathbf{x}_n \in \mathbb{R}^D$ represents the input and $\mathbf{y}_n \in \mathbb{R}^P$ gives the event counts for the P tasks. We aim at learning the latent intensity functions and make probabilistic predictions of the event counts. Our modelling approach, which we call MCPM, is characterized by Q latent functions which are uncorrelated a priori and are drawn from Q zero-mean GPs, i.e. $f_q|\boldsymbol{\theta}_f^q \sim \mathcal{GP}(\mathbf{0}, k(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta}_f^q))$,

for $q = 1, \ldots, Q$. Hence, the prior over the $N \times Q$ latent function values $\mathbf{f}$ is:

$$p(\mathbf{f}|\boldsymbol{\theta}_f^q) = \prod_{q=1}^{Q} p(\mathbf{f}_{\bullet q}|\boldsymbol{\theta}_f^q) = \prod_{q=1}^{Q} \mathcal{N}(\mathbf{f}_{\bullet q}; \mathbf{0}, \mathbf{K}_{xx}^q), \tag{4.1}$$

where $\boldsymbol{\theta}_f^q$ is the set of hyper-parameters for the q-th latent function and $\mathbf{f}_{\bullet q} = \{f_q(\mathbf{x}_n)\}_{n=1}^N$ denotes the values of latent function q for the observations $\{\mathbf{x}_n\}_{n=1}^N$. We model tasks' correlation by linearly combining the above latent functions with a set of *stochastic* task-specific mixing weights, $\mathbf{W} \in \mathbb{R}^{P \times Q}$, determining the contribution of each latent function to the overall LGCP intensity. We consider two possible prior distributions for $\mathbf{W}$, an independent prior and a correlated prior given by additional GPs.

Prior over weights We assume the mixing weights to be drawn from Q zero-mean GPs:

$$p(\mathbf{W}|\boldsymbol{\theta}_w^q) = \prod_{q=1}^{Q} p(\mathbf{W}_{\bullet q}|\boldsymbol{\theta}_w^q) = \prod_{q=1}^{Q} \mathcal{N}(\mathbf{W}_{\bullet q}; \mathbf{0}, \mathbf{K}_w^q),$$

where $\mathbf{W}_{\bullet q}$ represents the P weights for the q-th latent function and $\boldsymbol{\theta}_w^q$ denotes the hyper-parameters. For each task, the inputs for the GPs on the mixing weights are given by a set of task descriptors denoted by $H_p \in \mathbb{R}^{D'}$. These covariates, differently from the inputs of the latent function, are not defined on the spatio-temporal domain that is they don't change across locations but capture tasks' features at the global level. In the independent scenario, we assume uncorrelated weights across both tasks and latent functions by making $\mathbf{K}_w^q$ diagonal. Independently on the $\mathbf{K}_w^q$ structure, the observations across tasks are still correlated via the linear mixing of latent random functions.

Likelihood model The likelihood of observing events at locations $\{\mathbf{x}_{n_p}\}_{n_p,p}^{N_p,P}$ under P independent inhomogeneous Poisson processes each with rate function $\lambda_p(\cdot)$ is given by

$$P(\{\mathbf{x}_{n_p}\}_{n_p=1,p=1}^{N_p,P}|\boldsymbol{\lambda}) = \exp\left[-\sum_{p=1}^{P}\int_{\tau}\lambda_p(\mathbf{x})d\mathbf{x}\right]\prod_{p=1}^{P}\prod_{n_p=1}^{N_p}\lambda_p(\mathbf{x}_{n_p}),$$

where τ is the observation domain, $\{\mathbf{x}_{n_p}\}_{n_p=1,p=1}^{N_p,P}$ gives all the events observed across tasks and $\boldsymbol{\lambda} = \{\lambda_p\}_{p=1}^P$. Following a common approach, we introduce a regular computational grid [Diggle et al., 2013] on the spatial extent and represent each cell with its centroid. Under MCPM, the likelihood of the observed

counts $\mathbf{Y} = \{\mathbf{y}_n\}_{n=1}^N$ is defined as:

$$p(\mathbf{Y}|\mathbf{f}, \mathbf{W}) = \prod_{n=1}^{N} \prod_{p=1}^{P} \text{Poisson}\left(y_{np}; \exp\left(\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet} + \phi_p\right)\right),$$

where y_{np} denotes the event counts for the p-th task at $\mathbf{x}_n$, $\mathbf{W}_{p\bullet}$ represents the Q weights for the p-th task, $\mathbf{f}_{n\bullet}$ denotes the Q latent function values corresponding to $\mathbf{x}_n$, and ϕ_p indicates the task-specific offset to the log-mean of the Poisson process and it is thus a likelihood parameter.

As in the standard LGCP model, introducing a GP prior poses significant computational challenges during posterior estimation as naïvely, inference would be dominated by algebraic operations that are cubic in N. To make inference scalable, we follow the inducing-variable approach proposed by Titsias [2009b] and further developed by Bonilla et al. [2019]. See Section 2.1.2 for an introduction on inducing point approximations in GP regression.

We augment our prior over the latent functions in Eq. (4.1) with M underlying *inducing variables* for each latent process. We denote these M inducing variables for latent process q with $\mathbf{u}_{\bullet q}$ and their corresponding *inducing inputs* with the $M \times D$ matrix $\mathbf{Z}_q$. We will see that major computational gains are realized when $M \ll N$. Hence, we have that the prior distributions for the inducing variables and the latent functions are:

$$p(\mathbf{u}|\boldsymbol{\theta}) = \prod_{q=1}^{Q} \mathcal{N}(\mathbf{u}_{\bullet q}; \mathbf{0}, \mathbf{K}_{zz}^q)$$
$$p(\mathbf{f}|\mathbf{u}, \boldsymbol{\theta}) = \prod_{q=1}^{Q} \mathcal{N}(\mathbf{K}_{xz}^q (\mathbf{K}_{zz}^q)^{-1} \mathbf{u}_{\bullet q}, \mathbf{K}_{xx}^q - A_q \mathbf{K}_{zx}^q)$$

where $A_q = \mathbf{K}_{xz}^q (\mathbf{K}_{zz}^q)^{-1}$, $\mathbf{u}$ is the set of all the inducing variables. The matrices $\mathbf{K}_{xx}^q$, $\mathbf{K}_{xz}^q$, $\mathbf{K}_{zx}^q$ and $\mathbf{K}_{zz}^q$ are the covariances induced by evaluating the corresponding covariance functions at all pairwise rows of the training inputs $\mathbf{X}$ and the inducing inputs $\mathbf{Z}_q$. $\boldsymbol{\theta} = \{\boldsymbol{\theta}_f^q\}_{q=1}^Q$ represents the set of hyper-parameters for the Q latent functions. Note that integrating out $\mathbf{u}$ from the augmented prior distribution we can exactly recover the initial prior distribution (Eq. (4.1)).

4.2 Inference

Our goal is to estimate the posterior distribution over all latent variables given the data, i.e. $p(\mathbf{f}, \mathbf{u}, \mathbf{W}|\mathcal{D})$ which is given by:

$$p(\mathbf{f}, \mathbf{u}, \mathbf{W}|\mathcal{D}) = \frac{P(\mathbf{Y}|\mathbf{f}, \mathbf{u}, \mathbf{W}) p(\mathbf{f}, \mathbf{u}, \mathbf{W})}{\int\int\int P(\mathbf{Y}|\mathbf{f}, \mathbf{u}, \mathbf{W}) p(\mathbf{f}, \mathbf{u}, \mathbf{W}) \mathrm{d}\mathbf{f}\mathrm{d}\mathbf{u}\mathrm{d}\mathbf{W}}.$$

The non-Gaussian likelihood $P(\mathbf{Y}|\mathbf{f}, \mathbf{u}, \mathbf{W})$ makes this posterior analytically intractable. We thus resort to variational inference [Jordan et al., 1999] in order to get an approximate posterior. Recall from Section 2.1.2 that variational inference methods entail considering a tractable family of distributions and finding the member of this family that is closest to the true posterior. This is done by minimizing the Kullback-Leiber (KL) divergence between the joint approximated posterior and the true joint posterior which is equivalent to maximizing the so-called evidence lower bound, $\mathcal{L}_{\text{elbo}}$.

4.2.1 Variational Distributions

We consider a mean-field approximation scheme and assume a fully factorized variational distribution defined as:

$$q(\mathbf{f}, \mathbf{u}, \mathbf{W}|\boldsymbol{\nu}) = p(\mathbf{f}|\mathbf{u}) \prod_{q=1}^{Q} \underbrace{\mathcal{N}(\mathbf{m}_q, \mathbf{S}_q)}_{q(\mathbf{u}_{\bullet q}|\boldsymbol{\nu}_{u_q})} \prod_{q=1}^{Q} \underbrace{\mathcal{N}(\boldsymbol{\omega}_q, \boldsymbol{\Omega}_q)}_{q(\mathbf{W}_{\bullet q}|\boldsymbol{\nu}_{w_q})} \tag{4.2}$$

where $\boldsymbol{\nu}_u = \{\mathbf{m}_q, \mathbf{S}_q\}$ and $\boldsymbol{\nu}_w = \{\boldsymbol{\omega}_q, \boldsymbol{\Omega}_q\}$ are the variational parameters. The choice for this variational distribution, in particular with regards to the incorporation of the conditional prior $p(\mathbf{f}|\mathbf{u})$, is motivated by the work of Titsias [2009b], and will yield a convenient decomposition of the evidence lower bound in terms of computational cost. In addition, it will allow scalability to very large datasets through stochastic optimization. When considering an uncorrelated prior over the weights, we assume an uncorrelated posterior by forcing $\boldsymbol{\Omega}_q$ to be diagonal. Eq. (4.2) fully defines our approximate posterior. With this, we give details of the variational objective function, i.e. $\mathcal{L}_{\text{elbo}}$, we aim to maximize with respect to $\boldsymbol{\nu} = \{\boldsymbol{\nu}_u, \boldsymbol{\nu}_w\}$.

4.2.2 Evidence Lower Bound

Following standard variational inference arguments (see Section 2.1.2), it is straightforward to show that the evidence lower bound decomposes as the sum of a KL-divergence term ($\mathcal{L}_{\text{kl}}$) between the approximate posterior and the prior, and an expected log-likelihood term ($\mathcal{L}_{\text{ell}}$), where the expectation is taken with respect to the approximate posterior. We can write:

$$\mathcal{L}_{\text{elbo}}(\boldsymbol{\nu}) = \mathcal{L}_{\text{kl}}(\boldsymbol{\nu}) + \mathcal{L}_{\text{ell}}(\boldsymbol{\nu}) \tag{4.3}$$

$$\mathcal{L}_{\text{kl}}(\boldsymbol{\nu}) = -\text{KL}(q(\mathbf{f}, \mathbf{u}, \mathbf{W}|\boldsymbol{\nu}) \| p(\mathbf{f}, \mathbf{u}, \mathbf{W})) \tag{4.4}$$

$$\mathcal{L}_{\text{ell}}(\boldsymbol{\nu}) = \mathbb{E}_{q(\mathbf{f}, \mathbf{u}, \mathbf{W}|\boldsymbol{\nu})}[\log p(\mathbf{Y}|\mathbf{f}, \mathbf{W})]. \tag{4.5}$$

We will now show how these terms can be computed given the assumed variational distribution in Eq. (4.2).

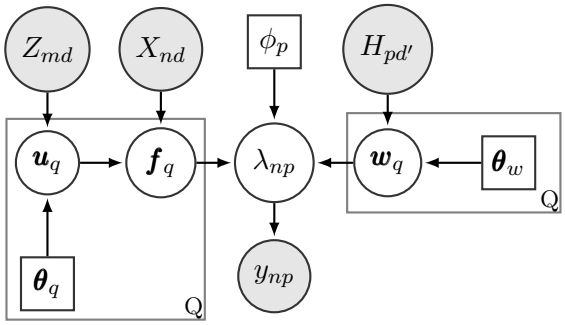


Figure 4.2: Graphical model representation of MCPM with GP prior on $\mathbf{W}$ and tasks' descriptors $H_{pd'}$. Square nodes denote optimised deterministic variables.

KL-divergence Term The variational distribution given in Eq. (4.2) significantly simplifies the computation of $\mathcal{L}_{\text{kl}}$ (Eq. (4.4)), where the terms containing the latent function values $\mathbf{f}$ vanish. We can write:

$$
\begin{aligned}
\mathcal{L}_{\text{kl}}(\boldsymbol{\nu}) &= -\text{KL}(q(\mathbf{f},\mathbf{u},\mathbf{W}|\boldsymbol{\nu})\|p(\mathbf{f},\mathbf{u},\mathbf{W})) \\
&= -\int\int\int q(\mathbf{f},\mathbf{u},\mathbf{W}|\boldsymbol{\nu}) \log\left(\frac{q(\mathbf{f},\mathbf{u},\mathbf{W}|\boldsymbol{\nu})}{p(\mathbf{f},\mathbf{u},\mathbf{W})}\right) \text{d}\mathbf{f}\text{d}\mathbf{u}\text{d}\mathbf{W} \\
&= -\int\int\int p(\mathbf{f}|\mathbf{u},\mathbf{W},\boldsymbol{\nu}) q(\mathbf{u},\mathbf{W}|\boldsymbol{\nu}) \log\left(\frac{\cancel{p(\mathbf{f}|\mathbf{u},\mathbf{W})}q(\mathbf{u},\mathbf{W}|\boldsymbol{\nu})}{\cancel{p(\mathbf{f}|\mathbf{u},\mathbf{W})}p(\mathbf{u},\mathbf{W})}\right) \text{d}\mathbf{f}\text{d}\mathbf{u}\text{d}\mathbf{W} \\
&= -\int\int q(\mathbf{u},\mathbf{W}|\boldsymbol{\nu}) \log\left(\frac{q(\mathbf{u},\mathbf{W}|\boldsymbol{\nu})}{p(\mathbf{u},\mathbf{W})}\right) \text{d}\mathbf{u}\text{d}\mathbf{W} \\
&= -\int\int q(\mathbf{u}|\boldsymbol{\nu}_u) q(\mathbf{W}|\boldsymbol{\nu}_w) \log\left(\frac{q(\mathbf{u}|\boldsymbol{\nu}_u) q(\mathbf{W}|\boldsymbol{\nu}_w)}{p(\mathbf{u},\mathbf{W})}\right) \text{d}\mathbf{u}\text{d}\mathbf{W}. \qquad (4.6)
\end{aligned}
$$

We can further expand Eq. (4.6) to write the KL term as $\mathcal{L}_{\text{kl}}(\boldsymbol{\nu}) = \mathcal{L}^u_{\text{ent}}(\boldsymbol{\nu}_u) + \mathcal{L}^u_{\text{cross}}(\boldsymbol{\nu}_u) + \mathcal{L}^w_{\text{ent}}(\boldsymbol{\nu}_w) + \mathcal{L}^w_{\text{cross}}(\boldsymbol{\nu}_w)$. We have:

$$
\begin{aligned}
\mathcal{L}_{\text{kl}}(\boldsymbol{\nu}) &= -\int\int q(\mathbf{u}|\boldsymbol{\nu}_u) q(\mathbf{W}|\boldsymbol{\nu}_w) \log\left(\frac{q(\mathbf{u}|\boldsymbol{\nu}_u) q(\mathbf{W}|\boldsymbol{\nu}_w)}{p(\mathbf{u},\mathbf{W})}\right) \text{d}\mathbf{u}\text{d}\mathbf{W} \\
&= -\int\int q(\mathbf{u}|\boldsymbol{\nu}_u) q(\mathbf{W}|\boldsymbol{\nu}_w) \left[\log q(\mathbf{u}|\boldsymbol{\nu}_u) q(\mathbf{W}|\boldsymbol{\nu}_w) - \log p(\mathbf{u}) p(\mathbf{W})\right] \text{d}\mathbf{u}\text{d}\mathbf{W} \\
&= -\int q(\mathbf{u}|\boldsymbol{\nu}_u) \log q(\mathbf{u}|\boldsymbol{\nu}_u) \text{d}\mathbf{u} - \int q(\mathbf{W}|\boldsymbol{\nu}_w) \log q(\mathbf{W}|\boldsymbol{\nu}_w) \text{d}\mathbf{W} \\
&\quad + \int q(\mathbf{u}|\boldsymbol{\nu}_u) \log p(\mathbf{u}) \text{d}\mathbf{u} + \int q(\mathbf{W}|\boldsymbol{\nu}_w) \log p(\mathbf{W}) \text{d}\mathbf{W}
\end{aligned}
$$

The first two terms represent the entropy terms for both $\mathbf{u}$ and $\mathbf{W}$ respectively. The last two terms give the negative cross-entropies between the prior distributions and the variational distributions. Full derivations for all these terms are given in Appendix A.1. Here we report the final expressions:

$$
\mathcal{L}^u_{\text{ent}}(\boldsymbol{\nu}_u) = \frac{1}{2}\sum_{q=1}^{Q} \left[M \log 2\pi + \log|\mathbf{S}_q| + M\right],
$$

$$\mathcal{L}^u_{\text{cross}}(\boldsymbol{\nu}_u) = \sum_{q=1}^{Q} \left[\log \mathcal{N}(\mathbf{m}_q; \mathbf{0}, \mathbf{K}^q_{zz}) - \frac{1}{2} \text{ tr } (\mathbf{K}^q_{zz})^{-1} \mathbf{S}_q \right],$$

$$\mathcal{L}^w_{\text{ent}}(\boldsymbol{\nu}_w) = \frac{1}{2} \sum_{q=1}^{Q} [\ P \log 2\pi + \log |\boldsymbol{\Omega}_q| + P],$$

$$\mathcal{L}^w_{\text{cross}}(\boldsymbol{\nu}_w) = \sum_{q=1}^{Q} \left[\log \mathcal{N}(\boldsymbol{\omega}_q; \mathbf{0}, \mathbf{K}^q_w) - \frac{1}{2} \text{ tr } (\mathbf{K}^q_w)^{-1} \boldsymbol{\Omega}_q \right].$$

When placing an independent prior and approximate posterior over $\mathbf{W}$, the terms $\mathcal{L}^w_{\text{ent}}$ and $\mathcal{L}^w_{\text{cross}}$ get simplified further, significantly reducing the computational cost when P is large, see Eq. (A.5) and Eq. (A.6) in Appendix A.1.

4.2.3 Moment generating function of log intensities

The MCPM formulation allows to derive a closed form expression for the moments of the intensity function. The first moment of $\exp(\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet})$ is particularly important as it can be used to evaluate $\mathcal{L}_{\text{ell}}$ in closed form thus avoiding additional Monte Carlo approximations in the evaluation of the ELBO. The t-th moment for the p-th task intensity evaluated at $\mathbf{x}_n$, namely $\mathbb{E}\left[\lambda_p(\mathbf{x}_n)^t\right]$, can be written as $\exp(t\phi_p)\mathbb{E}\left[\exp\left(t\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet}\right)\right] = \exp(t\phi_p)\text{MGF}_{\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet}}(t)$ where $\text{MGF}_{\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet}}(t)$ denotes the moment generating function of $\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet}$ in t. The random variable $\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet}$ is the sum of products of independent Gaussians [Craig, 1936] and its MGF is thus given by:

$$\text{MGF}_{\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet}}(t) = \prod_{q=1}^{Q} \frac{\exp\left[\frac{t\gamma_{pq}\tilde{\mu}_{nq} + \frac{1}{2}(\tilde{\mu}^2_{nq}K^{qp}_w + \gamma^2_{pq}\widetilde{K}^q(n))t^2}{1 - t^2 K^{qp}_w \widetilde{K}^q(n)}\right]}{\sqrt{1 - t^2 K^{qp}_w \widetilde{K}^q(n)}}, \tag{4.7}$$

where the expectation is computed with respect to the prior distribution of $\mathbf{W}_{p\bullet}$ and $\mathbf{f}_{n\bullet}$; γ_{pq} is the prior mean of w_{pq}; $\widetilde{K}^q(n)$ denotes the variance of f_{nq}; and K^{qp}_w is the variance of w_{pq}. Details about the derivations of Eq. (4.7) are given in Appendix A.2.

4.2.4 Closed-form Expected Log-Likelihood Term

Despite the additional model complexity introduced by the stochastic nature of the mixing weights, the expected log-likelihood term $\mathcal{L}_{\text{ell}}$ (Eq. (4.5)), can be evaluated in closed form by exploiting the moment generating function

introduced in Eq. (4.7). Specifically, we have:

$$\mathcal{L}_{\text{ell}}(\boldsymbol{\nu}) = -\sum_{n=1}^{N}\sum_{p=1}^{P}\exp(\phi_p)\underbrace{\mathbb{E}_{q(\mathbf{f}_{n\bullet})q(\mathbf{W}_{p\bullet})}\left(\exp\left(\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet}\right)\right)}_{\text{MGF}_{\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet}}(1)} + \sum_{n=1}^{N}\sum_{p=1}^{P}\sum_{q=1}^{Q}\left[y_{np}(\omega_{pq}\mu_{nq}+\phi_p)-\log(y_{np}!)\right]$$

where $\mu_{nq} = \mu_q(x^{(n)}) = A_q\mathbf{m}_q(x^{(n)})$. See Appendix A.2 for the full derivation of $\mathcal{L}_{\text{ell}}$. The term $\text{MGF}_{\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet}}(1)$ is computed evaluating Eq. (4.7) at $t = 1$ given the current variational parameters for $q(\mathbf{W})$ and $q(\mathbf{f})$. A closed-form expected log-likelihood term significantly speeds up the algorithm achieving similar performance but 2 times faster than a Monte Carlo approximation on the CRIME dataset (Section 4.4.2, see Fig. A.1 and Fig. A.2 in Appendix A.4). In addition, by providing an analytical expression for $\mathcal{L}_{\text{ell}}$, we avoid high-variance gradient estimates which are often an issue in modern variational inference methods relying on Monte Carlo estimates.

Algorithm complexity and implementation The time complexity of our algorithm is dominated by algebraic operations on $\mathbf{K}_{zz}^{q}$, which are $\mathcal{O}(M^3)$, while the space complexity is dominated by storing $\mathbf{K}_{zz}^{q}$, which is $\mathcal{O}(M^2)$ where M denotes the number of inducing variables per latent process. $\mathcal{L}_{\text{ent}}$ and $\mathcal{L}_{\text{cross}}$ only depend on distributions over M dimensional variables thus their computational complexity is independent of N. In addition, notice how $\mathcal{L}_{\text{ell}}$ decomposes as a sum of expectations over individual data points thus stochastic optimization techniques can be used to evaluate this term making it independent of N. Finally, the algorithm complexity does not depend on the number of tasks P but only on the number of latent processes Q thus making it scalable to large multi-task datasets. We provide an implementation of the algorithm that uses Tensorflow [Abadi et al., 2016]. Pseudocode is given in Algorithm 1.

4.3 Related work

The approach presented in this chapter relates to other works on (i) multi-task regression, (ii) models with black-box likelihoods and GP priors, (iii) and other GP-modulated Poisson processes. In this section, we discuss the studies more closely related within each group.

Multi-task regression A large proportion of the literature on multi-task learning methods [Caruana, 1998] with GPs has focused on regression problems [Álvarez and Lawrence, 2011; Alvarez et al., 2010; Bonilla et al., 2007; Gal et al.,

2014a; Teh et al., 2005b; Wilson et al., 2011a], perhaps due to the additional challenges posed by complex non-linear non-Gaussian likelihood models. Some of these methods are reviewed in Álvarez et al. [2012]. Of particular interest to this paper is the linear coregionalization model (LCM) of which the ICM is a particular instance. It can be shown that the MCPM prior covariance generalizes the LCM prior. In addition, the two methods differ substantially in terms of inference, model flexibility, and accuracy, see Appendix A.3 for a discussion on the links between these approaches. Unlike standard coregionalization methods, Schmidt and Gelfand [2003] consider a prior over the mixing weights but, unlike our method, their focus is on regression problems and they carry out posterior estimation via a costly MCMC procedure.

Black-box likelihood methods Modern advances in variational inference have allowed the development of generic methods for inference in models with GP priors and 'black-box' likelihoods including LGCPs [Dezfouli and Bonilla, 2015; Hensman et al., 2015; Matthews et al., 2017]. While these frameworks offer the opportunity to prototype new models quickly, they can only handle deterministic weights and are inefficient. In contrast, we exploit our likelihood characteristics and derive closed-form MGF expressions for the evaluation of the ELL term. By adjusting the ELBO to include the entropy and cross-entropy terms arising from the stochastic weights and using the closed-form MGF, we significantly improve the algorithm convergence and efficiency.

Algorithm 1 MCPM

Inputs: Dataset $\mathcal{D} = \{\mathbf{x}_{n_p} \in \tau, n_p = 1, \ldots, N_p, \forall p = 1, ..., P\}$ for bounded region τ where N_p denotes the number of events for the p-th task. Number of latent GPs Q. Number of mini-batches `b` of size B. Learning rate ρ.

Output: Optimized hyper-parameters, posterior moments of $\boldsymbol{\lambda}$

Discretize event locations $\mathcal{D}$ in $Y \in \mathbb{R}^{N \times P}$ given the grid size.

Initialize: $i \leftarrow 0$, $\boldsymbol{\eta}^{(0)} = (\boldsymbol{\theta}_f^q, \boldsymbol{\theta}_w^q, \boldsymbol{\phi}, \boldsymbol{\nu}_u, \boldsymbol{\nu}_w)$

repeat

  $\{X_{train} \in \mathbb{R}^{B \times D}, Y_{train} \in \mathbb{R}^{B \times P}\} \rightarrow$ `get-next-MiniBatch`$(\mathcal{D})$

  for `j=0 to b` **do**

    $\max_{\boldsymbol{\mu}} \mathcal{L}_{\text{elbo}}(\boldsymbol{\eta}^{(i)})$ (Eq. (4.3))

    $\boldsymbol{\eta}^{(i)} \leftarrow \boldsymbol{\eta}^{(i-1)} - \rho \nabla_{\boldsymbol{\eta}} \mathcal{L}_{\text{elbo}}(\boldsymbol{\eta}^{(i-1)})$

    $i = i + 1$

  end for

until convergence criterion is met.

$\boldsymbol{\eta}^* \leftarrow \boldsymbol{\eta}^{(i-1)}$

$\mathbb{E}\left[\boldsymbol{\lambda}(\mathbf{x})^t\right] = \exp(t\boldsymbol{\phi}^*)\text{MGF}_{\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet}|\boldsymbol{\eta}^*}(t)$

Aside from variational schemes, the integrated nested Laplace approximation (INLA) method [Rue et al., 2009] is a computational less-intensive alternative to MCMC that allows us to perform approximate Bayesian inference in latent GP models. While it has been shown to perform well for various Poisson point process models [Illian et al., 2012a, 2013, 2012b; Rue et al., 2009; Simpson et al., 2016b], INLA uses numerical integration to approximate the marginal likelihood, which makes it unsuitable for GP models that contain a large number of hyperparameters. As this is generally the case in multi-task models, in this work we use variational inference. We are not aware of any work using INLA for multi-task GP modulated Poisson point process thus we identify this direction as an interesting open problem.

Other GP-modulated Poisson processes Rather than using a GP prior over the log intensity, different transformations of the latent GPs have been considered as alternatives to model point data. For example, in the Permanental process, a GP prior is used over the squared root of the intensities [John and Hensman, 2018; Lian et al., 2015; Lloyd et al., 2015; Lloyd et al., 2016; Walder and Bishop, 2017]. Similarly, a sigmoidal transformation of the latent GPs was studied by Adams et al. [2009] and used in conjunction with convolution processes by Gunter et al. [2014]. Permanental and Sigmoidal Cox processes are very different from LGCP/MCPM both in terms of statistical and computational properties. There is no conclusive evidence in the literature on which model provides a better characterisation of point processes and under what conditions. The MCPM likelihood introduces computational issues in terms of ELL evaluation which in this work are solved by offering a closed-form MGF function. On the contrary, Permanental processes suffer from important identifiability issues such as reflection invariance and, together with sigmoidal processes, do not allow for a closed-form prediction of the intensity function. In order to avoid the computational issues introduced by different link functions, Samo and Roberts [2015] place an appropriate finite-dimensional prior on the values of the intensity function computed at the inducing points. This enables the development of an MCMC scheme carachterized by a time and memory requirement that is linear in the data size. Among the inhomogeneous Cox process models, the only two multi-task frameworks are Gunter et al. [2014] and Lian et al. [MTPP, 2015]. The former suffers from high computational cost due to the use of expensive MCMC schemes scaling with $\mathcal{O}(PN^3)$. In addition, while the framework is developed for an arbitrary number of latent functions, a single latent function is used in all of the presented experiments. MTPP restricts the input space to be unidimensional and does not handle missing data. Furthermore, none of these two methods can handle spatial segregation (Section 4.4.3) through a shared global mean function or a single latent function.

Table 4.1: Performance on the missing intervals. MCPM-N and MCPM-GP denote independent and correlated prior respectively. Lower values of RMSE and NLPL are better. CPU time is given in seconds per epoch.

	RMSE				NLPL				CPU
	1	**2**	**3**	**4**	**1**	**2**	**3**	**4**	time
MCPM-N	38.61	7.86	5.71	**4.68**	20.99	3.75	**3.31**	**3.02**	**0.18**
MCPM-GP	**38.58**	**7.69**	**5.70**	4.71	**20.95**	**3.70**	**3.31**	3.03	0.25
LGCP	48.17	14.32	11.83	5.38	43.40	8.78	8.98	3.27	0.32
ICM	39.07	7.96	7.88	6.03	21.81	3.76	3.77	3.38	0.52

4.4 Experiments

We assess MCPM performance in a variety of settings. First of all, we analyse MCPM on two synthetic datasets. In the first one, we illustrate the transfer capabilities in a missing data setting comparing against LGCP and ICM. In the second dataset, we assess the predictive performance against the MTPP model which cannot handle missing data. We then proceed to model two real-world datasets that exhibit very different correlation structures. The first one includes spatially segregated tasks while the second one is characterized by a strong positive correlation between tasks. Code and data for all the experiments are provided at `https://github.com/VirgiAgl/MCPM`.

Baselines We offer results on both complete and incomplete data settings while comparing against the MLGCP framework proposed by Taylor et al. [2015], a *variational* LGCP model [Nguyen and Bonilla, 2014] and a *variational* formulation of ICM with Poisson likelihood implemented in GPflow [Hensman et al., 2015; Matthews et al., 2017].

Performance measures We compare the algorithms evaluating the root mean square error (RMSE), the negative log predicted likelihood (NLPL) and the empirical coverage (EC) of the posterior predictive counts distribution. RMSE and NLPL values for the p-th task are computed as:

$$\text{RMSE}_p = \sqrt{\frac{1}{N}\sum_{n=1}^{N}\left(y_{np} - \mathbb{E}(\lambda_{np})\right)^2},$$

$$\text{NLPL}_p = -\frac{1}{S}\sum_{s=1}^{S}\frac{\sum_{n=1}^{N}\log p(y_{np}|\lambda_{np}^s)}{n}$$

where $\mathbb{E}(\lambda_{np})$ represents the posterior mean estimate for the p-th intensity at $\mathbf{x}_n$ and S denotes the number of samples from the variational distributions $q(\mathbf{f})$ and

Table 4.2: S2 dataset. Performance on the test intervals. MCPM-N and MCPM-GP denote independent and correlated prior respectively. Lower values of NLPL are better. CPU time is given in seconds per epoch.

	NLPL										CPU time
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	
MCPM-N	**1.47**	**1.46**	**0.95**	0.17	1.30	1.39	1.52	0.70	1.58	0.58	**0.03**
MCPM-GP	1.52	1.80	0.96	**0.13**	1.29	**1.37**	1.61	**0.65**	**1.50**	0.76	**0.03**
MTPP	1.60	3.05	1.13	0.15	**1.24**	1.44	**1.49**	1.13	1.70	**0.52**	5.97

$q(\mathbf{W})$. The EC is constructed by drawing random subregions from the training (to compute in-sample performance) or the test set (to compute out-of-sample performance) and evaluating the coverage of the 90% credible interval (CI) of the posterior ($p(N|\mathcal{D})$) and predictive ($p(N^*|\mathcal{D})$) counts distribution for each subregion B (this metric was previously used by Leininger et al. [2017]). These are in turn obtained by simulating from $N^{(l)}(B) \sim \text{Poisson}(\lambda^{(l)}(B))$ for $l = 1, ..., L$ with $\lambda^{(l)}(B)$ denoting the l-th sample from the intensity posterior and predictive distribution. The presented results consider $L = 100$ samples but consistent results were found when changing this value. In terms of subregions selection, we fix their size, say Z, and randomly select L of them among all the possible areas of size Z in the training or test set. The empirical coverage is equal to one when all CIs contain the ground truth. Finally, in order to assess transfer in the 2D experiments, we partition the spatial extent in Z subregions and create missing data "folds" by combining non-overlapping regions, one for each task. We repeat the experiment Z times until each task's overall spatial extend is covered thus accounting for areas of both high and low intensity.

4.4.1 Synthetic experiments

Synthetic missing data experiment (S1) To illustrate the *transfer* capabilities of MCPM we construct four correlated tasks by sampling from a multivariate point process with final intensities obtained as the linear combination of two latent functions via task-specific mixing weights (Fig. 4.3). The final count observations are obtained by adding noise to the Poisson counts generated through the constructed intensities. When using a coupled prior over the weights, we consider covariates describing tasks (e.g. minimum and maximum values) as inputs. MCPM is able to reconstruct the task intensities in the missing data regions by learning the inter-task correlation and transferring information across tasks. Importantly, it significantly outperforms competing approaches for all tasks in terms of EC (see Appendix A.5) and NLPL. In addition, it has a lower RMSE for $\frac{3}{4}$ of the tasks (Table 4.1) while being ≈ 3 times faster than ICM.

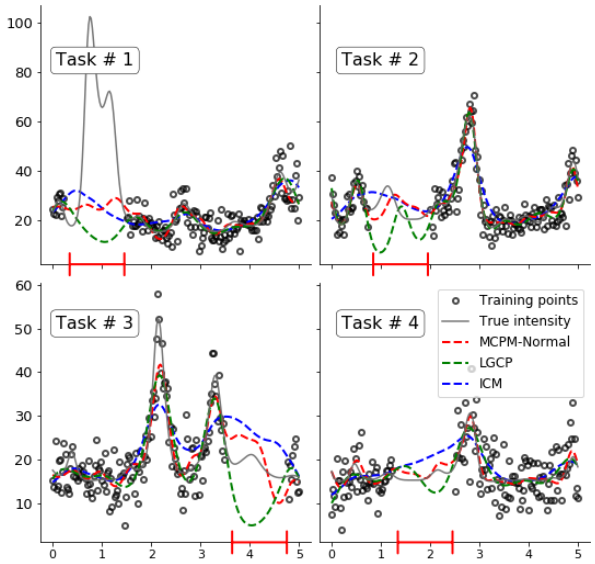


Figure 4.3: Four related tasks evaluated at 200 evenly spaced points in the interval $[0, 5]$. Empty black dots give the observed counts used as training data and sampled from the true underlying intensities (grey lines). The red annotations on the x-axis denote the missing data regions which include 50 contiguous observations removed from the training set of each task.

Synthetic comparison to MTPP (S2) To assess the *predictive* capabilities of MCPM against MTPP, which cannot handle missing data, we replicate the synthetic example proposed by Lian et al. [2015] (Section 6.1). We train the models with the observations in the interval $[0, 80]$ and predict in the interval $[80, 100]$. We then construct the predictive counts distribution for both models by sampling from the posterior intensity distributions. Fig. 4.4 shows how MCPM better recovers the true model event counts distribution with respect to MTPP. We found MCPM to outperform MTPP in terms of NLPL, EC and RMSE for $\frac{7}{10}$ tasks, see Table 4.2, Fig. 4.4 and Appendix A.5.

4.4.2 Crime events in NYC

In this section, we demonstrate the performance, transfer capabilities, and scalability of MCPM on a real-world dataset recording seven different types of crime in NYC. We refer to this dataset with the acronym CRIME. CRIME includes latitude and longitude locations of burglaries (1), felony assaults (2), grand larcenies (3), grand larcenies of motor vehicle (MV, 4), petit larcenies (5), petit larcenies of MV (6) and robberies (7) reported in 2016. Crimes location data are taken from the NYC police department website[1] and discretized into a 32×32 regular grid (see first row of Fig. 4.5). Lack of ground truth intensities for real-data settings typically restricts quantitative measures of generalization. Here we focus on validating and comparing MCPM from two different perspectives: i)

[1] `https://www1.nyc.gov/site/nypd/stats/crime-statistics`.

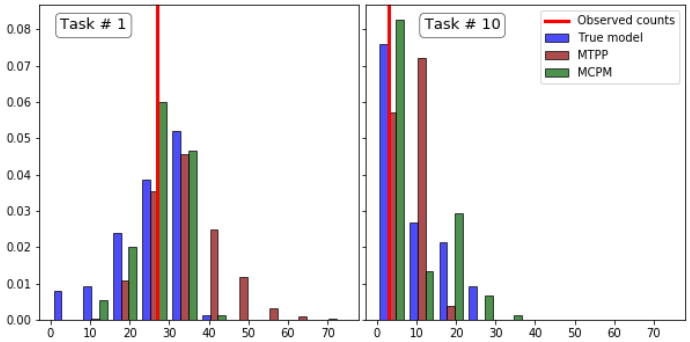


Figure 4.4: S2 dataset. Predicted empirical distribution of event counts for two tasks obtained by sampling from the posterior intensity distributions.

Table 4.3: CRIME dataset. NLPL performance on the missing regions. CPU time is given in seconds per epoch. Lower values of NLPL are better. MCPM-N and MCPM-GP denote independent and correlated prior respectively.

	Standardized NLPL (per cell)							CPU time
	1	**2**	**3**	**4**	**5**	**6**	**7**	
MCPM-N	**0.56** (0.10)	0.91 (0.27)	**0.66** (0.30)	**1.09** (0.27)	0.85 (0.52)	**10.29** (2.51)	**0.42** (0.05)	**2.85**
MCPM-GP	0.72 (0.18)	**0.75** (0.18)	0.94 (0.55)	1.53 (0.52)	**0.57** (0.19)	18.76 (8.25)	0.58 (0.12)	3.11
LGCP	9.90 (3.66)	9.32 (2.41)	19.34 (11.45)	5.30 (1.02)	18.18 (8.65)	36.73 (4.02)	9.68 (2.67)	2.87
ICM	0.87 (0.27)	1.36 (0.35)	0.91 (0.45)	1.19 (0.40)	0.69 (0.11)	12.30 (3.02)	0.93 (0.17)	44.13

using complete data so as to assess the quality of the recovered intensities as well as the *computational complexity* and scalability gains over MLGCP; and ii) using missing data so as to validate the *transfer* capabilities of MCPM when compared with LGCP and ICM.

Complete Data Experiment We first consider a full-data experiment and we spatially interpolate the crime surfaces running MCPM with $Q = 4$ latent functions characterized by Matérn 3/2 kernels. We repeat the experiment with MLGCP setting the algorithm parameters as suggested by Taylor et al. [2015]. Similar results are obtained with the two methods, see Fig. A.4 in Appendix A.5 for a visualisation of the estimated intensity surfaces. However, MCPM achieves significant *computational gains* with respect to MLGCP. A MLGCP run takes $\approx$ 14 hrs while MCPM requires $\approx$ 2 hrs.

Missing Data Experiment To assess *transfer*, we keep the same experimental settings and introduce missing data regions by partitioning the spatial extent

Table 4.4: CRIME dataset. In-sample/Out-of-sample 90% CI coverage for the predicted event counts distributions. Higher values of EC are better. MCPM-N and MCPM-GP denote independent and correlated prior respectively.

	Empirical Coverage (EC)						
	1	**2**	**3**	**4**	**5**	**6**	**7**
MCPM-N	0.99/0.80	**1.00**/0.73	0.97/**0.71**	**1.00**/0.73	0.98/0.61	**1.00**/**1.00**	0.99/**0.87**
MCPM-GP	**1.00**/**0.87**	**1.00**/**0.74**	**1.00**/**0.71**	**1.00**/**0.95**	**1.00**/**0.88**	0.80/**1.00**	**1.00**/0.85
LGCP	0.86/0.29	0.76/0.20	0.86/0.29	0.82/0.37	0.68/0.25	0.94/0.00	0.83/0.21
ICM	0.68/0.73	0.75/0.50	0.64/0.52	0.79/0.65	0.59/0.78	0.93/0.86	0.841/0.64

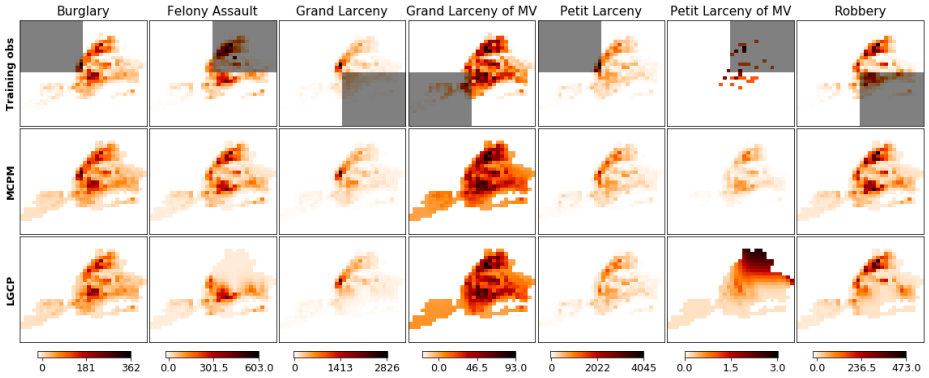


Figure 4.5: CRIME dataset. *First row:* Observed counts for seven different types of crimes on a 32×32 regular grid. The shaded regions represent one possible configuration of the missing data folds across the seven tasks. *Second row:* MCPM estimated intensities when introducing missing data. *Third row:* LGCP estimated intensities when introducing missing data.

in 4 subregions as explained above. The shaded regions in Fig. 4.5 represent one possible configuration of the missing data folds across tasks. Fig. 4.5 shows how MCPM successfully transfers information across tasks thereby recovering, for all crime types, the signal in the missing data regions. By exploiting task similarities, the algorithm outperforms competing approaches in all of the tasks, in terms of EC (Table 4.4), NLPL (Table 4.3) and RMSE (Table A.4 in Appendix A.5). Finally, MCPM significantly outruns ICM in terms of algorithm efficiency. MCPM-N converges in 1.19 hrs (1500 epochs) on a Intel Core i7-6t00U CPU (3.40GHz, 8GB of RAM) while ICM needs 12.26 hrs (1000 epochs).

4.4.3 Bovine Tuberculosis (BTB) in Cornwall

We showcase the performance of MCPM on the BTB dataset [Diggle et al., 2013; Taylor et al., 2015] consisting of locations of BTB incidents in Cornwall, UK (period 1989–2002) and covariates measuring cattle density, see first row in Fig. 4.6. We follow Diggle et al. [2013] and only consider the four most common

Table 4.5: BTB dataset. MCPM-N and MCPM-GP denote independent and correlated prior respectively. RMSE and NLPL with missing data. CPU time in given in seconds per epoch. Lower values of NLPL are better.

	RMSE				NLPL (per cell)				CPU time
	GT 9	**GT 12**	**GT 15**	**GT 20**	**GT 9**	**GT 12**	**GT 15**	**GT 20**	
MCPM-N	0.83 (0.15)	0.24 (0.07)	0.28 (0.07)	0.29 (0.10)	**1.23** (0.40)	0.20 (0.07)	**0.33** (0.11)	**0.35** (0.16)	7.73
MCPM-GP	**0.81** (0.14)	0.22 (0.08)	**0.27** (0.07)	**0.27** (0.09)	1.42 (0.42)	0.27 (0.09)	0.41 (0.14)	0.58 (0.24)	**7.63**
LGCP	1.37 (0.33)	0.61 (0.13)	0.63 (0.12)	1.24 (0.56)	1.70 (0.39)	0.48 (0.11)	0.72 (0.17)	0.86 (0.36)	8.76
ICM	0.91 (0.15)	**0.21** (0.07)	0.32 (0.08)	7.24 (5.48)	1.44 (0.40)	**0.18** (0.06)	0.34 (0.10)	0.37 (0.14)	67.06

Table 4.6: BTB dataset. MCPM-N and MCPM-GP denote independent and correlated prior respectively. In-sample/Out-of-sample 90% CI coverage for the predicted event counts distributions. Higher values of EC are better.

	Empirical Coverage (EC)			
	GT 9	**GT 12**	**GT 15**	**GT 20**
MCPM-N	0.87/**0.92**	0.97/**0.99**	0.93/0.96	0.95/**1.00**
MCPM-GP	**0.93**/0.91	**0.98**/0.98	**0.97**/**0.98**	**0.97**/0.99
LGCP	0.91/0.79	0.97/0.98	**0.97**/0.97	0.96/0.98
ICM	0.90/0.84	0.96/0.98	0.95/0.96	0.96/0.96

BTB genotypes (GT: 9, 12, 15 and 20).

Complete Data Experiment We estimate the four BTB intensities by fitting an MCPM with $Q = 4$ latent functions and Matérn 3/2 kernels. We initialise the kernel lenghtscales and variances to 1. For direct comparison, we train the MLGCP model following the grid size, prior, covariance and MCMC settings specified by Taylor et al. [2015]. We run the MCMC chain for 1 million iterations with a burn in of 100K and keep 1K thinned steps. Following Diggle et al. [2013], in Fig. 4.6 we report the probability surfaces computed as $\pi_p(x) = \lambda_p(x)/\sum_{p=1}^{P}\lambda_p(x)$ where $\lambda_p(x)$ is the posterior mean of the intensity for task p at location x. Estimated intensities surfaces can be found in the supplementary material (Appendix A.5). The probability surfaces are comparable with both approaches characterizing well the high and low intensities albeit varying at the level of smoothness. In terms of *computational gains*, we note that MLGCP takes $\approx$ 30 hours for an interpolation run on the four BTB tasks while MCPM only requires $\approx$ 8 hrs. The previously reported [Diggle et al., 2013] slow mixing

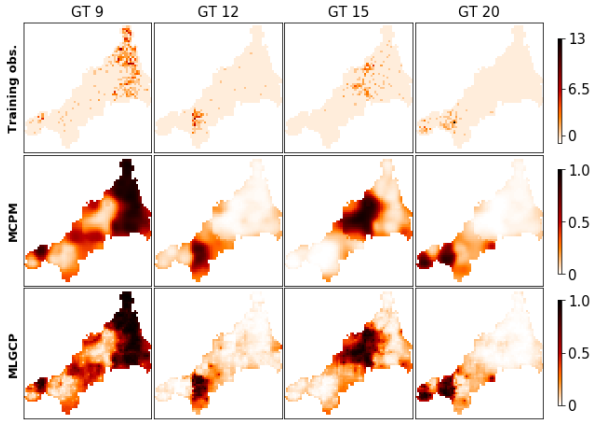


Figure 4.6: BTB dataset. *First row:* Observed counts for the four different BTB genotypes on a 64×64 regular grid. *Second row:* MCPM estimated conditional probabilities for the complete data setting. *Third row:* MLGCP estimated conditional probabilities for the complete data setting. For moth methods the estimated intensity surfaces are given in Appendix A.5.

and convergence problems of the chain, even after millions of MCMC iterations, renders MLGCP problematic for application to large-scale multivariate point processes. Finally, the built-in assumption of a single common GP latent process across tasks limits the number and the type of inter-task correlations that we can identify and model efficiently.

Missing Data Experiment *Transfer* is evaluated by partitioning the space into 16 subregions and constructing missing data regions as explained above. The shaded regions in the first row of Fig. 4.7 represent one such fold of the missing areas across tasks. We provide average quantitative metrics across folds for an MCPM with four latent functions, Matérn 3/2 kernels and 30% of the training inputs as inducing inputs. As in the complete data setting, we report estimated conditional probabilities in Fig. 4.7 and give additional results in Appendix A.5. MCPM manages to recover the overall behaviour of the process in the missing regions showing significant transfer of information across spatially segregated tasks while avoiding negative transfer in the case of negative spatial correlation. MCPM outperforms LGCP across all tasks and achieves better performance than ICM on $\frac{3}{4}$ of the tasks (see Table 4.5 and Table 4.6). In addition MCPM exhibits the highest EC both in-sample and out-of-sample. Finally, MCPM has a significant computational advantage: it converges in 3.18 hours (1500 epochs) while ICM converges in 18.63 hrs (1000 epochs).

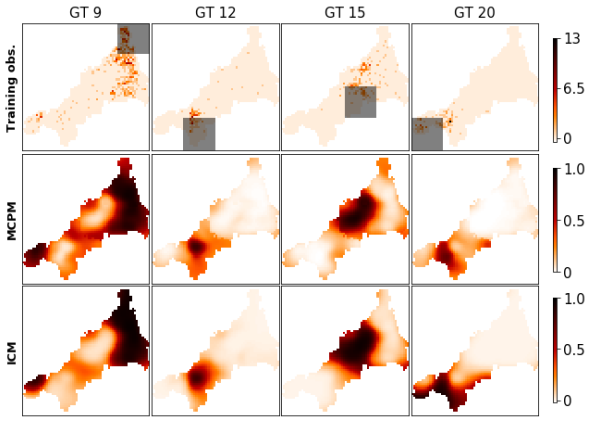


Figure 4.7: BTB dataset. *First row:* Observed counts for the four different BTB genotypes on a 64×64 regular grid. The shaded areas represent one possible configuration of the missing data folds across the four tasks. *Second row:* MCPM estimated conditional probabilities for the missing data setting. *Third row:* MLGCP estimated conditional probabilities for the missing data setting. For moth methods the estimated intensity surfaces are given in Appendix A.5.

4.5 Conclusions and Discussion

We propose a new multi-task learning framework for modelling correlated count data based on LGCP models. We consider observations on different tasks as being drawn from distinct LGCPs with correlated intensities determined by linearly combined GPs through task-specific random weights. By considering stochastic weights, we allow for the incorporation of additional dependencies across tasks while providing better uncertainty quantification. We derive closed-form expressions for the moments of the intensity functions and use them to develop an efficient variational inference scheme that is orders of magnitude faster than sampling based approaches. We show how MCPM achieves the state of the art performance on both synthetic and real datasets providing a more flexible and up to 15 times faster methodology compared to the benchmarks.

Models incorporating correlation structure in the likelihood function, as seen in this chapter, or in the posterior distribution, as we shall see in Chapter 5, are particularly useful when used within decision-making algorithms. By properly quantifying uncertainty, they allow the acquisition function constructed based on their properties to efficiently explore the actions space, correctly balancing exploration and exploitation. In addition, the efficient inference scheme developed in this chapter enables fast updating of the posterior distributions and can be thus applied when actions are selected sequentially. Interestingly, while MCPM captures *correlation* structure, multi-task models can be developed to capture *causation* structure. This will be the topic of Chapter 7. In particular, we will see how correlated functions measuring cause-effect relationships can be

modelled through a GP multi-task formulation similar to the one introduced here.

MCPM has two main weaknesses. On the one hand, the mixing weights interpretability is limited. Placing an alternative sparse prior distribution [Titsias and Lázaro-Gredilla, 2011] on $\mathbf{W}$ would induce sparsity and thus act as a model selection mechanism for Q. A sparse prior would also shed light on the contribution that each latent process has in determining the intensity function of different tasks while potentially speeding up the algorithm. On the other hand, MCPM considers discretized data on a computational grid. While this significantly simplifies inference, the grid size is an ad-hoc choice and it might lead to poor approximations, especially in high dimensional settings. The investigation of alternative prior structure for $\mathbf{W}$ remains an open research direction. We focus on continuous PPP in the next chapter and see how it is possible to develop a scalable PPP model which avoids the input space discretization while allowing for a scalable and efficient structured variational inference scheme.

Chapter 5

Structured Variational Inference in Continuous Cox Process Models

As discussed in the previous chapter, point processes have been effectively used to model various types of event data ranging from occurrences of diseases [Diggle et al., 2013; Lloyd et al., 2015] to the location of earthquakes [Marsan and Lengline, 2008] or crime events [Aglietti et al., 2019; Grubesic and Mack, 2008]. The most commonly adopted class of models for such discrete data are non-homogeneous Poisson processes and in particular Cox processes [Cox, 1955]. In these, the observed events are assumed to be generated from a Poisson point process (PPP) whose intensity is stochastic, enabling non-parametric inference and uncertainty quantification. Gaussian processes (GPs) have been used to model the intensity of a Cox process via a non-linear positive link function. Typical mappings are the exponential [Diggle et al., 2013; Møller et al., 1998], the square [Lian et al., 2015; Lloyd et al., 2015] and the sigmoidal [Adams et al., 2009; Donner and Opper, 2018; Gunter et al., 2014] transformations.

In general, inferring the intensity function of a GP modulated PPP over a continuous input space $\mathcal{X}$ is highly problematic, and different algorithms have been proposed to deal with this issue depending on the transformation used. For example, under the exponential transformation, a regular computational grid is commonly introduced [Diggle et al., 2013]. This is also the approach adopted in Chapter 4 to construct a multi-task PPP model enabling an efficient inference scheme. Discretization significantly simplifies inference but also leads to poor approximations, especially in high-dimensional settings. Increasing the resolution of the grid improves the approximation but yields computationally prohibitive algorithms that do not scale, highlighting the well-known trade-off between statistical performance and computational cost. A variety

of algorithms have been proposed to deal with a continuous $\mathcal{X}$ but they are computationally expensive [Adams et al., 2009; Gunter et al., 2014], are limited to simple covariance functions [Lloyd et al., 2015], require accurate numerical integration over the domain [Donner and Opper, 2018] or do not account for the model dependencies in the posterior distribution [Donner and Opper, 2018].

Both discretizing the input space and assuming factorized posterior distributions might severely hinder uncertainty quantification. As discussed above and as we shall see in Chapter 6 and Chapter 7, uncertainty quantification is a crucial feature of surrogate models used for selecting actions. Proper uncertainty quantification can be achieved by avoiding likelihood approximations, retaining posterior dependencies, and accounting for correlation structure across multiple processes, as done in Chapter 4. In addition, selecting actions in real-time (or near real-time) requires fast approximation of the posterior distributions as data are sequentially collected. All these important features will be tackled in this chapter by proposing an inference framework that addresses the modelling and inference limitations of existing continuous PPP frameworks. In particular, we develop a tractable representation of the PPP likelihood via augmentation with a superposition of PPPs. This enables a scalable structured variational inference algorithm (SVI) in the continuous space directly, where the approximate posterior distribution incorporates dependencies between the variables of interest. Our specific contributions are as follows.

Scalable inference in the continuous input space The augmentation of the input space via a process superposition view allows us to develop a scalable variational inference algorithm that does not require discretization of the inputs space or *accurate* numerical integration. With this view, we obtain a joint likelihood function that is readily normalized, providing a natural regularization over the latent variables in our model.

Efficient structured posterior estimation We estimate a joint posterior distribution via an efficient variational inference scheme that captures the complex variable dependencies in the model while being significantly faster than sampling approaches.

State-of-the-art performance We offer an extensive experimental evaluation that shows the benefits of our approach when compared to various state-of-the-art inference schemes, alternative link functions, and different augmentation schemes or representations of the input space $\mathcal{X}$.

5.1 The STVB framework

In this chapter we consider learning problems where we are given a dataset of N events $\mathcal{D} = \{\mathbf{x}_n\}_{n=1}^N$, where $\mathbf{x}_n$ is a d-dimensional vector in the compact space $\mathcal{X} \subset \mathbb{R}^D$. With our framework, which we call STVB, we aim at modelling these data via a PPP, inferring the latent intensity function $\lambda(\mathbf{x}) : \mathcal{X} \to \mathbb{R}^+$ and making probabilistic predictions. Notice how, differently from Chapter 4, we are focusing on a single-task model. The extension of the proposed model to multi-task settings is an interesting open challenge.

5.1.1 Sigmoidal Gaussian Cox process

Consider a realization $\xi = (N, \{\mathbf{x}_1, ..., \mathbf{x}_n\})$ of a PPP on $\mathcal{X}$ where the points $\{\mathbf{x}_1, ..., \mathbf{x}_n\}$ are treated as *indistinguishable* apart from their locations [Daley and Vere-Jones, 2003]. Conditioned on $\lambda(\mathbf{x})$, the Cox process likelihood function evaluated at ξ can be written as:

$$\mathcal{L}(\xi|\lambda(\mathbf{x})) = \exp\left(-\int_{\mathcal{X}} \lambda(\mathbf{x}) d\mathbf{x}\right) \prod_{n=1}^{N} \lambda(\mathbf{x}_n), \tag{5.1}$$

where the intensity is given by $\lambda(\mathbf{x}) = \lambda^\star \sigma(f(\mathbf{x}))$ with $\lambda^\star > 0$ being an upper-bound on $\lambda(\mathbf{x})$, $\sigma(\cdot)$ denoting the the logistic sigmoid function and f is drawn from a zero-mean GP prior with covariance function $k(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta})$ and hyperparameters $\boldsymbol{\theta}$, i.e. $f|\boldsymbol{\theta} \sim \mathcal{GP}(\mathbf{0}, k(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta}))$. We will refer to this joint model as the sigmoidal Gaussian Cox process (SGCP). Notice that, when considering the tuple $(\mathbf{x}_1, ..., \mathbf{x}_n)$ instead of the set $\{\mathbf{x}_1, ..., \mathbf{x}_n\}$, and thus the event $\xi_0 = (N, (\mathbf{x}_1, ..., \mathbf{x}_n))$, the likelihood function is given by $\mathcal{L}(\xi_0|\lambda(\mathbf{x})) = \frac{\mathcal{L}(\xi|\lambda(\mathbf{x}))}{N!}$. There are indeed $N!$ permutations of the events $\{\mathbf{x}_1, ..., \mathbf{x}_n\}$ giving the same point process realization. When the set $\{\mathbf{x}_1, ..., \mathbf{x}_n\}$ is known, considering $\mathcal{L}(\xi|\lambda(\mathbf{x}))$ or $\mathcal{L}(\xi_0|\lambda(\mathbf{x}))$ does not affect the inference procedure. The same holds for MCMC algorithms inferring the event locations. In this case, the factorial term disappears in the computation of the acceptance ratio. However, as we shall see later, when the event locations are latent variables in a model and inference proceeds via a variational approximation the difference between the two likelihoods is essential. Indeed, while $\mathcal{L}(\xi_0|\lambda(\mathbf{x}))$ is normalized with respect to N, one must be cautious when integrating the likelihood in Eq. (5.1) over sets and bring back the missing $N!$ factor so as to obtain a proper discrete probability mass function for N.

Inference in SGCP is *doubly intractable*, as it requires solving the integral in Eq. (5.1) and computing the intractable posterior distribution for the latent function at the N event locations and the bounding intensity, i.e. $p(\mathbf{f}_N, \lambda^\star|\{\mathbf{x}_n\}_{n=1}^N)$,

which in turns requires computing the marginal likelihood. One way to avoid the first source on intractability (integral in Eq. (5.1)) is through augmentation of the input space [Adams et al., 2009; Donner and Opper, 2018], a procedure that introduces precisely those latent (event) variables that require explicit normalization during variational inference. We will describe below a process superposition view of this augmented scheme that allows us to define a proper distribution over the joint space of observed and latent variables and carry out posterior estimation via variational inference. By superimposing two PPP with opposite intensities we obtain a homogeneous PPP and thus avoid the integration of the GP over $\mathcal{X}$ while reducing the integral in Eq. (5.1) to the computation of the measure of the input space $\int_{\mathcal{X}} d\mathbf{x}$.

5.1.2 Augmentation via superposition

A very useful property of independent PPPs is that their superposition, which is defined as the combination of events from two processes in a single one, is a PPP, see Section 2.3 for a formal definition. Consider two PPPs with intensities $\lambda(\mathbf{x})$ and $\nu(\mathbf{x})$ and realisations $(N, \{\mathbf{x}_1, ..., \mathbf{x}_n\})$ and $(K, \{\mathbf{y}_1, ..., \mathbf{y}_K\})$ respectively. The combined event $\xi_R = (R = N + K, \{\mathbf{v}_1, ..., \mathbf{v}_R\})$ is a realization of a PPP with intensity given by $(\lambda(\mathbf{x})+\nu(\mathbf{x}))$ where knowledge of which points originated from which process is assumed lost. The likelihood for $\mathcal{L}(\xi_R|\lambda(\mathbf{x}), \nu(\mathbf{x}))$ can be thus written as:

$$\sum_{N=0}^{R} \binom{R}{N} \sum_{P_N \in \mathbb{P}_N} \left(\frac{\exp(-\int_{\mathcal{X}} \lambda(\mathbf{x}) d\mathbf{x})}{N!} \prod_{r \in P_N} \lambda(r) \times \frac{\exp(-\int_{\mathcal{X}} \nu(\mathbf{x}) d\mathbf{x})}{K!} \prod_{r \in P_N^c} \nu(r) \right) \tag{5.2}$$

where $\mathbb{P}_N$ denotes the collection of all possible partitions of size N, P_N represents an element of $\mathbb{P}_N$ and P_N^c is its complement.

Consider now $R = N + K$ to be the total number of events resulting from thinning [Lewis and Shedler, 1979] where N is the number of observed events while K is the number of latent events with stochastic locations $\mathbf{y}_1, ..., \mathbf{y}_K$. We assume that the probability of observing an event is given by $\sigma(f(\mathbf{x}))$ while the probability for the event to be latent is $\sigma(-f(\mathbf{x}))$. In addition, let $\lambda^\star \int_{\mathcal{X}} d\mathbf{x}$ be the expected total number of events. We can see the realization $(N + K, (\mathbf{x}_1, ..., \mathbf{x}_N, \mathbf{y}_1, ..., \mathbf{y}_K))$ as the result of the superposition of two PPPs with intensities $\lambda(x) = \lambda^\star \sigma(f(\mathbf{x}))$ and $\nu(\mathbf{x}) = \lambda^\star \sigma(-f(\mathbf{x}))$. Differently from the standard superposition, we do know which events are observed and which are latent. In writing the likelihood for $(N + K, \{\mathbf{x}_1, ..., \mathbf{x}_N, \mathbf{y}_1, ..., \mathbf{y}_K\})$ we thus do not need to consider all the possible partitions of N. We can write the

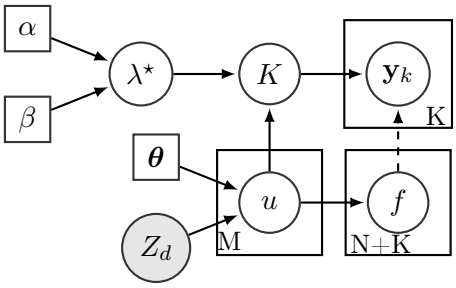


Figure 5.1: Plate diagram representing the posterior distribution accounting for all model dependencies. The only factorisation we introduce in our variational posterior (Eq. (5.6)) is given by the dashed line.

likelihood function $\mathcal{L}_{N+K} \stackrel{\text{def}}{=} \mathcal{L}(N+K, (\mathbf{x}_1, ..., \mathbf{x}_N, \mathbf{y}_1, ..., \mathbf{y}_K))$ as:

$$\begin{aligned}\mathcal{L}_{N+K} &= \frac{\exp(-\int_{\mathcal{X}} \lambda(\mathbf{x}) d\mathbf{x})}{N!} \prod_{r \in P_N} \lambda(r) \times \frac{\exp(-\int_{\mathcal{X}} \nu(\mathbf{x}) d\mathbf{x})}{K!} \prod_{r \in P_N^c} \nu(r) \\ &= \frac{1}{N!K!} \exp(-\lambda^\star \int_{\mathcal{X}} dx)(\lambda^\star)^{N+K} \prod_{n=1}^{N} \sigma(\mathbf{f}(\mathbf{x}_n)) \prod_{k=1}^{K} \sigma(-\mathbf{f}(\mathbf{x}_k)). \quad (5.3)\end{aligned}$$

There is a crucial difference between Eq. (5.3) and the usual likelihood considered in SGCP. Eq. (5.3) represents a distribution over tuples and thus, as mentioned above, is properly normalized. In addition, it makes a distinction between the observed and latent events and it is thus different from Eq. (5.1) written for the tuple $(N+K, \{\mathbf{x}_1, ..., \mathbf{x}_N, \mathbf{y}_1, ..., \mathbf{y}_K\})$. We can write the full joint distribution which is given by $\mathcal{L}^+_{N+K} \stackrel{\text{def}}{=} \mathcal{L}(\{\mathbf{x}_n\}_{n=1}^N, \{\mathbf{y}_k\}_{k=1}^K, K, \mathbf{f}, \lambda^\star | \tau, \boldsymbol{\theta})$ as:

$$\mathcal{L}^+_{N+K} = \frac{(\lambda^\star)^{N+K} \exp(-\lambda^\star \int_{\mathcal{X}} dx)}{N!K!} \prod_{n=1}^{N} \sigma(\mathbf{f}(\mathbf{x}_n)) \prod_{k=1}^{K} \sigma(-\mathbf{f}(\mathbf{y}_k)) \times p(\mathbf{f}) \times p(\lambda^\star), \quad (5.4)$$

where $p(\mathbf{f}) \stackrel{\text{def}}{=} p(\mathbf{f}_{N+K})$ denotes the joint prior at both $\{\mathbf{x}_n\}_{n=1}^N$ and $\{\mathbf{y}_k\}_{k=1}^K$ and $p(\lambda^\star)$ denotes the prior over the upper bound of the intensity function. We consider a prior distribution for $\lambda^\star$ given by $p(\lambda^\star) = \text{Gamma}(a, b)$ and set the parameters a and b so that $\lambda^\star$ as has mean and standard deviation equal to 2 times and 1 time the intensity we would expect from a homogeneous Poisson process on $\mathcal{X}$. Notice how the marginal likelihood corresponding to the joint distribution in Eq. (5.4) can be obtained by integrating out all latent variables. However, this cannot be derived analytically and the following variational scheme is derived starting from the full joint distribution.

5.1.3 Scalability via inducing variables

As in standard GP modulated models, the introduction of a GP prior poses significant computational challenges during posterior estimation as inference

would be dominated by algebraic operations that are cubic on the number of observations. In order to make inference scalable, we follow the inducing-variable approach proposed by Titsias [2009b] and further developed by Bonilla et al. [2019]. See Section 2.1.2 for a introduction on sparse GP approximations.

We consider an augmented prior $p(\mathbf{f}, \mathbf{u})$ with M underlying inducing variables denoted by $\mathbf{u}$. The corresponding inducing inputs are given by the $M \times D$ matrix $\mathbf{Z}$. Major computational gains are realized when $M \ll N + K$. The augmented prior distributions for the inducing variables and the latent functions are given by:

$$\begin{aligned} p(\mathbf{u}|\boldsymbol{\theta}) &= \mathcal{N}(\mathbf{0}, \mathbf{K}_{zz}) \\ p(\mathbf{f}|\mathbf{u}, \boldsymbol{\theta}) &= \mathcal{N}(\mathbf{K}_{xz}(\mathbf{K}_{zz})^{-1}\mathbf{u}, \mathbf{K}_{xx} - \mathbf{A}\mathbf{K}_{zx}) \end{aligned}$$

where $\mathbf{A} = \mathbf{K}_{xz}(\mathbf{K}_{zz})^{-1}$. The matrices $\mathbf{K}_{xx}$, $\mathbf{K}_{xz}$, $\mathbf{K}_{zx}$ and $(\mathbf{K}_{zz})^{-1}$ are the covariance matrices induced by evaluating the corresponding covariance functions at all pairwise rows of the event locations $\{\mathbf{x}_n, \mathbf{y}_k\}_{n=1,k=1}^{N,K}$ and the inducing inputs $\mathbf{Z}$.

5.2 Structured Variational Inference

Given the joint distribution in Eq. (5.4), our goal is to estimate the posterior distribution over all latent variables given the data. i.e. $p(\mathbf{f}, \mathbf{u}, M, \{\mathbf{y}_k\}_{k=1}^K, \lambda^\star|\mathcal{D})$ which can be obtained by computing:

$$p(\mathbf{f}, K, \{\mathbf{y}_k\}_{k=1}^K, \lambda^\star|\mathcal{D}) = \frac{\mathcal{L}_{N+K}^+}{\int\int\int\int \mathcal{L}_{N+K}^+ \mathrm{d}\mathbf{f}\mathrm{d}K\mathrm{d}\{\mathbf{y}_k\}_{k=1}^K\mathrm{d}\lambda^\star} \tag{5.5}$$

and requires integrating out all latent variables from Eq. (5.4). This posterior is analytically intractable and we approximate it by resorting to variational inference [Jordan et al., 1999]. Recall from Section 2.1.2 that variational inference entails defining an approximate posterior $q(\mathbf{f}, \mathbf{u}, K, \{\mathbf{y}_k\}_{k=1}^K, \lambda^\star)$ and optimizing the so-called evidence lower bound (ELBO) with respect to this distribution. In SGCP, the GP and the latent variables are highly coupled and breaking their dependencies would lead to poor approximations, especially in high dimensional settings. Fig. 5.1 shows the structure of a general posterior distribution for SGCP without any factorisation assumption.

We consider an approximate posterior distribution that takes dependencies into account:

$$q(\mathbf{f}, \mathbf{u}, K, \{\mathbf{y}_k\}_{k=1}^K, \lambda^\star) = p(\mathbf{f}|\mathbf{u})q(\{\mathbf{y}_k\}_{k=1}^K|K)q(K|\mathbf{u}, \lambda^\star)q(\mathbf{u})q(\lambda^\star). \tag{5.6}$$

With respect to the general posterior distribution, the only factorisation we impose in Eq. (5.6) is in the factor $q(\{\mathbf{y}_k\}_{k=1}^K|K)$ where we drop the dependency on $\mathbf{f}$, see dashed line in Fig. 5.1. Additionally we set:

$$\begin{aligned} q(\mathbf{u}) &= \mathcal{N}(\mathbf{m}, \mathbf{S}) \\ q(\lambda^\star) &= \text{Gamma}(\alpha, \beta) \\ q(\{\mathbf{y}_k\}_{k=1}^K|K) &= \prod_{k=1}^K \sum_{s=1}^S \pi_s \mathcal{N}_T(\mu_s, \sigma_s^2; \mathcal{X}) \end{aligned}$$

where $\mathcal{N}_T(\cdot; \mathcal{X})$ denotes a truncated Gaussian distribution on $\mathcal{X}$. The factorisation assumption between $\mathbf{f}$ and $\{\mathbf{y}_k\}_{k=1}^K$ can be relaxed by considering a PPP with intensity $\lambda^\star\sigma(-f(\mathbf{x}))$ as the joint variational distribution $q(K, \{\mathbf{y}_k\}_{k=1}^K)$, which is indeed the true posterior distributions for the number of thinned events and their locations [Daley and Vere-Jones, 2003].

Considering a fully structured posterior distribution significantly increases the computational cost of the algorithm as it would require sampling from the full posterior in the computation of the ELBO. The mixture of truncated Gaussians provides a flexible and computationally advantageous alternative while satisfying the constraint of being within the domain of interest. More importantly, we assume:

$$\begin{aligned} q(K|\mathbf{u}, \lambda^\star) &= \text{Poisson}(\eta) \quad \text{with} \\ \eta &= \lambda^\star \int_{\mathcal{X}} \sigma(-\mathbf{u}(\mathbf{x}))\mathrm{d}\mathbf{x}. \end{aligned}$$

This is indeed the *true* conditional posterior distribution for the number of thinned points, see Proposition (3.7) in Moller and Waagepetersen [2003]. Considering $q(K|\mathbf{u}, \lambda^\star)$ we thus fully account for the dependency structure existing among K, $\mathbf{u}$ and $\lambda^\star$. Crucially, while in this work we estimate $\int_{\mathcal{X}} \sigma(-\mathbf{u}(\mathbf{x}))d\mathbf{x}$ via a Monte Carlo integral approximation, STVB does not require *accurate* estimation of this term. Indeed, differently from the competing techniques, where the algorithm convergence and the posterior $q(\mathbf{f})$ is *directly* dependent on numerical integration, STVB only requires evaluation of the integral during the optimisation but $q(\mathbf{f})$ and thus $\lambda(\mathbf{x})$ do not directly depend on its value. In other words, the quality of the posterior intensity does not depend directly on how accurate the integral estimation is.

Table 5.1: Summary of related work. $\int$ and $\sum$ denote continuous and discrete models respectively. K represents the number of thinned points derived from the thinning of a PPP. M indicates the number of inducing inputs.

	STVB	LGCP [191]	SGCP [6]	Gunter et al. [2014]	VBPP [168]	Lian et al. [2015]	MFVB [64]
Inference	SVI	MCMC	MCMC	MCMC	VI-MF	VI-MF	VI-MF
$\mathcal{O}$	M^3	N^3	$(N+K)^3$	$(N+K)^3$	NM^2	NM^2	NM^2
$\boldsymbol{\lambda(x)}$	$\lambda^\star\sigma(f(x))$	$\exp(f(x))$	$\lambda^\star\sigma(f(x))$	$\lambda^\star\sigma(f(x))$	$(f(x))^2$	$(f(x))^2$	$\lambda^\star\sigma(f(x))$
$\boldsymbol{\mathcal{X}}$	$\int$	$\sum$	$\int$	$\int$	$\int$	$\sum$	$\int$

5.2.1 Evidence Lower Bound

Following standard variational inference arguments, it is straightforward to show that the ELBO decomposes as:

$$\begin{aligned}\mathcal{L}_{\text{elbo}} = N(\psi(\alpha) - \log(\beta)) - V\frac{\alpha}{\beta} - \log(N!) + \underbrace{\mathbb{E}_Q[K\log(\lambda^\star)]}_{T_1} - \underbrace{\mathbb{E}_Q[\log(K!)]}_{T_2} + \\ + \sum_{n=1}^{N}\mathbb{E}_{q(\mathbf{f})}[\log(\sigma(\mathbf{f}(\mathbf{x}_n)))] + \underbrace{\mathbb{E}_Q\left[\sum_{k=1}^{K}\log(\sigma(-\mathbf{f}(\mathbf{y}_k)))\right]}_{T_3} \\ - \mathcal{L}_{\text{kl}}^{\mathbf{u}} - \mathcal{L}_{\text{kl}}^{\lambda^\star} - \underbrace{\mathcal{L}_{\text{ent}}^{K}}_{T_4} - \underbrace{\mathcal{L}_{\text{ent}}^{\{\mathbf{y}_k\}_{k=1}^K}}_{T_5}\end{aligned} \tag{5.7}$$

where $V = \int_{\mathcal{X}} d\mathbf{x}$, $\psi(\cdot)$ is the digamma function and $q(\mathbf{f}) = \mathcal{N}(\mathbf{Am}, \mathbf{K}_{xx} - \mathbf{AK}_{zx} + \mathbf{ASA}')$. The terms denoted by $T_i, i = 1, ..., 5$ cannot be computed analytically. Naïvely, black-box variational inference algorithms could be used to estimate these terms via Monte Carlo, thus sampling from the full variational posterior (Eq. (5.6)). This would require sampling $\mathbf{f}$, $\lambda^\star$, K and $\{\mathbf{y}_k\}_{k=1}^K$ thus slowing down the algorithm while leading to slow convergence. On the contrary, we exploit the structure of the model and the approximate posterior to simplify these terms and increase the algorithm efficiency. Denote $\mu(\mathbf{u}) = \int_{\mathcal{X}} \sigma(-\mathbf{u}(\mathbf{x}))d\mathbf{x}$, we can write:

$$T_1 = \mathbb{E}_{q(\lambda^\star)}[\lambda^\star\log(\lambda^\star)]\mathbb{E}_{q(\mathbf{u})}[\mu(\mathbf{u})] \tag{5.8}$$

$$T_3 = \frac{\alpha}{\beta}\mathbb{E}_{q(\mathbf{u})}[\mu(\mathbf{u})]\mathbb{E}_{q(\mathbf{f})q(\mathbf{y}_k)}[\log(\sigma(-\mathbf{f}(\mathbf{y}_k)))] \tag{5.9}$$

$$T_4 = \frac{\alpha}{\beta}\mathbb{E}_{q(\mathbf{u})}[\mu(\mathbf{u})\left[\log(\mu(\mathbf{u})) - 1\right]] \tag{5.10}$$

$$+ \mathbb{E}_{q(\lambda^\star)}[\lambda^\star\log(\lambda^\star)]\mathbb{E}_{q(\mathbf{u})}[\mu(\mathbf{u})] - \mathbb{E}_Q[\log(K!)] \tag{5.11}$$

$$T_5 = \frac{\alpha}{\beta}\mathbb{E}_{q(\mathbf{y}_k)}[\log q(\mathbf{y}_k)]\mathbb{E}_{q(\mathbf{u})}[\mu(\mathbf{u})] \tag{5.12}$$

Notice how the term $-\mathbb{E}_Q[\log(K!)]$ in T_4, which would require further approximations, appears with opposite sign in T_2 and thus cancels out in the

computation of the ELBO in Eq. (5.7). See Appendix B.1 in the supplementary material for the full derivations.

Eqs. (5.8)–(5.12) give an expression for $\mathcal{L}_{\text{elbo}}$ which avoids sampling from $q(K|\mathbf{u}, \lambda^\star)$ and $q(\{\mathbf{y}_k\}_{k=1}^K|K)$ and does not require computing the GP on the stochastic locations. The remaining expectations are with respect to reparameterizable distributions. We thus avoid the use of score function estimators which would lead to high-variance gradient estimates. Stochastic optimisation techniques can be used to evaluate T_3 and $\sum_{n=1}^N \mathbb{E}_{q(\mathbf{f})}[\log(\sigma(\mathbf{f}(\mathbf{x}_n)))]$ thus reducing the computational cost by making it independent of K and N. This would further reduce the computational complexity of the algorithm to $\mathcal{O}(M^3)$. However, when the number of inputs used per mini-batch equals N, the time complexity becomes $\mathcal{O}(NM^2)$. In the following experiments, we show how the proposed structured approach together with these efficient ELBO computations leads to higher predictive performances and better uncertainty quantification. The presented results do not exploit the computational gains attainable via stochastic optimisation thus the CPU times and performances are directly comparable across all methods.

5.3 Related work

We review the more closely related works and, to facilitate comparison, we provide a summary table of the main differences across them (Table 5.1).

Discretization and numerical integration GP-modulated Poisson point processes are the gold standard for modelling event data. Performing inference in these models, e.g. under the exponential transformation has typically required discretization where the domain is gridded and the intensity function is assumed to be constant over each grid cell [Brix and Diggle, 2001; Cunningham et al., 2008; Diggle et al., 2013; Møller et al., 1998]. Alternatively, Lasko [2014] also considers an exponential link function and performs inference over a renewal process resorting to numerical integration within a computationally expensive sampling scheme which scales as $\mathcal{O}(k^3) + \mathcal{O}(N)$ with k denoting the number of integration points. These methods suffer from poor scaling with the dimensionality of $\mathcal{X}$ and sensitivity to the choice of the discretization or numerical integration technique. Several approaches have been proposed to deal with inference in the continuous domain directly by using alternative transformations along with additional modelling assumptions and computational tricks or by constraining the GP [López-Lopera et al., 2019].

Alternative link functions One of those alternative transformation is the squared mapping which leads to the so-called Permanental process [John and

Hensman, 2018; Lian et al., 2015; Lloyd et al., 2015; Lloyd et al., 2016; Walder and Bishop, 2017] where the intensity function is given by $\lambda(\mathbf{x}) = f^2(\mathbf{x})$. Although the square transformation enables analytical computation of the required integrals over $\mathcal{X}$, this only holds for certain standard types of kernels such as the squared exponential. In addition, Permanental processes suffer from important identifiability issues such as reflection invariance and lead to models with "nodal lines" [John and Hensman, 2018].

Another frequently used transformation is the scaled logistic sigmoid function which leads to the so called sigmoidal Gaussian Cox process (SGCP) where the intensity function is defined as $\lambda(\mathbf{x}) = \sigma(f(\mathbf{x}))$ with $\sigma(\mathbf{x}) = (1+\exp(-\mathbf{x}))^{-1}$. This transformation has been used by Adams et al. [2009], which exploits augmentation of the input space via thinning [Lewis and Shedler, 1979] to achieve tractability. Their proposed inference algorithm is based on Markov chain Monte Carlo (MCMC), which enables drawing 'exact' samples from the posterior intensity. However, as acknowledged by the authors, it has significant computational demands making it inapplicable to large datasets. As an extension to this work, Gunter et al. [2014] introduce the concept of "adaptive thinning" and propose an expensive MCMC scheme which scales as $\mathcal{O}(N^3)$. More recently, Donner and Opper [2018] introduced a neat double augmentation scheme for SGCP which enables closed-form updates using a mean-field approximation (VI-MF). However, it requires accurate numerical integration over $\mathcal{X}$, which makes the performance of the algorithm highly dependent on the number of integration points.

The framework proposed in this chapter overcomes the limitations of the mentioned VI-MF and MCMC schemes by proposing a SVI framework that takes into account the complex posterior dependencies while being scalable and thus applicable to high-dimensional real-world settings. To the best of our knowledge, this is the first formulation offering a fast structured variational inference framework for GP modulated Poisson point process models.

5.4 Experiments

We test our algorithm on three 1D synthetic data settings, two 2D real-world applications and one 3D setting. Code and data for all the experiments are provided at `https://github.com/VirgiAgl/STVB`.

Baselines We compare against alternative inference schemes, different link functions and a different augmentation scheme. In terms of continuous models, we consider the sampling approach of Adams et al. [2009] (SGCP), a Permanental Point process model by Lloyd et al. [2015] (VBPP) and a mean-field

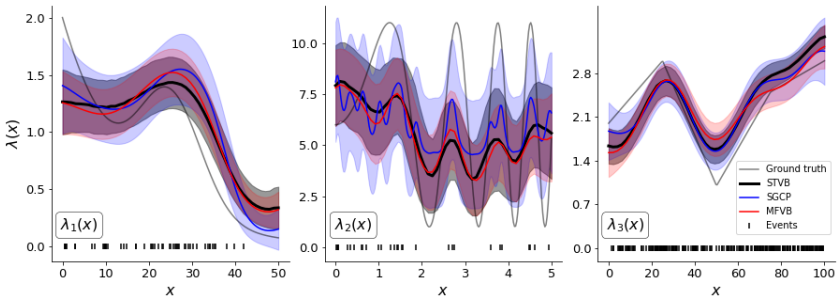


Figure 5.2: Qualitative results on synthetic data. Solid colored lines denote posterior mean intensities while shaded areas are ± standard deviation.

approximation based on a Pólya-Gamma augmentation proposed by Donner and Opper [2018] (MFVB). In addition, we compare against a discrete variational log Gaussian Cox process model based on Nguyen and Bonilla [2014] (LGCP). Further details about the experimental settings are given in Appendix B.3.

Performance measures We test the algorithms evaluating the l_2 norm to the true intensity function (for the synthetic datasets), the test log-likelihood (ℓ_{test}) on the test set, and the negative log predicted likelihood (NLPL) on the training set. In order to assess the model capabilities in terms of uncertainty quantification, we compute the empirical coverage (EC), i.e. the coverage of the empirical count distributions obtained by sampling events from the posterior intensity function. We do that for different credible intervals (CI) on both the training set, to compute the in-sample distribution $p(N|\mathcal{D})$), and test set to compute the out-of-sample distribution $p(N^*|\mathcal{D})$). Details on the metrics computation are given in Appendix B.2. For the synthetic data experiments, we run the algorithms with 10 training datasets each including a different PPP realization sampled from the ground truth intensity function. For each different training set, we then evaluate the performance on 10 unseen realizations sampled again from the ground truth intensity. We compute the mean and the standard deviation for the presented metrics averaging across the training and test sets. For the real data settings, we compute the NLPL and in-sample EC on the observed events. We then test the algorithm computing both ℓ_{test} and out-of-sample EC on the held-out events. In order to compute the out-of-sample EC we rescale the intensity function as $\lambda_{test}(\mathbf{x}) = \lambda_{train}(\mathbf{x}) - N_{train}/V + N_{test}/V$ with $V = \int_{\mathcal{X}} d\mathbf{x}$. We then sample events from $\lambda_{test}(\mathbf{x})$ and generate the predicted count distributions for different seeds.

Synthetic experiments We test our approach using the three toy example proposed by Adams et al. [2009]:

- $\lambda_1(\mathbf{x}) = 2\exp(-1/15) + \exp(-[(x-15)/10]^2)$ with $x \in [0, 50]$,

Table 5.2: Average performances on synthetic data across 10 training and 10 test datasets with standard errors in brackets. Our method is denoted by STVB. *Top*: Lower values of l_2 and NLPL are better. Higher values of ℓ_{test} are better. *Bottom*: Out-of-sample EC for different CI, higher values are better.

	$\lambda_1(x)$			$\lambda_2(x)$			$\lambda_3(x)$			CPU time (s)
	l_2	ℓ_{test}	NLPL	l_2	ℓ_{test}	NLPL	l_2	ℓ_{test}	NLPL	
STVB	**3.44** (1.43)	**-1.39** (1.05)	4.71 (0.51)	46.28 (9.95)	56.04 (4.47)	5.62 (0.72)	**7.39** (2.76)	153.98 (11.91)	6.41 (0.64)	315.59
MFVB	4.56 (1.43)	-2.84 (1.0)	4.74 (0.1)	44.44 (10.7)	55.35 (4.72)	5.52 (1.29)	8.17 (3.43)	155.08 (10.20)	5.82 (0.61)	0.01
VBPP	9.19 (2.32)	-7.71 (3.31)	8.91 (1.19)	48.15 (13.16)	**56.82** (4.42)	5.20 (1.33)	20.54 (6.53)	152.82 (11.43)	8.35 (2.28)	0.44
SGCP	4.22 (1.88)	**-1.39** (1.28)	**4.21** (1.04)	**43.50** (8.69)	55.05 (1.35)	**3.77** (0.54)	14.44 (2.97)	**165.66** (2.12)	**4.78** (0.33)	2764.88
LGCP	67.76 (24.38)	-5.26 (8.84)	26.26 (8.09)	106.74 (13.89)	28.56 (6.88)	15.75 (3.36)	19.24 (6.44)	147.67 (11.76)	10.84 (1.36)	4.74

	EC–$\lambda_1(x)$			EC–$\lambda_2(x)$			EC–$\lambda_3(x)$		
	30% CI	40% CI	50% CI	30% CI	40% CI	50% CI	30% CI	40% CI	50% CI
STVB	**0.81** (0.27)	**0.72** (0.27)	**0.6** (0.34)	**0.91** (0.24)	**0.88** (0.23)	**0.86** (0.22)	**0.99** (0.03)	0.97 (0.09)	0.92 (0.15)
MFVB	0.76 (0.25)	0.61 (0.28)	0.52 (0.29)	0.89 (0.23)	0.84 (0.29)	0.82 (0.29)	0.97 (0.09)	0.91 (0.14)	0.78 (0.15)
VBPP	0.75 (0.21)	0.41 (0.25)	0.04 (0.09)	0.76 (0.26)	0.45 (0.26)	0.05 (0.05)	0.83 (0.19)	0.43 (0.14)	0.03 (0.05)
SGCP	0.39 (0.28)	0.27 (0.22)	0.08 (0.12)	0.64 (0.09)	0.14 (0.05)	0.00 (0.00)	0.49 (0.03)	0.34 (0.07)	0.02 (0.04)
LGCP	0.08 (0.12)	0.03 (0.09)	0.01 (0.03)	0.04 (0.08)	0.00 (0.00)	0.00 (0.00)	**0.99** (0.00)	**0.99** (0.12)	**0.95** (0.10)

- $\lambda_2(\mathbf{x}) = 5\sin(x^2) + 6 \quad$ with $\quad x \in [0, 5]$,
- $\lambda_3(\mathbf{x})$ piecewise linear through $(0, 20)$, $(25, 3)$, $(50, 1)$, $(75, 2.5)$ and $(100, 3)$ with $\quad x \in [0, 100]$.

For LGCP, we discretize the input space considering a grid cell width of one for $\lambda_1(\mathbf{x})$ and $\lambda_3(\mathbf{x})$ and of 0.5 for $\lambda_2(\mathbf{x})$. For MFVB we consider 1000 integration points. In terms of $q(\{\mathbf{y}_k\}_{k=1}^K|K)$, we set $S = 5$ but consistent results where found across different values of this parameter. The results are given in Fig. 5.2 and Table 5.2, where we see that all algorithms recover similar predicted mean intensities and give roughly comparable performances across all metrics. Out of all 9 settings and metrics (top section of Table 5.2) our method (STVB) outperforms competing methods on 3 cases and it is only second to SGCP on 6 cases. However, the CPU time of SGCP is almost an order of magnitude larger than ours even in these simple low-dimensional problems, making that method inapplicable to large datasets. This confirms the benefits of having structured approximate posteriors within a computationally efficient inference algorithm such as VI. In terms of uncertainty quantification (bottom section of Table 5.2), our algorithm outperforms all competing approaches for $\lambda_1(\mathbf{x})$ and $\lambda_2(\mathbf{x})$.

2D real data experiments In this section we show the performance of the algorithm on two 2D real-world datasets. In both cases, we assume independent

Table 5.3: Average performances on real-data experiments with standard errors in brackets. EC is computed across 100 replications using different seeds. Higher ℓ_{test} and EC values are better. Lower NLPL values are better. EC figures are given as In-sample - Out-of-sample.

	Neuronal data				
	$\ell_{test}[\times 10^3]$	NLPL	EC-30% CI	EC-40% CI	CPU time (s)
STVB	-84.55 (16.05)	**10.10** (7.02)	**1.00**-**1.00** (0.00)-(0.00)	**0.99**-**0.56** (0.10)-(0.50)	193.07
MFVB	**-83.54** (4.60)	10.71 (3.39)	**1.00**-0.03 (0.00)-(0.17)	0.78-0.00 (0.41)-(0.00)	0.35
VBPP	-83.89 (12.49)	11.39 (8.18)	**1.00**-0.00 (0.00) - (0.00)	0.83-0.00 (0.38)-(0.00)	26.23
	Taxi data				
	$\ell_{test}[\times 10^6]$	NLPL $[\times 10^4]$	EC-30% CI	EC-40% CI	CPU time (s)
STVB	**-27.96** (9.16)	**27.96** (9.16)	0.81-**0.37** (0.39)-(0.48)	0.09-**0.01** (0.29)-(0.10)	290.34
MFVB	-40.8 (6.41)	40.65 (6.41)	0.00-0.00 (0.00)-(0.00)	0.00-0.00 (0.00)-(0.00)	0.24
VBPP	-31.32 (8.18)	31.32 (8.18)	**0.98**-0.00 (0.14)-(0.00)	**0.48**-0.00 (0.50)-(0.00)	3.62

two-dimensional truncated Gaussian distributions for $q(\{\mathbf{y}_k\}_{k=1}^K|K)$ so that they factorize across input dimensions. Qualitative and quantitative results are given in Fig. 5.3, Fig. 5.4 and Table 5.3.

Our first dataset is concerned with neuronal data, where event locations correspond to the position of a mouse moving in an arena when a recorded cell fired [Sargolini et al., 2006]. We randomly assign the events to either training ($N = 583$) or test ($N = 29710$) and we run the model using a regular grid of 10×10 inducing inputs. We see that the intensity function recovered by the three methods varies in terms of smoothness with MFVB estimating the smoothest $\lambda(\mathbf{x})$ and VBPP recovering an irregular surface (Fig. 5.3). MFVB gives slightly better performance in terms of ℓ_{test} but our method (STVB) outperforms competing approaches in terms of NLPL and EC figures. Remarkably, STVB contains the true number of test events in the 30% credible intervals for 56% of the simulations from the posterior intensity (Table 5.3 and Fig. 5.4).

As a second dataset, we consider the Porto taxi dataset[1] which contains the trajectories of 7000 taxi travels in the years 2013/2014 in the city of Porto. As in Donner and Opper [2018], we consider the pick-up locations as observations of a PPP and restrict the analysis to events happening within the coordinates $(41.147, -8.58)$ and $(41.18, -8.65)$. We select $N = 1000$ events at random as training set and train the model with 400 inducing points placed on a regular grid. The test log-likelihood is then computed on the remaining 3401 events. We see that our method (STVB) outperforms competing methods on all performance

[1] `http://www.geolink.pt/ecmlpkdd2015-challenge/dataset.html`.

metrics (Table 5.3), recovering an intensity that is smoother than VBPP and captures more structure compared to MFVB (Fig. 5.3). In terms of uncertainty quantification, the coverage of $p(N^*|\mathcal{D})$ are the highest for STVB across all CI. Notice how the irregularity of the VBPP intensity leads to good performance on the training set but results in a $p(N^*|\mathcal{D})$ which is centered on a significantly higher number of test events (Fig. 5.4). As expected, the SVI approach implies wider counts distributions compared to the mean-field approximation. This generally yields better predictive performances in a variety of settings and especially in higher-dimensional experiments.

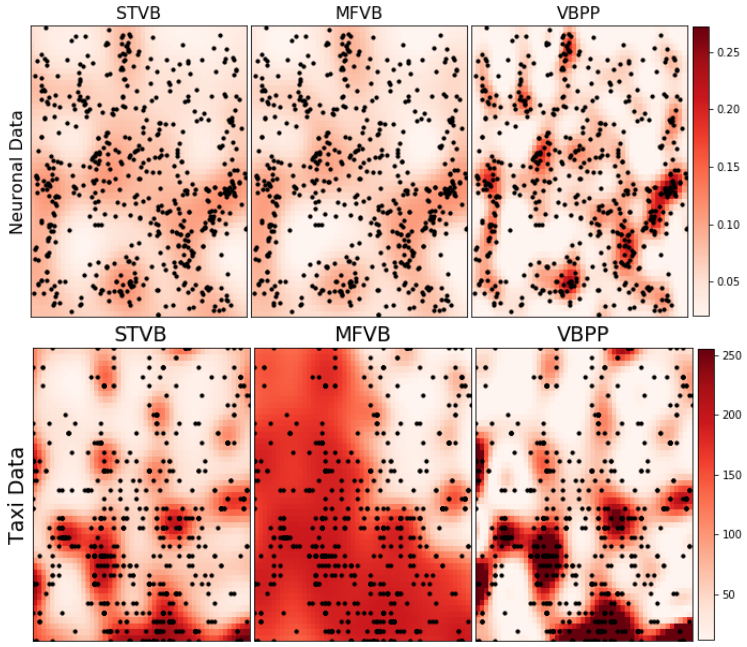


Figure 5.3: Real data. The red surfaces represent the posterior mean intensities inferred with STVB (first column) or the baseline methods (second and third column). The black dots give the observed events on the two-dimensional input space. *Upper*: Neuronal Data. *Lower*: Taxi Data.

3D real data experiment Finally, we show the performance of the STVB algorithm on the spatio-temporal Taxi dataset used above where, for each taxi travel, we consider both the trajectory and the pickup time in seconds. VBPP does not currently support $\mathrm{D} > 2$ thus we compared STVB to MFVB. We found STVB to outperform MFVB both in terms of performance metrics and uncertainty quantification, see Table 5.4 and Fig. 5.5.

5.5 Conclusions and Discussion

In this chapter, we propose a new variational inference framework for estimating the intensity of a continuous sigmoidal Cox process. By considering

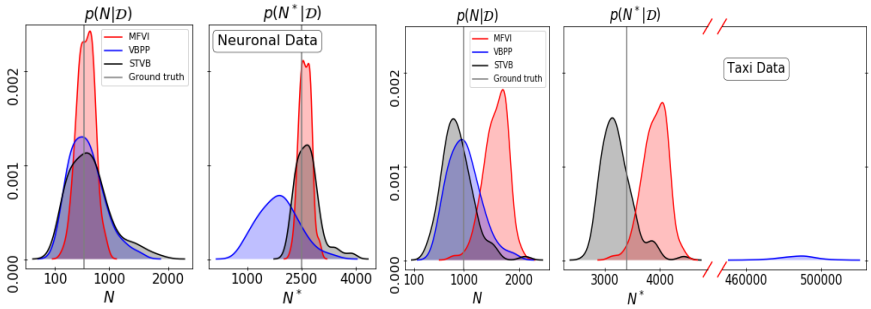


Figure 5.4: Predicted counts distributions for the training set ($p(N|\mathcal{D})$) and the test set ($p(N^*|\mathcal{D})$) on the taxi data (left plots) and the neuronal data (right plots). The gray line denotes the number of observed events. The red bars on the x-axis denote breaks in the axis due to the different shifts of the distributions.

Table 5.4: Average performances on the spatio-temporal Taxi dataset. Standard errors in brackets. EC is computed across 100 replications using different seeds. Higher ℓ_{test} and EC are better. Lower NLPL are better. EC figures are given as In-sample - Out-of-sample.

	Spatio-temporal Taxi Data				
	$\ell_{test}[\times 10^7]$	NLPL$[\times 10^5]$	EC-30% CI	EC-40% CI	CPU time (s)
STVB	**-31.26** (10.88)	**31.26** (10.88)	**1.00**-0.00 (0.00)-(0.00)	**0.98**-0.00 (0.14)-(0.00)	1208.00
VBPP	-42.97 (9.56)	42.97 (9.56)	0.00-0.00 (0.00)-(0.00)	0.00-0.00 (0.00)-(0.00)	1.00

an augmented input space as the result of the superposition of two PPPs, we derive a scalable and computationally efficient structured variational approximation. Our framework does not require discretization or accurate numerical computation of integrals on the input space, it is not limited to specific kernel functions and properly accounts for the strong dependencies existing across the latent variables. Through extensive empirical evaluation, we demonstrate that our method compares favourably against "exact" but computationally costly MCMC schemes while being almost an order of magnitude faster. More importantly, our inference scheme outperforms all competing approaches in terms of uncertainty quantification. The benefit of the proposed scheme and resulting SVI are particularly pronounced on multivariate input settings where accounting for the highly coupled variables becomes crucial for interpolation and prediction and methods based on numerical integration fail.

It is possible to identify two major open challenges. Firstly, the integration of the proposed framework with a multi-task model would make STVB applicable

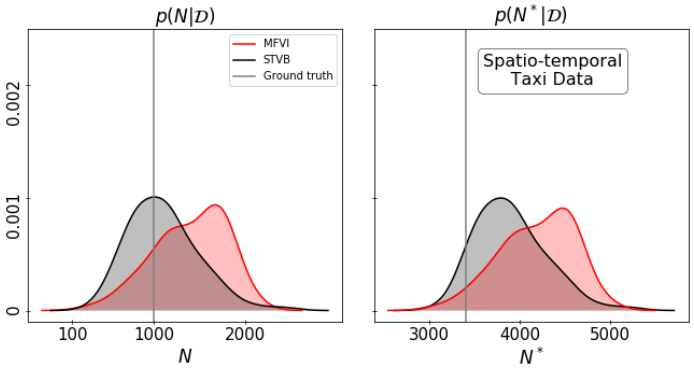


Figure 5.5: Predicted counts distributions for the training set ($p(N|\mathcal{D})$) and the test set ($p(N^*|\mathcal{D})$).

to settings such as those analysed in Chapter 4. Secondly, the relaxation of the factorization assumption between the GP and the latent points. A fully structured variational inference scheme would further improve the accuracy performance of the method but would require introducing additional approximations in the variational objective.

Despite the open research questions mentioned above, the approach developed in this chapter provides a flexible GP model capable of capturing complex data distributions while allowing to quantify uncertainty in a principled way. Retaining structured approximate posteriors while enabling fast scalable inference are two crucial properties of surrogate models used within decision-making algorithms. Indeed, models incorporating correlation structure in the likelihood function, as seen in Chapter 4, or in the posterior approximation, as done with STVB, allow the acquisition function constructed based on their properties to efficiently explore the actions space, correctly balancing exploration and exploitation. In addition, sequential decision-making requires updating the posterior distributions as actions are performed and data are collected. When posterior inference is not closed form, as in Poisson point processes, fast approximate inference schemes are essential to allow sequential selection of actions in "near real-time" or even in real-time. All these important features characterize the models developed in Chapter 4 and Chapter 5.

In the coming chapters, we will see how probabilistic models, such as those developed so far, can be combined with an acquisition function to obtain sequential decision-making algorithms. In particular, we will see how a causation structure rather than a correlation structure can be incorporated in GP surrogate models allowing to select actions based on cause-effect relationships. Interestingly, in Chapter 7, we will link correlation and causation through a causal multi-task formulation that, as done in Chapter 4, captures the correla-

tion structure across a set of functions but where each function represents a causal quantity. In turn, this causal multi-task formulation will lead to complex structured posterior distributions, such as those seen in this chapter, that will significantly improve the performance of decision-making algorithms.

Part II

Causal Sequential Decision-Making with Gaussian Processes

Chapter 5 concludes the part of the thesis focusing on structured models for PPP and connected scalable variational inference schemes. While this first part studied how to incorporate *correlation structure* in the model specification or in the posterior approximation, the second part will investigate how to integrate a *causation structure* in the model specification when a causal graph is available. In addition, it will link correlation and causation through a causal multi-task formulation that captures the correlation structure across a set of causal functions. Specifically, we will study sequential decision-making algorithms which require two main elements: a surrogate model and an acquisition function. Similar to those introduced in the previous chapters, we will construct GP based surrogate models where the prior distribution reflects a set of causal assumptions. Indeed, as seen in Chapter 4 and Chapter 5, GPs capture a variety of data distributions and quantify uncertainty in a principled way. Based on the surrogate models, the acquisition function enables the agent to select the next action by effectively trading off exploration and exploitation. In the following chapters, we will generalise sequential decision-making algorithms to incorporate knowledge about the causation mechanism existing among input and output variables. In Chapter 6 we will extend the Bayesian Optimization framework while in Chapter 7 we will discuss Active Learning and show how to deal with multi-task causal settings. Finally, in Chapter 8 we will focus on dynamic scenarios and show how to take decisions in an evolving causal system.

Chapter 6

Causal Bayesian Optimization

Sequential decision-making problems in a variety of domains, such as biological systems, modern industrial processes, or social systems, require implementing interventions and manipulating variables in order to optimize an outcome of interest. For instance, in strategic planning, companies need to decide how to allocate scarce resources across different projects or business units in order to achieve performance goals. In biology, it is common to change the phenotype of organisms by acting on individual components of complex gene networks. This chapter describes how to find such optimal interventions or policies.

Focusing on a specific example, consider a setting in which Y denotes the crop yield for a specific agricultural product, X denotes soil fumigants, and $\mathbf{Z} = \{Z_1, Z_3, Z_4\}$ represents the eel-worm population at different times [Cochran and Cox, 1957]. Given a causal graph [Pearl, 1995] representing the investigator's understanding of the major causal influences among the variables (Fig. 6.1(a)), she aims at finding the highest yielding intervention in a limited number of seasons and subject to a budget constraint. Each intervention has a cost which is determined by the number of intervened variables, each manipulated variable's cost, and the implemented intervention level.

In order to solve this problem, the investigator could resort to Bayesian Optimization (BO). As seen in Section 3.1, BO is an efficient heuristic to optimize objective functions whose evaluations are costly and when no explicit functional form is available [Jones et al., 1998]. In the setting described above, BO would model the objective function with a surrogate e.g. a GP model and would try to find the global optima by making a series of function evaluations in which all variables are manipulated. BO would thus break the dependency structure existing among X and $\mathbf{Z}$, potentially leading to suboptimal solutions. Indeed, as described later in detail, depending on the structural relationships between variables, intervening on a subgroup might lead to a propagation of effects in the causal graph and a higher final yield. In addition, intervening on all variables is cost-ineffective in cases when the same yield can be obtained

by setting only a subgroup of them. The framework proposed in this chapter combines BO, Gaussian process modelling (GP), and causal inference, offering a novel approach for decision making under uncertainty. Probabilistic causal models are commonly used in disciplines where explicit experimentation may be difficult such as social science or economics. In particular, the *do*-calculus [Pearl, 1995] relates observational distributions to interventional ones (see Section 3.2). It allows predicting the effect of an intervention without explicitly performing it and by solely using observational data. We develop a model which integrates observational and interventional data so as to further reduce the uncertainty around the optimal intervention value and the number of interventions required to find it. Particularly, we make the following contributions:

Causal Global Optimization We formulate a new class of optimization problems called *Causal Global Optimization* (CGO) where the causal structure existing among the input variables is accounted for in the objective functions.

Causal GP surrogate model We solve CGO problems by combining ideas from BO and causal calculus. We propose a Gaussian process (GP) surrogate model, the causal GP, that integrates observational and interventional data via the definition of a causal prior distribution computed through *do*-calculus.

Causal acquisition function We propose an acquisition function, the causal expected improvement (EI), which drives the exploration of different intervention sets and allows selecting the optimal intervention value for each of them.

Causal Bayesian Optimization algorithm We develop an algorithm, henceforth named *Causal Bayesian Optimization* (CBO), that exploits the topological characteristics of the graph, the causal EI, and the proposed GP prior to find an optimal intervention. In doing that, it balances the emerging trade-off between observation and intervention via an ϵ-greedy policy.

Experimental comparison on true and synthetic causal graphs We show the benefits of the proposed approach in a variety of experimental settings featuring different dependency structures, unobserved confounders, and non-manipulative variables. Additionally, we demonstrate the performance of CBO when selecting optimal interventions in two real-world settings.

6.1 Problem Setup

We consider a SCM as defined in Definition 3.1 of Section 3.2 and the associated directed causal acyclic graph (DAG) denoted by $\mathcal{G}$. Within the complete

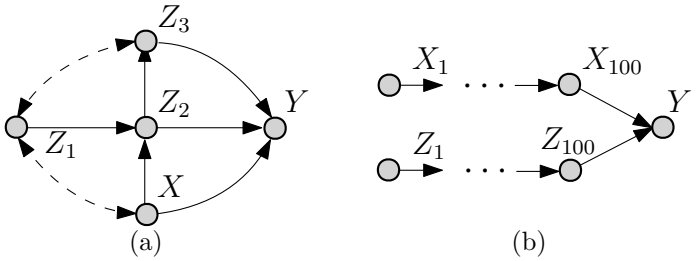


Figure 6.1: Examples of causal graphs. Nodes denote variables, arrows represent causal effects and dashed edges indicate unobserved confounders. *(a)*: Yield optimization example. Y is the crop yield, X denotes soil fumigants and $\mathbf{Z}$ represents the eel-worm population. *(b)*: A 200-dimensional optimization problem with causal intrinsic dimensionality equal to 2.

set of variables in the SCM, we distinguish between three different types of variables: non-manipulative variables $\mathbf{C}$, which cannot be modified, treatment variables $\mathbf{X}$ that can be set to specific values and an output variable Y that represents the agent's outcomes of interest. We denote the *interventional distribution* for two disjoint sets in $\mathbf{V}$, say $\mathbf{X}$ and Y, by $P(Y|\text{do}\,(\mathbf{X}=\mathbf{x}))$. This is the distribution of Y obtained by intervening on $\mathbf{X}$ and fixing its value to $\mathbf{x}$ in the data generating mechanism, irrespective of the values of its parents, and keeping $\mathbf{C}$ unchanged. Conversely $P(\mathbf{Y}|\mathbf{X}=\mathbf{x})$ represents an *observational distribution* which only requires "observing" the system. $\mathcal{D}^O$ and $\mathcal{D}^I$ denote observational and interventional datasets respectively. In this work we assume $\mathcal{G}$ to be known. Causal discovery [Glymour et al., 2019] is a complex topic and analysing what happens when the graph is unknown is left as an open question. As seen in Section 3.2.3, *do*-calculus offers a powerful tool to estimate interventional distributions and causal effects from observational distributions. If the causal effects are identifiable, we can apply the three rules of *do*-calculus to link interventional distributions with observational distributions which can be approximated with e.g. Monte Carlo estimates. The *do*-calculus involves computing integrals which are generally not tractable. When this is the case, observational data can be used to get a Monte Carlo estimate, e.g. $\hat{P}(\mathbf{Y}|\text{do}\,(\mathbf{X}=\mathbf{x})) \approx P(\mathbf{Y}|\text{do}\,(\mathbf{X}=\mathbf{x}))$, which is consistent when the number of samples drawn from $P(\mathbf{V})$ is sufficiently large.

Causal Global Optimization We define a novel class of global optimization problems called *Causal Global Optimization* (CGO). Given $\mathcal{G}$ and $\langle \mathbf{U}, \mathbf{V}, F, P(\mathbf{U})\rangle$, the goal is to select the set of intervention variables $\mathbf{X}_s^\star$ and intervention levels $\mathbf{x}_s^\star$ optimizing the expected target outcomes $\mathbf{Y}$. Formally, the goal is to find:

$$\mathbf{X}_s^\star, \mathbf{x}_s^\star = \underset{\mathbf{X}_s \in \mathcal{P}(\mathbf{X}), \mathbf{x}_s \in D(\mathbf{X}_s)}{\arg\min} \mathbb{E}_{P(\mathbf{Y}|\text{do}(\mathbf{X}_s=\mathbf{x}_s))}[\mathbf{Y}], \tag{6.1}$$

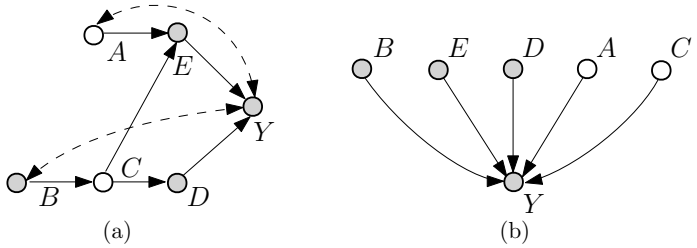


Figure 6.2: DAG representation of a CGO problem (a) and the DAG considered when using BO (b) to address the same problem. Black nodes represent $\mathbf{X}$ while grey shaded nodes give $\mathbf{C}$. Dashed edges indicate unobserved confounders.

where $\mathcal{P}(\mathbf{X})$ is the power set of $\mathbf{X}$, $D(\mathbf{X}_s) = \times_{X \in \mathbf{X}_s}(D(X))$ with $D(X)$ denoting the interventional domain of X and the expectation is computed according to the interventional distribution. For notational convenience, we denote $\mathbb{E}_{P(\mathbf{Y}|\text{do}(\mathbf{X}_s=\mathbf{x}_s))}[\mathbf{Y}] \doteq \mathbb{E}[\mathbf{Y}|\text{do}\,(\mathbf{X}_s = \mathbf{x}_s)]$ and $\mathbb{E}_{\hat{P}(\mathbf{Y}|\text{do}(\mathbf{X}_s=\mathbf{x}_s))}[\mathbf{Y}] \doteq \hat{\mathbb{E}}[\mathbf{Y}|\text{do}\,(\mathbf{X}_s = \mathbf{x}_s)]$. The optimal subset of intervention variables $\mathbf{X}_s$ belongs to $\mathcal{P}(\mathbf{X})$ which includes the empty set $\varnothing$ and $\mathbf{X}$ itself. When $\mathbf{X}_s = \varnothing$, no intervention is implemented in the system and the target expected values correspond to the observational expected outcomes. When $\mathbf{X}_s = \mathbf{X}$, all variables are intervened upon except for the context variables $\mathbf{C}$ that can only be observed.

The problem given in Eq. (6.1) is challenging because of two main reasons. Firstly, the cardinality of $\mathcal{P}(\mathbf{X})$ grows exponentially with $|\mathbf{X}|$ and finding the optimal set requires, in principle, a combinatorial search. Secondly, for every set $\mathbf{X}_s \in \mathcal{P}(\mathbf{X})$, finding $\mathbf{x}_s^\star$ requires evaluating the objective function and thus implementing multiple interventions in the system at different intervention levels. In most settings, the number of function evaluations, whose cost is assumed to be given by some cost function $Co(\mathbf{X}_s, \mathbf{x}_s)$, needs to be kept low. Given a budget H, we thus want to find the optimal configuration with the minimal total cost $\sum_{h=1}^{H} Co(\mathbf{X}_h, \mathbf{x}_h)$.

6.2 Related Work

As mentioned in Section 3.1, there exists an extensive literature on BO algorithms that can be applied in various settings. Examples include multi-fidelity settings [McLeod et al., 2017; Song et al., 2019; Swersky et al., 2013], batch optimization [Alvi et al., 2019; González et al., 2016a], non-myopic optimization [González et al., 2016b] or dynamic settings [Nyikosa et al., 2018] just to name a few; see Shahriari et al. [2015] for a review. The same holds for causality. In this field, research has focused on various directions including but not limited to learning structural causal models [Goudet et al., 2018; Lucas and Griffiths, 2010; Rubenstein et al., 2017a; Silva and Gramacy, 2010], computing causal

effects from observational and/or interventional data [Alaa and Van der Schaar, 2017; Hoyer et al., 2008; Kaddour et al., 2021; Kilbertus et al., 2020; Louizos et al., 2017; Silva, 2016], discovering causal relationships [Chickering, 2002; Cooper and Yoo, 1999; Hauser and Bühlmann, 2012; Ke et al., 2019; Silva et al., 2005; Spirtes et al., 2000; Sun et al., 2007; Zheng et al., 2018] and transferring causal information across environments [Bareinboim and Pearl, 2012, 2013, 2014; Magliacane et al., 2018; Pearl and Bareinboim, 2011; Rojas-Carulla et al., 2018; Zhang et al., 2013]; see Guo et al. [2020] for a review. The literature on sequential causal decision-making is instead limited.

As discussed in the introduction, recent works have focused on causal multi-armed bandit (MAB) problems and causal reinforcement learning (RL) settings where actions or arms correspond to interventions on an arbitrary causal graph and there exist complex links between the agent's decisions and the received rewards. Bareinboim et al. [2015] and Lu et al. [2018] focus on settings with unobserved confounders. Lee and Bareinboim [2018] identify a set of possibly-optimal arms that an agent should play in order to maximize its expected reward in a MAB problem. Lee and Bareinboim [2019] extend this work to graphs with non-manipulable variables. Lattimore et al. [2016] study a specific family of MAB problems called parallel bandit problems. Finally, Ortega and Braun [2014] focus on causal discovery in Causal MAB. In the causal RL literature, Buesing et al. [2019] leverage structural causal models for counterfactual evaluation of arbitrary policies on individual off-policy episodes. Foerster et al. [2018] focus on the multi-agents setting and propose a framework in which each agent learns from a shaped reward that compares the global reward to the counterfactual reward received when that agent's action is replaced with a default action. As mentioned in Chapter 1, existing causal MAB and causal RL algorithms cannot be straightforwardly applied to solve the problems considered by CBO. Indeed, causal RL algorithms are characterized by a state variable and focus on finding an optimal policy, that is a mapping between states and actions, with the final goal of minimizing the cumulative regret. Apart from some exceptions (e.g. Lattimore et al. [2016]), cumulative regrets are also considered by causal MAB. As in standard BO, in CBO, we don't have an explicit notion of state and the goal is to find the optimum of a function in the lowest number of trials. More importantly, the actions space in both causal MAB and causal RL is generally discrete. In those cases, agents have to select the intervention variables to manipulate but not the intervention level. In causal BO, we aim at jointly identifying both the optimal intervention set and level.

6.2.1 Connections and Generalizations

Bayesian Optimization The CBO framework can be seen as a generalization of BO incorporating causal information about the system. Consider the DAG in Fig. 6.2(a). For this DAG, the problem in Eq. (6.1) can be solved resorting to BO and disregarding all causal information. In order to find the optimal intervention, BO breaks the input variables dependencies (Fig. 6.2(b)) and intervenes simultaneously on all of them thus setting $\mathbf{X}_s = \mathbf{X}$. Therefore, BO only considers one objective function $\mathbf{Y} = f(\mathbf{X})$ corresponding to the full intervention and places a GP surrogate model $p(f) = \mathcal{GP}(m, k)$ on it. As seen in Section 2.1, given a dataset $\mathcal{D}_n = \{\mathbf{x}_i, y_i\}_{i=1}^n$, the posterior distribution of f under Gaussian likelihood is also a GP with closed form posterior mean and variance. This posterior parameters are used to compute an acquisition function $\alpha(\mathbf{x}, \mathcal{D}_n)$ which is then numerically optimized to select the next evaluation (see Section 3.1 for an introduction on BO).

Causal Dimensionality It is well known [Wang et al., 2016] that the performance of standard BO algorithms deteriorates in high dimensional problems as the number of evaluations needed to find the global optimum increases exponentially with the space dimensionality. Interestingly, knowing the causal graph allows us to reason about the effective dimensionality of the problem. We formalize this idea by defining the notion of *causal intrinsic dimensionality*:

Definition 6.1. (Causal Intrinsic Dimensionality) The causal intrinsic dimensionality of a causal function $\mathbb{E}_{P(Y|\text{do}(\mathbf{X}=\mathbf{x}))}[Y]$ is given by the number of parents of the target variable, that is $|\text{Pa}(Y)|$.

For instance, in Fig. 6.2(b) the input space dimensionality is 200. However, $\mathbb{E}[Y|\text{do}(X_1, \dots, X_{100}, Z_1, \dots, Z_{100})] = \mathbb{E}[Y|\text{do}(X_{100}, Z_{100})]$, thus X_{100} and Z_{100} are the only two relevant variables and the intrinsic dimensionality of the problem is two. For the general problem in Eq. (6.1) we have $\mathbb{E}[\mathbf{Y}|\text{do}(\mathbf{X})] = \mathbb{E}[\mathbf{Y}|\text{do}(\text{Pa}(\mathbf{Y}))]$. Related to the concept of causal dimensionality, Wang et al. [2016] proposed to perform Bayesian optimization in a low-dimensional space which reflects the *intrinsic dimensionality* of a function. Provided that the objective function has low intrinsic dimensionality, Wang et al. [2016] use random embeddings to reduce the problem dimensionality without knowing which dimensions are important. This idea can be formalized and made explicit by taking a causal perspective on the optimization problem. The causal graph allows to determine not only if the function has low intrinsic dimensionality but also to identify which dimensions are important.

Causal Bandits There is a significant link between our problem setup and the settings tackled by causal MAB algorithms [Bareinboim et al., 2015]. Causal

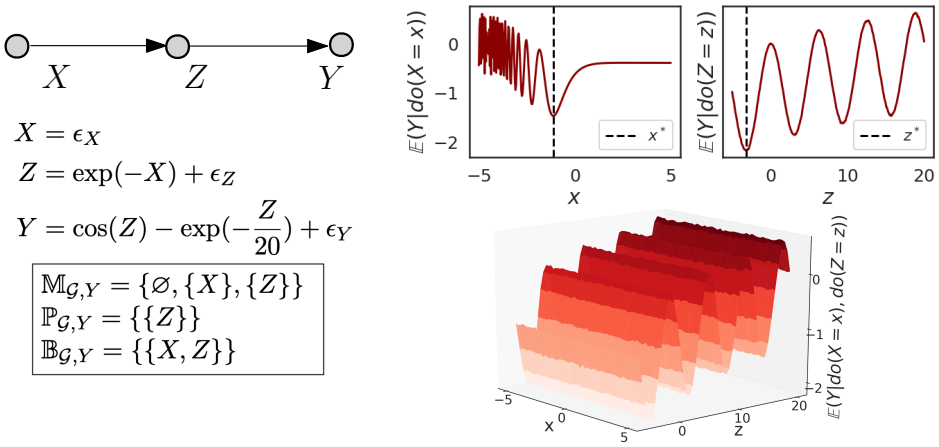


Figure 6.3: Toy example illustrating the elements of CBO. *Left*: DAG, SCM and optimal sets considered by CBO and BO. *Right*: Objective functions for different intervention sets. Notice how the intervention function for $\{X, Z\}X$ is invariant with respect to X when the value of Z is fixed. Therefore this intervention set does not need to be explored and the causal intrinsic dimensionality of the problem reduces to one.

MAB algorithms interpret decisions as interventions, target a causal effect function, and account for complex dependency structures between actions that are encoded in the causal graph. Indeed, when all intervention variables $\mathbf{X}$ are binary, the CGO setting reduces to the causal MAB setting. However, Eq. (6.1) gives a more general formulation of the problem where variables can be continuous or categorical and, more importantly, where both the intervention values and the intervention set need to be jointly determined.

6.3 Methodology

This section details a new methodology, which we call *Causal Bayesian Optimization*, addressing the problem in Eq. (6.1). The building blocks of this approach are the following:

- an exploration set (Section 6.3.1) determining a set of variables which is worth intervening on based on the topology of $\mathcal{G}$;
- a surrogate model (Section 6.3.2), called Causal GP model, that enables the integration of observational and interventional data;
- an acquisition function (Section 6.3.3) solving the exploration/exploitation trade-off *across* interventions;
- an ϵ-greedy policy (Section 6.3.4) solving the observation/intervention trade-off *within* every single intervention.

In this chapter we consider settings where a data set $\mathcal{D}^{\text{O}} = \{(\mathbf{v}^n, y^n)\}_{n=1}^N$ from an observational study is available and all causal effects in $\mathcal{G}$ are identifiable. Here $\mathbf{v}^n \in \mathbb{R}^{|\mathbf{V}|}$, $y^n \in \mathbb{R}$ and the joint distribution follows the conditional independence assumptions encoded in $\mathcal{G}$.

6.3.1 Selecting the Optimal Exploration Set

A naive approach to find $\mathbf{X}_s^\star$ would be to explore the $2^{|\mathbf{X}|}$ sets in $\mathcal{P}(\mathbf{X})$. Albeit this is a valid strategy, its complexity grows exponentially with $|\mathbf{X}|$. However, exploiting the rules of *do*-calculus and the partial orders among subsets, Lee and Bareinboim [2018] identify invariances in the interventional space and potentially optimal intervention set which we define below.

Definition 6.2. (Minimal Intervention set (MIS)) Given $\langle \mathcal{G}, \mathbf{Y}, \mathbf{X}, \mathbf{C} \rangle$, a set of variables $\mathbf{X}_s \in \mathcal{P}(\mathbf{X})$ is said to be a MIS if there is no $\mathbf{X}'_s \subset \mathbf{X}_s$ such that $\mathbb{E}[Y|\text{do}\,(\mathbf{X}_s = \mathbf{x}_s)] = \mathbb{E}[Y|\text{do}\,(\mathbf{X}'_s = \mathbf{x}'_s)]$.

We denote by $\mathbb{M}_{\mathcal{G},Y}^{\mathbf{C}}$ the set of MISs for $\langle \mathcal{G}, \mathbf{Y}, \mathbf{X}, \mathbf{C} \rangle$ where each MIS represents a set of variables that is worth intervening on. When $\mathbf{C} = \varnothing$, we use $\mathbb{M}_{\mathcal{G},Y}$. Incorporating into MIS the partial orderedness among subsets of $\mathcal{P}(\mathbf{X})$ we define the so-called POMIS.

Definition 6.3. (Possibly-Optimal Minimal Intervention set (POMIS)) Given $\langle \mathcal{G}, \mathbf{Y}, \mathbf{X}, \mathbf{C} \rangle$, let $\mathbf{X}_s \in \mathbb{M}_{\mathcal{G},Y}^{\mathbf{C}}$. $\mathbf{X}_s$ is a POMIS if there exists a SCM conforming to $\mathcal{G}$ such that $\mathbb{E}[Y|\text{do}\,(\mathbf{X}_s = \mathbf{x}^*)] > \forall_{\mathbf{W} \in \mathbb{M}_{\mathcal{G},Y}^{\mathbf{C}} \setminus \mathbf{X}_s} \mathbb{E}[Y|\text{do}\,(\mathbf{W} = \mathbf{w}^*)]$ where $\mathbf{x}^*$ and $\mathbf{w}^*$ denote the optimal intervention values.

We denote by $\mathbb{P}_{\mathcal{G},Y}^{\mathbf{C}}$ the set of POMIS for $\langle \mathcal{G}, \mathbf{Y}, \mathbf{X}, \mathbf{C} \rangle$ where each POMIS represents a variable on which intervening always improves Y with respect to the remaining elements in $\mathbb{M}_{\mathcal{G},Y}^{\mathbf{C}}$. For completeness, we also use $\mathbb{B}_{\mathcal{G},Y}^{\mathbf{C}}$ to denote the unique set on which BO performs interventions that includes all manipulative variables $\mathbf{X}$. In Fig. 6.3 we give an example in which $|\mathbb{M}_{\mathcal{G},Y}| < |\mathcal{P}(\mathbf{X})|$ and intervening on $\mathbb{B}_{\mathcal{G},Y}$ is suboptimal. Indeed, Fig. 6.3 shows how $\mathbb{E}[Y|\text{do}\,(X = x)\,, \text{do}\,(Z = z)] = \mathbb{E}[Y|\text{do}\,(Z = z)]$ and the causal intrinsic dimensionality of this problem is $|\text{Pa}(Y)| = 1$. In addition, $\mathbb{E}[Y|\text{do}\,(X = x^*)] > \mathbb{E}[Y|\text{do}\,(Z = z^*)]$ thus Z is optimal with respect to X and is the only set in $\mathbb{P}_{\mathcal{G},Y}$. For notational convenience, we refer to the exploration set, which can be $\mathbb{M}_{\mathcal{G},Y}^{\mathbf{C}}$ or $\mathbb{P}_{\mathcal{G},Y}^{\mathbf{C}}$, as **ES**. The choice of the **ES** depends on the causal graph and the next sections are agnostic to the choice of the **ES**.

6.3.2 Causal GP Model

To integrate experimental and observational data, for each $\mathbf{X}_s \in$ **ES**, we place a GP prior on $f_s(\mathbf{x}) = \mathbb{E}[Y|\text{do}\,(\mathbf{X}_s = \mathbf{x})]$ with prior mean and kernel function

computed via *do*-calculus:

$$f_s(\mathbf{x}) \sim \mathcal{GP}(m_s(\mathbf{x}), k_s(\mathbf{x}, \mathbf{x}')) \tag{6.2}$$

$$m_s(\mathbf{x}) = \hat{\mathbb{E}}[Y|\text{do}\,(\mathbf{X}_s = \mathbf{x})] \tag{6.3}$$

$$k_s(\mathbf{x}, \mathbf{x}') = k_{RBF}(\mathbf{x}, \mathbf{x}') + \sigma_s(\mathbf{x})\sigma_s(\mathbf{x}') \tag{6.4}$$

where $\sigma_s(\mathbf{x}) = \sqrt{\hat{\mathbb{V}}(Y|\text{do}\,(\mathbf{X}_s = \mathbf{x}))}$ with $\hat{\mathbb{V}}$ representing the variance estimated from observational data and k_{RBF} representing the radial basis function kernel defined as $k_{RBF}(\mathbf{x}, \mathbf{x}') := \exp(-\frac{||\mathbf{x}-\mathbf{x}'||^2}{2l^2})$. Fig. 6.4 illustrates the difference between the posterior GP distribution obtained with a zero mean prior distribution and RBF kernel (lower plot) and with the proposed causal prior (upper plot). Alternative kernel functions, e.g. a non stationary kernel to capture the behaviour of the intervention function for X in Fig. 6.3, could be easily combined with the additional vartiance term. In terms of computations, both the prior mean function $m_s(\mathbf{x})$ and the variance adjustment term $\sigma_s(\mathbf{x})$ can be obtained by estimating the conditional densities required by the do-calculus using observational data and then approximating the intractable integrals via Monte Carlo integration. Overall, the causal prior mean function captures the behaviour of the target function in areas where observational data is available (crosses at the bottom) despite the lack of interventional data. In addition, the causal posterior variance is higher in areas where $\sigma_s(\mathbf{x})$ is inflated due to the lack of observations, which enables proper uncertainty quantification around the causal effects in a system.

However, note that in this work we estimate the kernel hyperparameters from the data by maximizing the marginal likelihood via the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm. While this facilitates inference as posteriors can be derived in closed form, it suffers from two main issues. On the one hand, using point estimates of hyperparameters yields overconfident predictions, by failing to account for hyperparameters' uncertainty. On the other hand, the non-convexity of the marginal likelihood surface can lead to poor estimates located at local minima. Multiple starting points for the hyperparameters optimization should be used to alleviate this issue. Extending the Bayesian treatment to hyperparameters in a hierarchical framework leads to intractable posterior and thus requires resorting to approximate inference methods, see e.g. Lalchand and Rasmussen [2020] for an example of a fully bayesian GP regression.

6.3.3 Causal Acquisition Function

For each $\mathbf{X}_s \in \mathbf{ES}$, we define the acquisition function as the expected improvement (EI) with respect to the current best observed interventional setting across all sets in $\mathbf{ES}$. Indeed, using the EI as an acquisition function ensures computational tractability. In addition, the standard EI formulation can be easily extended to compare improvements over multiple surrogate models. Alternatively, one could develop causal versions of mutual information-based acquisition functions such as entropy search [Hennig and Schuler, 2012] or max-value entropy search [Wang and Jegelka, 2017]. However, this would require defining a distribution over the global optimal input or output which would significantly complicate derivations.

At every step of the optimization, denote by $y_s = \mathbb{E}[Y|\text{do}\,(\mathbf{X}_s = \mathbf{x})]$ and $y^\star$ the optimal value of y_s, $s = 1, \ldots, |\mathbf{ES}|$ observed so far. The EI is given by:

$$\text{EI}^s(\mathbf{x}) = \mathbb{E}_{p(y_s)}[\max(y_s - y^\star, 0)]/Co(\mathbf{X}_s, \mathbf{x}). \tag{6.5}$$

Let $\alpha_1, \ldots, \alpha_{|\mathbf{ES}|}$ be solutions of the optimization of $\text{EI}^s(\mathbf{x})$ for each set in $\mathbf{ES}$ and $\alpha^\star := \max\{\alpha_1, \ldots, \alpha_{|\mathbf{ES}|}\}$. The best new intervention set is given by $s^\star = \text{argmax}_{s \in \{1,\cdots,|\mathbf{ES}|\}} \alpha_s$. Therefore, the set-value pair to intervene on is $(s^\star, \alpha^\star)$. Fig. 6.5 shows the acquisition functions for $\mathbf{ES} = \mathbb{M}^{\mathbf{C}}_{\mathcal{G},Y}$ in the toy example. The new intervention is selected by comparing the maxima of the acquisition functions across interventions (red and back dots).

6.3.4 ϵ-greedy Policy

For some graph structures, such as the one in Fig. 6.2, the empty set, which represents the observational case, is part of $\mathbf{ES}$. A mechanism in the optimization process is then needed to observe the system when that is the optimal strategy. One could take a Bayesian approach and decide whether to observe the system by reasoning about the values we could observe if this was our selected action. To do that, we would simulate observations given our current posterior distributions on the functions in the SCM and get an estimate of the expected observational output. However, if the variances of our current posterior distributions on the SCM functions are large or if our likelihoods for the SCM are wrong, we might end up under exploring the system and observing it for a high number of trials before revising our functions estimates thus potentially intervening and realising higher outputs can be achieved by performing interventions. Therefore, inspired by the ϵ-greedy policies in RL [Tokic, 2010], we propose an ϵ-greedy approach where ϵ determines the probability of observing the system instead of intervening. The value of ϵ, which is a parameter of CBO, can be selected

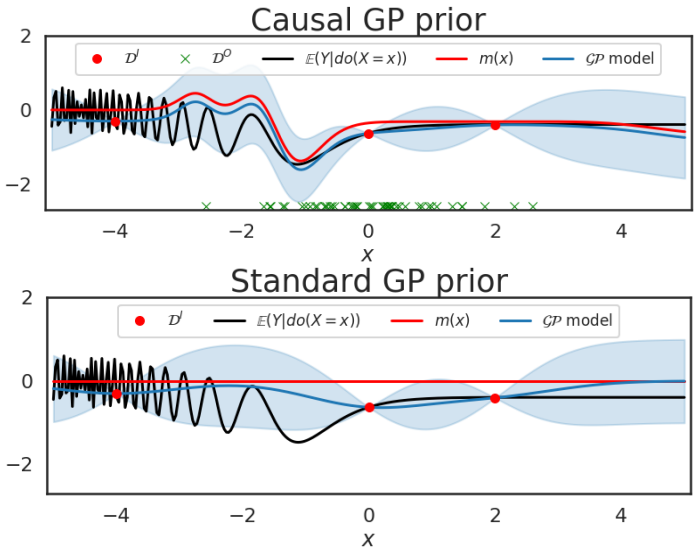


Figure 6.4: Posterior GP obtained with two different prior formulations. *First row:* Posterior distribution associated to the Causal GP prior which integrates both the interventional data (red dots) and the observational data (green crosses). *Second row:* Posterior distribution associated to a GP prior with zero mean and RBF kernel. In this case the GP model only considers interventional data (red dots) thus not capturing the true function in areas where observational data are available, e.g. the interval $[-2, 0]$.

in several different ways and needs to balance the *observation-intervention trade-off* emerging in CBO. On the one hand, collecting observational data allows to reliably estimate causal effects via *do*-calculus. On the other hand, computing consistent causal effects for values outside of the observational range, requires intervening. The agent needs to find the right combination of the two actions so as to exploit observational information while intervening in areas where the uncertainty is higher. Here we define ϵ as:

$$\epsilon = \frac{\mathsf{Vol}(\mathcal{C}(\mathcal{D}^O))}{\mathsf{Vol}(\times_{X \in \mathbf{X}}(D(X)))} \times \frac{N}{N_{\max}}, \tag{6.6}$$

where $\mathsf{Vol}(\mathcal{C}(\mathcal{D}^O))$ represents the volume of the convex hull for the observational data and $\mathsf{Vol}(\times_{X \in \mathbf{X}}(D(X)))$ gives the volume of the interventional domain, see Fig. 6.6 for a visualization of the convex hull for the example of Fig. 6.3. $N_{\max}$ represents the maximum number of observations the agent is willing to collect and N is the current size of $\mathcal{D}^O$. When $\mathsf{Vol}(\mathcal{C}(\mathcal{D}^O))$ is small with respect to N, the interventional space is bigger than the observational space. We thus intervene and explore part of the interventional space not covered by $\mathcal{D}^O$. On the contrary, if $\mathsf{Vol}(\mathcal{C}(\mathcal{D}^O))$ is large with respect to N, we obtain consistent estimates of the causal effects by collecting more observations and computing them via the *do*-calculus. We thus observe and update the prior GP in Eqs. (6.3) - (6.4). Other ϵ-greedy policies can be formulated in order to

solve the trade-off differently. For instance, the agent could define an adaptive ϵ which favours observations in the first stages of the optimization procedure and interventions as N increases. Alternatively, the value of ϵ could depend on the agent's budget and favours interventions when their cost is low. Finding the optimal ϵ-greedy policy is left as an open research direction.

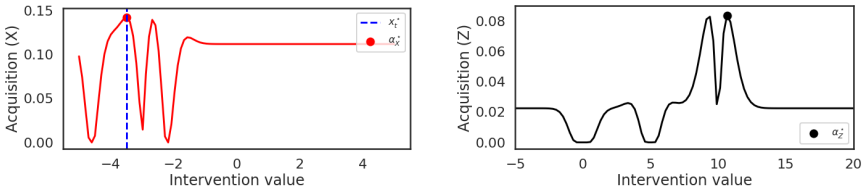


Figure 6.5: Toy example. Acquisition functions for the variables in $\mathbb{M}^{\mathbf{C}}_{\mathcal{G},Y}$. Each surrogate model is associated to an acquisition function. The maxima across different functions (red and black dots) are compared to select the next function evaluation. In this plot, the dashed blue line gives the next optimal evaluation which corresponds to an intervention on X.

6.3.5 The CBO Algorithm

We give the complete CBO algorithm in Algorithm 2. The time complexity of CBO is dominated by algebraic operations on the kernel matrix $k_s(\mathbf{x}, \mathbf{x}')$ which are $\mathcal{O}(H^3)$ where H denotes the number of function evaluations of the BO algorithm. The space complexity is also dominated by storing $k_s(\mathbf{x}, \mathbf{x}')$ which is $\mathcal{O}(H^2)$. Both the time and space complexities can be improved by resorting to inducing point approximations (see Section 2.1.2 for an introduction). Given the acquisition function and the surrogate model, the theoretical guarantees of CBO are limited and follow directly from the theoretical properties of *do*-calculus. However, one could extend CBO to use a GP-UCB acquisition function for which a cumulative regret bound has been derived [Srinivas et al., 2012]. Adapting

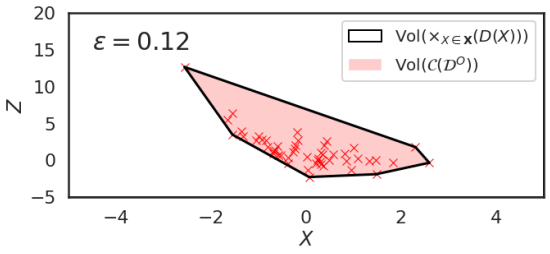


Figure 6.6: Toy example. Convex hull in the X-Z computed considering the observational dataset $\mathcal{D}^O$ represented by the red crosses. The boundaries of the plot correspond to the interventional domain. The rescaled ratio between the volume of the convex hull (red shaded area) and the volume of the interventional domain (white area) gives the ϵ value used to select observation vs intervention.

other acquisition functions such as the GP-UCB to the causal setting remains an interesting open challenge.

6.4 Experiments

We test CBO on two synthetic settings and on two real-world applications for which a DAG is available and can be used as a simulator. We run CBO to explore both $\mathbb{M}^{\mathbf{C}}_{\mathcal{G},Y}$ and $\mathbb{P}^{\mathbf{C}}_{\mathcal{G},Y}$ and show how the optimal intervention set, intervention values, and cost incurred to achieve the optimum change depending on the DAG and the SCM. For all variables in $\mathbf{X}$ and their combinations, we assume to have data from previous interventions which we denote by $\mathcal{D}^{\mathrm{I}} = \{(\mathbf{x}^i, \mathbb{E}\left[Y|\mathrm{do}\left(\mathbf{X}^i_s = \mathbf{x}^i\right)\right])\}_{i=1,s=1}^{N^I_s,|\mathbf{ES}|}$. Typically the number of interventional outputs observed for each set, that is N^I_s, is very small and future interventions are prohibitive to implement. Code and data for all the experiments is provided at `https://github.com/VirgiAgl/CBO`.

Baselines We compare CBO against a standard BO algorithm, in which all variables are intervened upon, and a CBO version where a standard GP prior given by $p(f_s(\mathbf{x})) = \mathcal{GP}(0, k_{\mathrm{RBF}}(\mathbf{x}, \mathbf{x}'))$ is used.

Algorithm 2 CBO

Inputs: $\mathcal{D}^{\mathrm{O}}$, $\mathcal{D}^{\mathrm{I}}$, $\mathcal{G}$, **ES**, number of steps H.
Output: $\mathbf{X}^\star_s, \mathbf{x}^\star_s, \hat{\mathbb{E}}[\mathbf{Y}^\star|\mathrm{do}\,(\mathbf{X}^*_s = \mathbf{x}^\star_s)]$
Initialize: Set $\mathcal{D}^{\mathrm{I}}_0 = \mathcal{D}^{\mathrm{I}}$ and $\mathcal{D}^{\mathrm{O}}_0 = \mathcal{D}^{\mathrm{O}}$
for `h=1 to H` **do**
    Compute ϵ and sample $u \sim \mathcal{U}(0,1)$
    if $\epsilon > u$ **then**
        (Observe)
        1. Observe new observations $(\mathbf{x}_h, c_h, \mathbf{y}_h)$.
        2. Augment $\mathcal{D}^{\mathrm{O}} = \mathcal{D}^{\mathrm{O}} \cup \{(\mathbf{x}_h, c_h, \mathbf{y}_h)\}$.
        3. Update prior of the causal GP (Eq. (6.2))
    else
        (Intervene)
        1. Compute $\mathrm{EI}^s(\mathbf{x})$ for each element $\mathbf{X}_s \in \mathbf{ES}$ (Eq. (6.5)).
        2. Obtain the optimal interventional set-value pair $(s^\star, \alpha^\star)$.
        3. Intervene on the system.
        4. Update posterior of the causal GP.
    end if
end for
Return the optimal intervention couple $(\mathbf{X}^\star_s, \mathbf{x}^\star_s)$ and corresponding output.

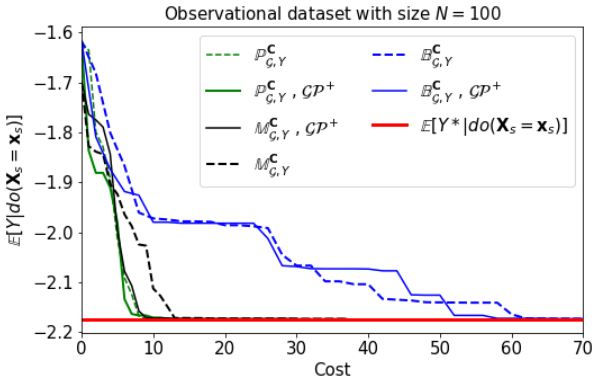


Figure 6.7: Toy example. Convergence of CBO and standard BO across different initializations of $\mathcal{D}^I$. The red line gives the optimal Y^* when intervening on sets in $\mathbb{M}^{\mathbf{C}}_{\mathcal{G},Y}$, $\mathbb{P}^{\mathbf{C}}_{\mathcal{G},Y}$ or $\mathbb{B}^{\mathbf{C}}_{\mathcal{G},Y}$. Solid lines give CBO results when using the causal GP model which is denoted by $\mathcal{GP}^+$. Dotted lines correspond to CBO with a standard GP prior model $p(f(\mathbf{x}_s)) = \mathcal{GP}(0, k_{\text{RBF}}(\mathbf{x}_s, \mathbf{x}'_s))$. See Fig. C.1 in the supplement for standard deviations.

Performance measures We run CBO with different initializations of $\mathcal{D}^{\text{I}}$ and report the average convergence performances together with standard errors. In the synthetic setting, we consider three different cost configurations: equal unit cost per node, different fix costs per node, and variable costs per node. The total cost at each optimization step is computed as the sum of the cost for each intervened node. We show the results for equal unit cost per node and report the full comparison in the supplement.

6.4.1 Toy Experiment

We show the convergence results for CBO and competing algorithms for the toy example described in the text. For this experiment we set $N = 100$ and $N^I_s = 3$ for all $\mathbf{X}_s \in \mathbf{ES}$. Given the SCM in Fig. 6.3 (left panel), the optimal configuration is $(X^\star_s, x^\star_s) = (Z, -3.20)$. CBO converges to the optimum faster than BO which requires intervening on all nodes and it is thus twice more expensive (Fig. 6.7).

6.4.2 Synthetic Experiment

We test the algorithm on the DAG given in Fig. 6.2(a). This DAG includes unobserved confounders, non-manipulative variables and requires to apply both front-door and back-door adjustment formulas to estimate the causal effects. We set $N^I_s = 10$ for all $\mathbf{X}_s \in \mathbf{ES}$ and test different values of N. The exploration set for BO is given by $\mathbb{B}^{\mathbf{C}}_{\mathcal{G},Y} = \{B, D, E\}$, while for CBO we have $\mathbb{M}^{\mathbf{C}}_{\mathcal{G},Y} = \{\varnothing, \{B\}, \{D\}, \{E\}, \{B, D\}, \{B, E\}, \{D, E\}\}$, and $\mathbb{P}^{\mathbf{C}}_{\mathcal{G},Y} = \{\varnothing, \{B\}, \{D\}, \{E\}, \{B, D\}, \{D, E\}\}$. All the intervention sets in $\mathbb{M}^{\mathbf{C}}_{\mathcal{G},Y}$ and $\mathbb{P}^{\mathbf{C}}_{\mathcal{G},Y}$ include a maximum of two variables. On the contrary, BO only considers

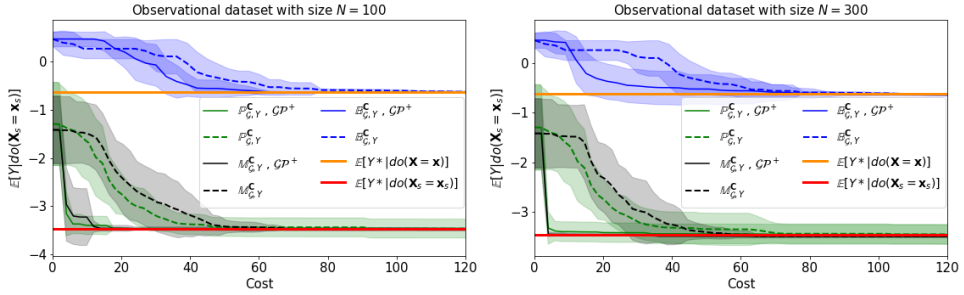


Figure 6.8: Synthetic example. Convergence of CBO and BO across different initialization of $\mathcal{D}^I$. The orange line gives the optimal Y^* when intervening on $\mathbb{B}^{\mathbf{C}}_{\mathcal{G},Y}$. The red line gives the optimal Y^* when intervening on sets in $\mathbb{M}^{\mathbf{C}}_{\mathcal{G},Y}$ or $\mathbb{P}^{\mathbf{C}}_{\mathcal{G},Y}$. Solid lines give CBO results when using the causal GP model, denoted by $\mathcal{GP}^+$, while dotted lines correspond to CBO with a standard GP prior model. Shaded areas are $\pm$ standard deviation.

interventions on three variables thus increasing the dimensionality of the problem by one. The SCM, the *do*-calculus computations and further details about this experiment are given in Appendix C.2 - Appendix C.5.

Fig. 6.8 shows how CBO outperforms standard BO and achieves the best performance when the causal GP model is used. There are two main reasons why BO leads to a suboptimal solution. Firstly, notice that the causal effect when intervening on all variables is equal to $\mathbb{E}[Y|\text{do}\,(B = b)\,, \text{do}\,(D = d)\,, \text{do}\,(E = e)] = \mathbb{E}[Y|\text{do}\,(D = d)\,, \text{do}\,(E = e)]$. This means that the same outcome can be achieved by only intervening on $\{D, E\}$ at a significantly lower cost. Secondly, although it may seem counter-intuitive, intervening only on a subset of variables leads to better outcomes. Manipulating all variables breaks the causal links between them and blocks the propagation of causal effects in the graph. In this example, intervening on B, E, D blocks the causal effect of B on Y. Manipulating only B leads to a propagation of its causal effect through D and E. Given the SCM, $\mathbb{E}[Y|\text{do}(B, = b, D = d, E = e)] < \mathbb{E}[Y|\text{do}(B, = b, D = d)]$ for each $b \in D(B)$, $d \in D(D)$ and $e \in D(E)$. Indeed, setting the level of B makes D and E take values outside of their interventional domains $D(D)$ and $D(E)$ thus leading to function values not achievable in BO. Furthermore, the causal GP prior determines the locations of the function evaluations thus reducing the number of steps required to find the optimum. As expected, the benefit of incorporating $\mathcal{D}^{\text{O}}$ into the prior becomes more evident when N increases. The optimal configuration for this setting is $(\mathbf{X}^\star_s, \mathbf{x}^\star_s) = (\{B, D\}, (-5.0, 3.28))$.

6.4.3 Example in Ecology

We apply CBO to a large-scale optimization problem in ecology. We consider the issue of maximizing the net coral ecosystem calcification (NEC) in the Bermuda given a set of environmental variables. The causal graph (Fig. C.3(b)

in the supplement) is taken from Courtney et al. [2017] and modified so as to avoid directed cycles. We consider a subset of 5 variables as manipulative, that is $\mathbf{X} = \{\text{Nut}, \Omega_A, Chl\alpha, \text{TA}, \text{DIC}\}$, and assume to be able to intervene *contemporaneously* on a maximum of 3 variables. Given these assumptions and the DAG, $\mathbb{M}^{\mathbf{C}}_{\mathcal{G},Y}$ includes the single variable interventions and all the 2 and 3 variables interventions that can be performed selecting variables in $\mathbf{X}$. The cardinality of $\mathbb{M}^{\mathbf{C}}_{\mathcal{G},Y}$ is thus 25. Notice that the size of $\mathbb{B}^{\mathbf{C}}_{\mathcal{G},Y} = \{\text{Nut}, \Omega_A, Chl\alpha, \text{TA}, \text{DIC}\}$ is greater than 3 thus BO is not a viable strategy for this application. We first construct a simulator by fitting a linear SCM with the 50 observations provided by Andersson and Bates [2018]. We then use the simulator to generate $N = 500$ observations and $N^I_s = 1$ initial interventional data points for all $\mathbf{X}_s \in \mathbf{ES}$. We set the interventional domains to $D(\text{Nut}) = [-2, 5]$, $D(\Omega_A) = [2, 4]$, $D(Chl\alpha) = [0.3, 0.4]$, $D(\text{TA}) = [2200, 2550]$ and $D(\text{DIC}) = [1950, 2150]$. We run CBO on $\mathbb{M}^{\mathbf{C}}_{\mathcal{G},Y}$, with and without the causal GP prior. We found CBO to successfully explore $\mathbb{M}^{\mathbf{C}}_{\mathcal{G},Y}$, especially when the causal GP prior is used (Fig. 6.9). The optimal intervention for this experiment is given by $(\mathbf{X}^\star_s, \mathbf{x}^\star_s) = (\{\Omega_A, \text{TA}, \text{DIC}\}, (2, 2550, 1950))$.

6.4.4 Example in Healthcare

Finally, we apply our method to an example in healthcare. The DAG (Fig. C.3(a) in the supplement) is taken from Thompson [2019] and Ferro et al. [2015] and is used to model the causal effect of statin drugs on the levels of prostate specific antigen (PSA). Our goal is to minimize PSA by intervening on statin and aspirin usage. $\mathcal{D}^O$ consists of $N = 500$ instances sampled from the simulator while $N^I_s = 3$ for all $\mathbf{X}_s \in \mathbf{ES}$. Given the causal structure, $\mathbb{M}^{\mathbf{C}}_{\mathcal{G},Y} = \{\varnothing, \{\text{aspirin}\}, \{\text{statin}\}, \{\text{aspirin}, \text{statin}\}\}$ while $\mathbb{P}^{\mathbf{C}}_{\mathcal{G},Y} = \{\{\text{aspirin}, \text{statin}\}\}$. We set the domain $D(\text{aspirin}) = D(\text{statin}) = [0.0, 1.0]$ and run CBO on both $\mathbb{M}^{\mathbf{C}}_{\mathcal{G},Y}$ and $\mathbb{P}^{\mathbf{C}}_{\mathcal{G},Y}$. We found the optimal intervention to be the couple $(\mathbf{X}^\star_s, \mathbf{x}^\star_s) = (\{\text{aspirin}, \text{statin}\}, (0.0, 1.0))$ which is consistent with domain knowledge [Algotar et al., 2010]. This experiment shows how CBO can help doctors, and decision-makers in general, to find optimal interventions in real-life scenarios based on simulators and thus avoiding expensive and invasive interventions.

6.5 Conclusions and Discussion

This chapter formalizes the problem of globally optimizing a variable that is part of a causal model in which a sequence of interventions can be performed. We propose a Causal Bayesian Optimization (CBO) algorithm which solves the global optimization problem by exploring a set of potentially optimal sets defined on a causal graph. This is achieved via a causal expected improvement

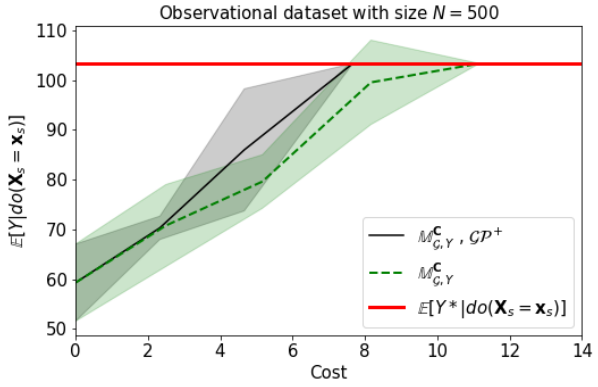


Figure 6.9: NEC example. Convergence of CBO across different initialization of interventional data $\mathcal{D}^I$ and with and without causal GP prior. The red line gives the optimal $Y^\star$ when intervening on $\mathbb{M}^{\mathbf{C}}_{\mathcal{G},Y}$.

acquisition function and an ϵ-greedy policy solving the emerging observation-intervention trade-off together with the well-known exploration-exploitation trade-off. In addition, we formulate a causal GP model which allows us to integrate observational and interventional data via the *do*-calculus and properly calibrates uncertainty around the causal effects. We show the benefits of the proposed approach in a variety of settings, both synthetic and real, characterized by different causal graph structures. Our results demonstrate how CBO outperforms BO and reaches the global optimum after a significantly lower number of optimization steps.

We can identify different CBO limitations. Firstly, CBO requires placing a different GP on the causal effect of each intervention set we want to explore. This means that the number of surrogate models becomes prohibitive when the number of nodes in the causal graph increases. In addition, having multiple single-task models limits the transfer of information across interventions. We will address this limitation in Chapter 7 through a multi-task formulation that allows capturing the correlation structure among different causal effects, similar to the model formulation we have developed in Chapter 4 in the context of Poisson point processes. We will see how using a multi-task formulation speeds up the optimization while reducing the number of causal GPs used by CBO. Another important limitation of CBO is the lack of a time dependency structure among variables. In real settings, the distribution of both the output and the input variables might change over time, often in a non-stationary manner, thus modifying the optimal intervention dynamically, both in terms of intervention set and intervention value. This aspect will be the focus of Chapter 8. Finally, CBO assumes full knowledge of the causal graph which is often an unrealistic assumption in real-world settings. Combining the proposed framework with a causal discovery algorithm remains an important open problem that we are

currently investigating in Branchini et al. [2022]. Branchini et al. [2022] offers a framework for joint optimization and causal discovery that properly accounts for uncertainty in the graph structure. In addition, it provides an acquisition function that selects interventions that are useful in jointly identifying the optimal configuration and the true underlying graph.

Chapter 7

Multi-task Causal Learning with Gaussian Processes

As seen in the previous chapter, solving sequential decision-making problems in a variety of domains such as healthcare, systems biology, or operations research, often requires experimentation. By performing interventions one can understand how a system behaves when an action is taken and thus infer the cause-effect relationships of a phenomenon. Experiments are especially useful when observational causal inference methods do not provide an accurate estimation of the causal effects. For instance, in healthcare, drugs are tested in randomized clinical trials before commercialization. Biologists might want to understand how genes interact in a cell once one of them is knocked out. Finally, engineers investigate the impact of design changes on complex physical systems by conducting experiments on digital twins [Ye et al., 2019]. Experiments in these scenarios are usually expensive, time-consuming, and, especially for field experiments, they may present ethical issues. Therefore, researchers generally have to trade-off cost, time, and other practical considerations to decide which experiments to conduct, if any, to learn about the system's behaviour.

Consider the causal graph in Fig. 7.1 which describes how crop yield Y is affected by soil fumigants X and the level of eel-worm population at different times $\mathbf{Z} = \{Z_1, Z_2, Z_3\}$ [Cochran and Cox, 1957; Pearl, 1995]. By performing a set of experiments, the investigator aims at learning the *intervention functions* relating the expected crop yield to each possible intervention set and level. Naïvely, one could achieve that by modelling each intervention function separately. This is indeed the approach taken in CBO (see Chapter 6) where each causal effect is modelled via a single-task causal GP. Similar to what is seen in Chapter 4 in the context of Poisson point processes, using single-task models might decrease the algorithm performance while increasing the computational complexity of the problem. Indeed, separate single-task models would disregard

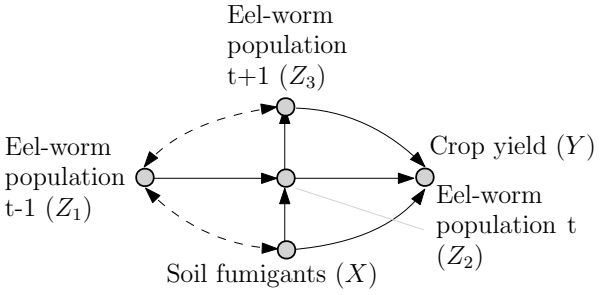


Figure 7.1: DAG for the crop yield. Nodes denote variables, arrows represent causal effects and dashed edges indicate unobserved confounders.

the correlation structure existing among a set of functions, which in Chapter 4 were intensity functions, and here are experimental outputs. In addition, as discussed in Section 6.5, in CBO the number of required GP models increases exponentially with the size of the causal graph. Finally, single-task models disregard the causal information each experiment carries about the yield we would obtain by performing alternative interventions in the graph. For instance, observing the yield when running an experiment on the *intervention set* $\{X, Z_1\}$ and setting the value to the *intervention value* $\{x, z_1\}$, provides information about the yield we would get from intervening only on X or on $\{X, Z_1, Z_2, Z_3\}$. In this chapter, we study how to jointly model such intervention functions so as to transfer knowledge across different experimental setups and integrate observational and interventional data. The model proposed here enables proper uncertainty quantification of the causal effects thus allowing the definition of optimal experimental design strategies.

The framework proposed in this chapter combines causal inference with multi-task learning via Gaussian processes (GP). Probabilistic causal models are commonly used in disciplines where explicit experimentation may be difficult and the *do*-calculus (see Section 3.2.3 for an introduction) enables prediction of the effect of an intervention without performing the experiment. In the *do*-calculus, different intervention functions are modelled individually and there is no information shared across experiments. Modelling the correlation across experiments is crucial, especially when the number of observational data points is limited and experiments on some variables cannot be performed. Multi-task GP methods have been extensively used to model non-trivial correlations between outputs (see Chapter 4, Section 2.2 and Álvarez et al. [2012] for a review). However, to the best of our knowledge, this is the first study focusing on intervention functions, possibly of different dimensionality, defined on a causal graph. Particularly, we make the following contributions:

Theoretical results on multi-task causal model We give theoretical results detailing *when* and *how* a causal multi-task model for the experimental outputs can be developed depending on the topology of the DAG.

DAG-GP model Exploiting our theoretical results, we develop a joint probabilistic model for all intervention functions, henceforth named DAG-GP, which flexibly accommodates different assumptions in terms of data availability – both observational and interventional.

Experimental comparison on different decision-making algorithms We demonstrate how DAG-GP achieves the best fitting performance in a variety of experimental settings, both synthetic and real, while enabling proper uncertainty quantification and thus optimal decision making when used within Active Learning (AL) and Bayesian Optimization (BO).

7.1 Problem setup

We consider a SCM as defined in Definition 3.1 of Section 3.2 and the associated directed causal acyclic graph (DAG) denoted by $\mathcal{G}$[1]. Within the complete set of variables in the SCM, we distinguish between two different types of variables: treatment variables $\mathbf{X}$ that can be manipulated and set to specific values and the output variable Y that represents the agent's outcome of interest[2]. As done in Chapter 6, we denote the *interventional distribution* for two disjoint sets in $\mathbf{V}$, say $\mathbf{X}$ and Y, as $P(Y|\text{do}\,(\mathbf{X}=\mathbf{x}))$. This is the distribution of Y obtained by intervening on $\mathbf{X}$ and fixing its value to $\mathbf{x}$ in the data generating mechanism, irrespective of the values of its parents. The interventional distribution differs from the *observational distribution* which is denoted by $P(Y|\mathbf{X}=\mathbf{x})$. In this chapter, we assume the causal effect for $\mathbf{X}$ on Y to be identifiable $\forall \mathbf{X} \in \mathcal{P}(\mathbf{X})$ with $\mathcal{P}(\mathbf{X})$ denoting the power set of $\mathbf{X}$. When this is the case (see Galles and Pearl [1995] for the set of identifiability conditions given a causal graph), *do*-calculus allows the estimation of interventional distributions and thus causal effects from observational distributions [Pearl, 1995]. However, the *do*-calculus involves computing integrals which are generally not tractable. When this is the case, observational data can be used to get a Monte Carlo estimate, e.g. $\hat{P}(\mathbf{Y}|\text{do}\,(\mathbf{X}=\mathbf{x})) \approx P(\mathbf{Y}|\text{do}\,(\mathbf{X}=\mathbf{x}))$, which is consistent when the number of samples drawn from $P(\mathbf{V})$ is sufficiently large.

[1] In this chapter we assume $\mathcal{G}$ to be known. However, one could run a causal discovery algorithm as a pre-processing step or use interventional data to discriminate among graphs within the Markov equivalence class.

[2] This setting can be extended to include non-manipulative variables as done in Chapter 6. See Lee and Bareinboim [2019] for a definition of such nodes.

Goal Consider a DAG $\mathcal{G}$ and the related SCM. Define the set of intervention functions for Y in $\mathcal{G}$ as:

$$\mathbf{T} = \{t_s(\mathbf{x})\}_{s=1}^{|\mathcal{P}(\mathbf{X})|} \qquad t_s(\mathbf{x}) = \mathbb{E}_{p(Y|\text{do}(\mathbf{X}_s=\mathbf{x}))}[Y] = \mathbb{E}[Y|\text{do}\,(\mathbf{X}_s = \mathbf{x})] \quad (7.1)$$

with $\mathbf{X}_s \in \mathcal{P}(\mathbf{X})$ where $\mathcal{P}(\mathbf{X})$ is the power set of $\mathbf{X}$ minus the empty set[3] and $\mathbf{x} \in D(\mathbf{X}_s)$ where $D(\mathbf{X}_s) = \times_{X\in\mathbf{X}_s} D(X)$ with $D(X)$ denoting the *interventional domain* of X. Let $\mathcal{D}^O = \{\mathbf{x}_n, y_n\}_{n=1}^N$, with $\mathbf{x}_n \in \mathbb{R}^{|\mathbf{X}|}$ and $y_n \in \mathbb{R}$, be an observational dataset of size N from this SCM. Consider an interventional dataset $\mathcal{D}^I = (\mathbf{X}^I, \mathbf{Y}^I)$ with $\mathbf{X}^I = \bigcup_s \{\mathbf{x}_s^i\}_{i=1}^{N_s^I}$ and $\mathbf{Y}^I = \bigcup_s \{y_s^i\}_{i=1}^{N_s^I}$ denoting the intervention levels and the function values observed from previously run experiments across sets in $\mathcal{P}(\mathbf{X})$. N_s^I represents the number of experimental outputs observed for the intervention set $\mathbf{X}_s$. Our goal is to define a joint prior distribution $p(\mathbf{T})$ and compute the posterior $p(\mathbf{T}|\mathcal{D}^I)$ so as to make probabilistic predictions for $\mathbf{T}$ at some unobserved intervention sets and levels.

7.2 Related work

While there exists an extensive literature on multi-task learning with GPs [Bonilla et al., 2007; Álvarez et al., 2012] and causality [Guo et al., 2020; Pearl, 2009b], the literature on causal multi-task learning and causal decision-making is very limited. Here we will review the closest works within these two fields.

Causal multi-task models In the causality literature, studies have focused on observational causal inference and have investigated the problem of transferring the causal effect of one given variable *across* environments [Bareinboim and Pearl, 2012, 2013, 2014; Pearl and Bareinboim, 2011]. Several works have focused on domain adaptation problems [Magliacane et al., 2018; Rojas-Carulla et al., 2018; Zhang et al., 2013] where data for a source domain is given, and the task is to predict the distribution of a target variable in a target domain. Closer to our work in this chapter, Alaa and Van der Schaar [2017] have developed a linear coregionalization model for learning the individual treatment effects via observational data. While Alaa and Van der Schaar [2017] is the first paper conceptualizing causal inference as a multi-task learning problem, its focus is on modelling the correlation across intervention levels for a single intervention function. In addition, the model is developed for a dichotomous intervention variable. Finally, Lee et al. [2020] studied the problem of the identification of the causal effect of one intervention set in terms of available observational *and* experimental distributions. While one could repeat their procedure for all possible intervention sets in the causal graph, Lee et al. [2020] does not allow

[3] We exclude the empty set as it corresponds to the observational distribution $t_\varnothing(\mathbf{x}) = \mathbb{E}[Y]$.

to express all causal effects via a shared interventional distribution which is instead the focus of this chapter.

Differently from these previous works, the framework proposed in this chapter focuses on transfer *within* a single environment, *across* experiments, and *across* intervention levels. The set of functions we wish to learn have continuous input spaces of different dimensionality. Therefore, capturing their correlation requires placing a probabilistic model over the inputs which enables mapping between input spaces. The DAG, which we assume to be known and is not available in standard multi-task settings, allows us to define such a model. Therefore, *existing multi-output* GP *models are not applicable to our problem.*

Causal Decision Making Our work is also related to the literature on causal decision-making. As discussed in the previous chapter, studies in this field have focused on multi-armed bandit problems [Bareinboim et al., 2015; Lattimore et al., 2016; Lee and Bareinboim, 2018; Lu et al., 2018] and reinforcement learning [Buesing et al., 2019; Foerster et al., 2018] settings where arms or actions correspond to interventions on a DAG. More recently, we introduced Causal Bayesian Optimization (CBO) as a framework to solve the problem of finding an optimal intervention in a DAG by modelling the intervention functions with GPs. As mentioned before, in CBO each function is modelled independently and their correlation is not accounted for when exploring the intervention space. The model proposed in this chapter overcomes this limitation by introducing a multi-task model for experimental outputs. Finally, in the causal literature there has been a growing interest for experimental design algorithms to learn causal graphs [Greenewald et al., 2019; Hauser and Bühlmann, 2014; He and Geng, 2008] or the observational distributions in a graph [Rubenstein et al., 2017b]. In this chapter, we use our multi-task model within an AL framework so as to efficiently learn the experimental outputs in a causal graph.

7.3 Multi-task learning of intervention functions

In this section we address the following question: *can we develop a joint model for the set of functions* $\mathbf{T}$ *defined in a causal graph and thus transfer information across experiments?*

To answer this question we study the correlation among functions in $\mathbf{T}$ which varies with the topology of the causal graph $\mathcal{G}$. Inspired by previous works on latent force models [Álvarez et al., 2009], we show how any functions in $\mathbf{T}$ can be written as an integral transformation of some base function f, also defined starting from $\mathcal{G}$, via some integral operator L_s such that $t_s(\mathbf{x}) =$

$L_s(f)(\mathbf{x})$, $\forall \mathbf{X}_s \in \mathcal{P}(\mathbf{X})$. We first characterize the latent structure among experimental outputs and provide an explicit expression for both f and L_s for each intervention set (Section 7.3.1). Based on the properties of $\mathcal{G}$, we clarify the conditions for the existence of this function. Exploiting these results, we detail a new model to learn $\mathbf{T}$ which we call the DAG-GP model (Section 7.3.2). In DAG-GP, we place a GP prior on the base function f and propagate our prior assumptions on the remaining part of the graph to analytically derive a joint distribution of the elements in $\mathbf{T}$. The resulting prior distribution incorporates the causal structure and enables the integration of observational and interventional data.

7.3.1 Characterization of the latent structure in a DAG

The following results provide a theoretical foundation for the multi-task causal GP model introduced later. In particular, they characterize when f and L_s exist and detail how to compute them thus fully characterizing when transfer across experiments is possible. All proofs are given in Appendix D.

Definition 7.1. Consider a DAG $\mathcal{G}$ where the treatment variables are denoted by $\mathbf{X}$. Let $\mathbf{L}$ be the set of variables directly confounded with Y, $\mathbf{L}^N$ be the set of variables in $\mathbf{L}$ that are not colliders[4] and $\mathbf{I}$ be the set of parents of Y that is $\text{Pa}(Y)$. For each $\mathbf{X}_s \in \mathcal{P}(\mathbf{X})$ we define the following sets:

- $\mathbf{I}_s^N = \mathbf{I} \backslash (\mathbf{X}_s \cap \mathbf{I})$ represents the set of variables in $\mathbf{I}$ not included in $\mathbf{X}_s$.
- $\mathbf{L}_s^I = \mathbf{L}^N \cap \mathbf{X}_s$ is the set of variables in $\mathbf{L}$ which are included in $\mathbf{X}_s$ and are not colliders.
- $\mathbf{L}_s^N = \mathbf{L}^N \backslash \mathbf{L}_s^I$ is the set of variables in $\mathbf{L}$ that are neither included in $\mathbf{X}_s$ nor colliders.

In the following theorem we denote by $\mathbf{v}_s^N$ the values for the variables in the set $\mathbf{I}_s^N$ while $\mathbf{l}$ represents the values for the set $\mathbf{L}^N$. These values are partitioned in $\mathbf{l}_s^N$ for $\mathbf{L}_s^N$ and $\mathbf{l}_s^I$ for $\mathbf{L}_s^I$ depending on the set $\mathbf{X}_s$ we are considering.

Theorem 7.1. Causal operator. *Consider a causal graph $\mathcal{G}$ and the related* SCM *where the output variable and the treatment variables are denoted by Y and $\mathbf{X}$ respectively. Denote by $\mathbf{L}$ the set of variables in $\mathcal{G}$ that are directly confounded with Y and let $\mathbf{I}$ be the set $Pa(Y)$. Assume that $\mathbf{L}$ does not include nodes that have both unconfounded incoming and outcoming edges. It is possible to prove that, $\forall \mathbf{X}_s \in \mathcal{P}(\mathbf{X})$, the intervention function $t_s(\mathbf{x}) : D(\mathbf{X}_s) \to \mathbb{R}$ can*

[4]Here we call colliders variables that are such on all paths. This means that they only have incoming edges. A collider on a path between e.g. X and Y is a variable that is causally influenced by both X and Y.

be written as $t_s(\mathbf{x}) = L_s(f)(\mathbf{x})$ *where*

$$L_s(f)(\mathbf{x}) = \int \cdots \int \pi_s(\mathbf{x}, (\mathbf{v}_s^N, \mathbf{l}))f(\mathbf{v}, \mathbf{l})\, d\mathbf{v}_s^N\, d\mathbf{l}, \tag{7.2}$$

with $f(\mathbf{v}, \mathbf{l}) = \mathbb{E}\left[Y|do\left(\mathbf{I} = \mathbf{v}\right), \mathbf{L}^N = \mathbf{l}\right]$ *representing a shared latent function and* $\pi_s(\mathbf{x}, (\mathbf{v}_s^N, \mathbf{l})) = p(\mathbf{l}_s^I|\mathbf{l}_s^N)p(\mathbf{v}_s^N, \mathbf{l}_s^N|do\left(\mathbf{X}_s = \mathbf{x}\right))$ *giving the integrating measure for the set* $\mathbf{X}_s$.

We call $L_s(f)(\mathbf{x})$ the *causal operator*, $(\mathbf{I} \cup \mathbf{L}^N)$ the *base set*, $f(\mathbf{v}, \mathbf{l})$ the *base function* and $\pi_s(\cdot, \cdot)$ the *integrating measure* of the set $\mathbf{X}_s$. A simple limiting case arises when the DAG does not include variables directly confounded with Y or $\mathbf{L}$ only includes colliders. In this case $\mathbf{L}^N = \varnothing$ and the base function is included in $\mathbf{T}$ that means that one of the intervention function, namely $\mathbb{E}[Y|\text{do}(\text{Pa}(Y))]$, is itself the base function. Note that we exclude colliders from $\mathbf{L}$ as conditioning on them could open some causal paths. We instead include the non-colliders as conditioning on them in the base function blocks the back-door paths from any intervention set in $\mathcal{P}(\mathbf{X})$ to the target variable via the confounded edge. Theorem 7.1 provides a mechanism to reconstruct all causal effects emerging from $\mathcal{P}(\mathbf{X})$ using the base function as a "driving force". In particular, the integrating measures can be seen as Green's functions incorporating the DAG structure [Álvarez et al., 2009]. Note that, in order to construct a surrogate model based on the result of Theorem 7.1, one needs to estimate the integrating measures. Given the identifiability assumption we make in this work, they can be reduced to do-free expressions and thus estimated using observational data. While the result in Theorem 7.1 can be further generalized to select $\mathbf{I}$ to be different from $\text{Pa}(Y)$, this choice is particularly useful due to the following result.

Corollary 7.1. Minimality of I. *The smallest set* $\mathbf{I}$ *for which Eq.* (7.2) *holds is given by* $Pa(Y)$.

The dimensionality of $\mathbf{I}$, when chosen as $\text{Pa}(Y)$, has properties that have been previously studied in the literature. In the context of causal optimization, it corresponds to the so-called causal intrinsic dimensionality defined in Chapter 6, which refers to the effective dimensionality of the space in which a function is optimized when causal information is available. In addition, the existence of f depends on the properties of the nodes in $\mathbf{L}$ which also represents the smallest set for which Eq. (7.2) holds:

Theorem 7.2. Existence of f. *If* $\mathbf{L}$ *includes nodes that have both unconfounded incoming and outcoming edges the function* f *does not exist.*

Corollary 7.2. Minimality of L. *The set* $\mathbf{L}$ *represents the smallest set for which Eq.* (7.2) *holds.*

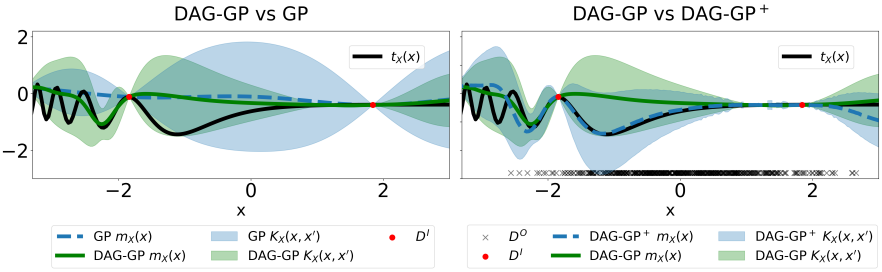


Figure 7.2: Posterior mean and variance for $t_X(x)$ in the DAG of Fig. 7.4(a) (without the red edge). For both plots $m_X(\cdot)$ and $k_X(\cdot,\cdot)$ give the posterior mean and standard deviation respectively. *Left*: Comparison between the DAG-GP model and a single-task GP model (GP). DAG-GP captures the behaviour of $t_X(\mathbf{x})$ in areas where $\mathcal{D}^I$ is not available (see area around $x=-2$) while reducing the uncertainty via transfer due to available data for $\mathbf{z}$. *Right*: Comparison between DAG-GP with the causal prior (DAG-GP$^+$) and a standard prior with zero mean and RBF kernel (DAG-GP). In addition to transfer, DAG-GP$^+$ captures the behaviour of $t_X(x)$ in areas where $\mathcal{D}^O$ (black $\times$) is available (see region $[-2,0]$) while inflating the uncertainty in areas with no observational data.

When f does not exist, full transfer across *all* functions in $\mathbf{T}$ is not possible. Examples of such settings are given in Fig. 7.4 where the DAGs with red edges do not admit a base function. However, these theoretical results can be used to construct a model for partial transfer across a subset of $\mathbf{T}$. This is further discussed in Appendix D.2.

7.3.2 The DAG-GP model

Based on the theoretical results derived in the previous section we now introduce our multi-task causal GP model henceforth called the DAG-GP model.

Model Likelihood Let $\mathcal{D}^I = (\mathbf{X}^I, \mathbf{Y}^I)$ be the interventional dataset defined in Section 7.1. Denote by $\mathbf{T}^I$ the collection of intervention vector-valued functions computed at $\mathbf{X}^I$. Each entry y_s^i in $\mathbf{Y}^I$, is assumed to be a noisy observation of the corresponding function t_s at $\mathbf{x}_s^i$:

$$y_s^i = t_s(\mathbf{x}_s^i) + \epsilon_s^i, \text{ for } s = 1,\dots,|\mathcal{P}(\mathbf{X})| \text{ and } i = 1,\dots,N_s^I, \tag{7.3}$$

with $\epsilon_s^i \sim \mathcal{N}(0,\sigma^2)$. In compact form, we can write the joint likelihood function for the set of observed interventional outputs as $p(\mathbf{Y}^I|\mathbf{T}^I,\sigma^2) = \mathcal{N}(\mathbf{T}^I,\sigma^2\mathbf{I})$.

Prior distribution on T To define a join prior distribution on the set of intervention functions, that is $p(\mathbf{T})$, we take the following two steps. First, we follow the approach adopted in CBO and place a *causal prior* on f (see Section 6.3.2 in Chapter 6), the base function of the DAG. Second, we propagate this prior on f through all elements in $\mathbf{T}$ via the causal operator in Eq. (7.2)

to obtain the full prior.

Step 1. *Causal prior on the base function.* The key idea of the causal prior, introduced for CBO in Chapter 6, is to use the observational dataset $\mathcal{D}^O$ and the *do*-calculus to construct the prior mean and variance of a GP that is used to model an intervention function. In this setting, we compute such prior for the causal effect of the set $\mathbf{I}$ on Y while conditioning on $\mathbf{L}^N$. The causal prior has the benefit of carrying causal information but at the expense of requiring $\mathcal{D}^O$ to estimate the causal effect. Any sensible prior can be used in this step, so the availability of $\mathcal{D}^O$ is not strictly a necessity. However, in this chapter, we stick to the causal prior since it provides an explicit way of combining experimental and observational data and, as shown in Chapter 6, it significantly improves the estimation of the causal effects in a variety of causal graphs.

For simplicity we use $\mathbf{b} = (\mathbf{v}, \mathbf{l})$ to denote in compact form the values of the variables in the base set $\mathbf{I} = \mathbf{v}$ and $\mathbf{L}^N = \mathbf{l}$. Given our identifiability assumption, using *do-calculus* we can compute $\hat{f}(\mathbf{b}) = \hat{f}(\mathbf{v}, \mathbf{l}) = \hat{\mathbb{E}}[Y|\text{do}\left(\mathbf{I} = \mathbf{v}\right), \mathbf{l}]$ and $\hat{\sigma}(\mathbf{b}) = \hat{\sigma}(\mathbf{v}, \mathbf{l}) = \hat{\mathbb{V}}[Y|\text{do}\left(\mathbf{I} = \mathbf{v}\right), \mathbf{l}]^{1/2}$. Here $\hat{\mathbb{V}}$ and $\hat{\mathbb{E}}$ represent the variance and expectation of the causal effects estimated from $\mathcal{D}^O$. The *causal* GP *prior* for the base function is thus defined as:

$$
\begin{aligned}
f(\mathbf{b}) &\sim \mathcal{GP}(m(\mathbf{b}), K(\mathbf{b}, \mathbf{b}')) \\
m(\mathbf{b}) &= \hat{f}(\mathbf{b}) \\
k(\mathbf{b}, \mathbf{b}') &= k_{\text{RBF}}(\mathbf{b}, \mathbf{b}') + \hat{\sigma}(\mathbf{b})\hat{\sigma}(\mathbf{b}')
\end{aligned}
$$

where the term $k_{\text{RBF}}(\mathbf{b}, \mathbf{b}') := \sigma_f^2 \exp(-||\mathbf{b} - \mathbf{b}'||^2/2l^2)$ denotes the radial basis function (RBF) kernel and is added to provide additional flexibility to the model. Note that alternative kernel functions, e.g. a non stationary kernel to capture the behaviour of the intervention function for X in Fig. 7.2, could be easily combined with the additional vartiance term. As done in Chapter 6, in this work we estimate the kernel hyperparameters by maximizing the marginal likelihood via the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm. While this facilitates inference, it fails to account for hyperparameters' uncertainty thus leading to overconfident predictions. In addition, due to the non-convexity of the marginal likelihood, optimization may not converge to the global maxima. However, a fully Bayesian approach would lead to intractable posterior and would thus require resorting to approximate inference methods.

Step 2. *Propagating the distribution to all elements in* $\mathbf{T}$. In Section 7.3.1 we showed how, $\forall \mathbf{X}_s \in \mathcal{P}(\mathbf{X})$, we have $t_s(\mathbf{x}) = L_s(f)(\mathbf{x})$ with f given by the intervention function defined in Theorem 7.1. By linearity of the causal operator, placing a GP prior on f induces a well-defined joint GP prior distribution on $\mathbf{T}$.

In particular, for each $\mathbf{X}_s \in \mathcal{P}(\mathbf{X})$, we have:

$$t_s(\mathbf{x}) \sim \mathcal{GP}(m_s(\mathbf{x}), k_s(\mathbf{x}, \mathbf{x}'))$$

$$m_s(\mathbf{x}) = \int \cdots \int m(\mathbf{b}) \pi_s(\mathbf{x}, \mathbf{b}_s) \, \mathrm{d}\mathbf{b}_s \tag{7.4}$$

$$k_s(\mathbf{x}, \mathbf{x}') = \int \cdots \int K(\mathbf{b}, \mathbf{b}') \pi_s(\mathbf{x}, \mathbf{b}_s) \, \pi_s\left(\mathbf{x}', \mathbf{b}'_s\right) \mathrm{d}\mathbf{b}_s \mathrm{d}\mathbf{b}'_s. \tag{7.5}$$

where $\mathbf{b}_s = (\mathbf{v}_s^N, \mathbf{l})$ is the subset of $\mathbf{b}$ including only the $\mathbf{v}$ values corresponding to the set $\mathbf{I}_s^N$. Let D be a finite set of inputs for the functions in $\mathbf{T}$, that is $D = \bigcup_{s,i} \{\mathbf{x}_s^i\}$. $\mathbf{T}$ computed in D follows a multivariate Gaussian distribution that is $\mathbf{T}^D \sim \mathcal{N}(m_{\mathbf{T}}(D), K_{\mathbf{T}}(D, D))$ with $K_{\mathbf{T}}(D, D) = (K_{\mathbf{T}}(\mathbf{x}, \mathbf{x}'))_{\mathbf{x} \in D, \mathbf{x}' \in D}$ and $m_{\mathbf{T}}(D) = (m_{\mathbf{T}}(\mathbf{x}))_{\mathbf{x} \in D}$. In particular, for two generic data points $\mathbf{x}_s^i, \mathbf{x}_{s'}^j \in D$ with s and s' denoting two *distinct* functions we have $m_{\mathbf{T}}(\mathbf{x}_s^i) = \mathbb{E}\left[t_s(\mathbf{x}_s^i)\right] = m_s(\mathbf{x}_s^i)$ and $K_{\mathbf{T}}(\mathbf{x}_s^i, \mathbf{x}_{s'}^j) = \mathrm{Cov}[t_s(\mathbf{x}_s^i), t_{s'}(\mathbf{x}_{s'}^j)]$.

When computing the covariance function across intervention sets and intervention levels we differentiate between two cases. When both t_s and $t_{s'}$ are different from f, we have:

$$\mathrm{Cov}[t_s(\mathbf{x}_s^i), t_{s\prime}(\mathbf{x}_{s'}^j)] = \int \cdots \int k(\mathbf{b}, \mathbf{b}') \pi_s\left(\mathbf{x}_s^i, \mathbf{b}_s\right) \pi_{s'}\left(\mathbf{x}_{s'}^j, \mathbf{b}'_{s'}\right) \mathrm{d}\mathbf{b}_s \mathrm{d}\mathbf{b}'_{s'}.$$

If one of the two functions equals f, in this case the s-th function, this expression further reduces to:

$$\mathrm{Cov}[t_s(\mathbf{x}_s^i), t_{s\prime}(\mathbf{x}_{s'}^j)] = \int k(\mathbf{b}, \mathbf{b}') \pi_{s'}\left(\mathbf{x}_{s'}^j, \mathbf{b}'_{s'}\right) \mathrm{d}\mathbf{b}'_{s'}.$$

Note that the integrating measures $\pi_s(\cdot, \cdot)$ and $\pi_{s'}(\cdot, \cdot)$ allow to compute the covariance between points that are defined on spaces on possibly different dimensionality, *a scenario that traditional multi-output* GP *models are unable to handle*. The prior $p(\mathbf{T})$ enables to merge different data types and to account for the natural correlation structure among interventions defined by the topology of the DAG. For this reason, we call this formulation the DAG-GP model. The parameters in Eqs. (7.4)–(7.5) can be computed in closed form only when $k(\mathbf{b}, \mathbf{b}')$ is an RBF kernel and the integrating measures are assumed to be Gaussian distributions. In all other cases, one needs to resort to numerical approximations e.g. Monte Carlo integration in order to compute the parameters of each $t_s(\mathbf{x})$. This is the approach used in this chapter.

Posterior distribution on T: The posterior distribution $p(\mathbf{T}^D | \mathcal{D}^I)$ can be derived analytically via standard GP updates. For any set D the posterior is a

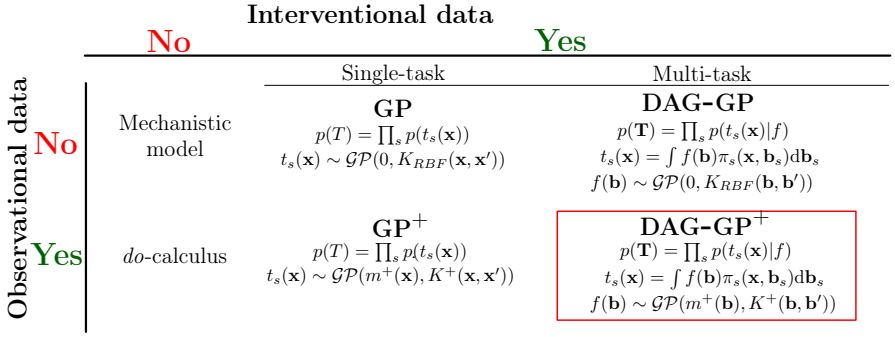


Figure 7.3: Models for learning the intervention functions $\mathbf{T}$ defined on a DAG. The *do*-calculus allows estimating $\mathbf{T}$ when only the observational data is available. When the interventional data is also available, one can use a single-task model (denoted by GP) for each intervention function or a joint multi-task model (denoted by DAG-GP) when the base function exists. When both data types are available one can combine them using the causal GP construction with parameters represented by $m^+(\cdot)$ and $k^+(\cdot,\cdot)$. The resulting single-task and multi-task models are denoted by GP$^+$ and DAG-GP$^+$ respectively.

Gaussian distribution with parameters given by:

$$
\begin{aligned}
p(\mathbf{T}^D|\mathcal{D}^I) &= \mathcal{N}(m_{\mathbf{T}|\mathcal{D}^I}(D), K_{\mathbf{T}|\mathcal{D}^I}(D,D)) \\
m_{\mathbf{T}|\mathcal{D}^I}(D) &= m_{\mathbf{T}}(D) + K_{\mathbf{T}}(D, \mathbf{X}^I)[K_{\mathbf{T}}(\mathbf{X}^I, \mathbf{X}^I) + \sigma^2\mathbf{I}](\mathbf{T}^I - m_{\mathbf{T}}(\mathbf{X}^I)) \\
K_{\mathbf{T}|\mathcal{D}^I}(D,D) &= K_{\mathbf{T}}(D,D) - K_{\mathbf{T}}(D, \mathbf{X}^I)[K_{\mathbf{T}}(\mathbf{X}^I, \mathbf{X}^I) + \sigma^2\mathbf{I}]K_{\mathbf{T}}(\mathbf{X}^I, D)
\end{aligned}
$$

where $m_{\mathbf{T}}(\cdot)$ and $K_{\mathbf{T}}(\cdot,\cdot)$ are the prior parameters of the joint distribution on $\mathbf{T}$ obtained by concatenating $m_s(\cdot)$ and $k_s(\cdot,\cdot)$ corresponding to the sets $\mathbf{X}_s$ included in D. See Fig. 7.2 for an illustration of the DAG-GP model compared to a single-task GP when used to model the function $t_X(x)$ in the DAG of Fig. 7.4. Notice how the DAG-GP model captures the behaviour of $t_X(x)$ in areas where neither the observational nor the interventional data is available (left panel). This is due to the interventional information transferred by the intervention set $\{Z\}$. When the causal prior construction is used, the performance of the DAG-GP model in capturing the target function further improves (right panel). The time complexity of the algorithm is $\mathcal{O}(N^3)$ with N denoting the size of $\mathcal{D}^I$. This complexity can be reduced by resorting to sparse GP approximations e.g. inducing points approximations.

7.4 A helicopter view

In this section, we discuss the links between different model specifications and clarify which approach should be used depending on the availability of different data types that is observational $\mathcal{D}^O$ and interventional data $\mathcal{D}^I$. Our goal here is not to be exhaustive, nor prescriptive, but to help to give some perspective.

A summary table of the methods is provided in Fig. 7.3.

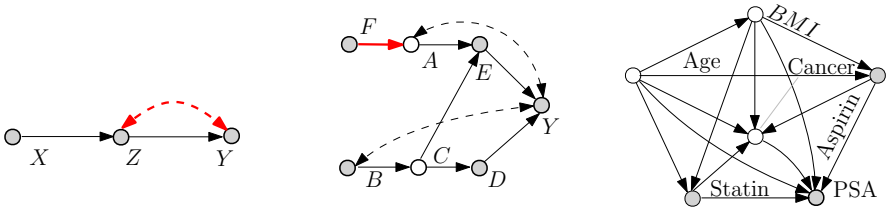
When interventional data $\mathcal{D}^I$ is not available, the do-calculus is the only way to learn $\mathbf{T}$ and compute approximate causal effects in a DAG. In turn, this requires observational data $\mathcal{D}^O$ which is used to estimate the conditional distributions in the SCM. When both data types are not available, learning $\mathbf{T}$ via a probabilistic model is not possible unless the causal effects can be transported from an alternative population exhibiting the same SCM. In this case, mechanistic models based on physical knowledge of the process under investigation are the only option.

When instead interventional data $\mathcal{D}^I$ are available one can consider a single-task or a multi-task model. If the base function f does not exist, a single GP model needs to be considered for each intervention function. This can be defined via a standard prior with zero mean function and RBF kernel, denoted by GP in the table or integrating observational data via the causal GP prior when these are available. This option is denoted by GP$^+$ in the table and corresponds to the surrogate model used for CBO in Chapter 6. Recall that, independently on the prior construction, with this formulation the experimental information is not shared across functions and learning $\mathbf{T}$ requires intervening on all sets in $\mathcal{P}(\mathbf{X})$. When instead the base function f exists, DAG-GP can be used to transfer interventional information and, depending on $\mathcal{D}^O$, also incorporating observational information a priori. This is the formulation proposed in this chapter and denoted by DAG-GP$^+$ in the table.

7.5 Experiments

This section evaluates the performance of the DAG-GP model on two synthetic settings and on a real world healthcare application (Fig. 7.4). We first learn $\mathbf{T}$ with fixed observational and interventional data (Section 7.5.1) and then use the DAG-GP model to solve active learning (AL) (Section 7.5.2) and Causal Bayesian Optimization (CBO) (Section 7.5.3). Code and data for all the experiments is provided at `https://github.com/VirgiAgl/DAG-GP`.

Baselines We run our algorithm both with (DAG-GP$^+$) and without (DAG-GP) causal prior and compare against the alternative models described in Fig. 7.3. Note that we do not compare against alternative multi-task GP models because, as mentioned in Section 7.2, the models existing in the literature cannot deal with functions defined on different inputs spaces and thus can not be straightforwardly applied to our problem.



(a) (b) (c)

Figure 7.4: Examples of DAGs (in black) for which the base function f exists and the DAG-GP model can be formulated. Shaded nodes give manipulative variables while empty nodes represent non-manipulative nodes. Y and PSA are the target variables. The red edges, if added, prevent the identification of f making the transfer via the DAG-GP model not possible.

Table 7.1: RMSE performances across 10 initializations of $\mathcal{D}^I$. See Fig. 7.3 for a summary on the compared methods. *do* stands for the *do*-calculus. N is the size of $\mathcal{D}^O$. Standard errors in brackets.

	$N = 30$					$N = 100$				
	DAG-GP$^+$	DAG-GP	GP$^+$	GP	*do*	DAG-GP$^+$	DAG-GP	GP$^+$	GP	*do*
DAG1	**0.46**	0.57	0.60	0.77	0.70	**0.43**	0.57	0.45	0.77	0.52
	(0.06)	(0.09)	(0.2)	(0.27)	-	(0.05)	(0.08)	(0.05)	(0.27)	-
DAG2	**0.44**	0.45	0.62	1.26	1.40	**0.36**	0.41	0.58	1.28	1.41
	(0.1)	(0.13)	(0.10)	(0.11)	-	(0.09)	(0.12)	(0.07)	(0.11)	-
DAG3	**0.05**	0.44	0.23	0.89	0.18	**0.06**	0.44	0.48	0.89	0.23
	(0.04)	(0.12)	(0.03)	(0.23)	-	(0.04)	(0.12)	(0.06)	(0.23)	-

Performance measures We run all models with different initialisation of $\mathcal{D}^I$ and different sizes of $\mathcal{D}^O$. We report the root mean square error (RMSE) performances together with standard errors across replicates. For the AL experiments, we show the RMSE evolution as the size of $\mathcal{D}^I$ increases. For the CBO experiments we report the convergence performances to the global optimum.

7.5.1 Learning T from data

We test the algorithm on the DAGs in Fig. 7.4 and refer to them as (a) DAG1, (b) DAG2 and (c) DAG3. DAG3 is taken from Thompson [2019] and Ferro et al. [2015] and is used to model the causal effect of statin drugs on the levels of prostate specific antigen (PSA). We consider the nodes $\{A, C\}$ in DAG2 and $\{\text{age}, \text{BMI}, \text{cancer}\}$ in DAG3 to be non-manipulative. We set the size of the interventional dataset $\mathcal{D}^I$ to $5 \times |\mathbf{T}|$ for DAG1 where $|\mathbf{T}| = 2$, to $3 \times |\mathbf{T}|$ for DAG2 where $|\mathbf{T}| = 6$ and to $|\mathbf{T}|$ for DAG3 where $|\mathbf{T}| = 3$. As expected, GP$^+$ outperforms GP incorporating the information in $\mathcal{D}^O$ (Table 7.1). Interestingly, GP$^+$ also outperforms DAG-GP in DAG3 when $N = 30$ and in DAG1 when $N = 100$. This depends on the effect that $\mathcal{D}^O$ has, through its size N and its coverage of the interventional domains, on both the causal prior and the

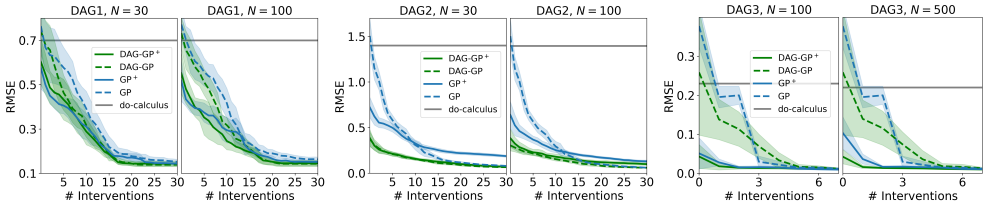


Figure 7.5: AL results. Convergence of the RMSE performance across functions in $\mathbf{T}$ and across replicates as more experiments are collected. DAG-GP$^+$ gives our algorithm with the causal prior while DAG-GP is our algorithm with a standard prior. # interventions is the number of experiments for each $\mathbf{X}_s$. Shaded areas give $\pm$ standard deviation. See Fig. 7.3 for a summary on the compared methods.

estimation of the integrating measures. Lower N and coverage imply not only a less precise estimation of the *do*-calculus but also a worse estimation of the integrating measures and thus a lower transfer of information. Higher N and coverage imply more accurate estimation of the causal prior parameters and enhanced transfer of information across experiments. In addition, the way in which $\mathcal{D}^O$ affects the performance results is specific to the DAG structure and to the distribution of the exogenous variables in the SCM which in turn affects the conditional distribution estimations. Table 7.1 shows how DAG-GP$^+$ consistently outperforms all competing methods by successfully integrating different data sources and transferring interventional information across functions in $\mathbf{T}$. Differently from competing methods, these results hold across different N and $\mathcal{D}^I$ values making DAG-GP$^+$ a robust default choice for any application.

7.5.2 DAG-GP as surrogate model in Active Learning

We now analyse the effect of using the DAG-GP framework as a surrogate model for AL. The goal of AL is to design a sequence of function evaluations to perform in order to learn a target function, or a set of target functions, as quickly as possible. Denote by D a set of inputs for the functions in $\mathbf{T}$, that is $D = \bigcup_s D_s$ with $D_s \subset D(\mathbf{X}_s)$ and consider a subset $A \subset D$ of size k. We would like to select A, that is select both the functions to be observed and the locations, so as to maximize the reduction of entropy in the remaining unobserved locations:

$$A^\star = \underset{A:|A|=k}{\operatorname{argmax}}\, H(\mathbf{T}(D\backslash A)) - H(\mathbf{T}(D\backslash A)|\mathbf{T}(A)).$$

where $H(\cdot)$ represents the entropy, $\mathbf{T}(D\backslash A)$ denotes the set of functions $\mathbf{T}$ evaluated in $D\backslash A$ and $\mathbf{T}(D\backslash A)|\mathbf{T}(A)$ gives the distribution for $\mathbf{T}$ at $(D\backslash A)$ given that we have observed $\mathbf{T}(A)$. While this problem is NP-complete, Krause et al. [2008] proposed an efficient greedy algorithm providing an approximation for $A^\star$. This algorithm starts with an empty set $A = \varnothing$

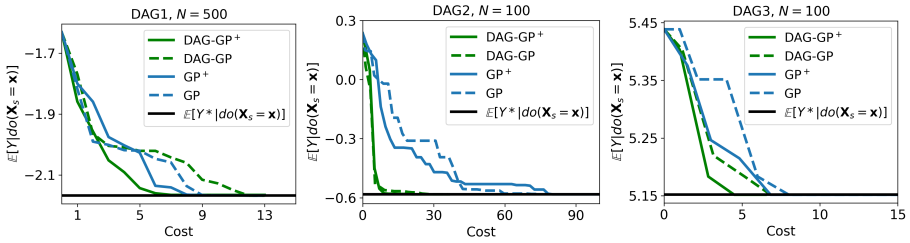


Figure 7.6: BO results. Convergence of the CBO algorithm to the global optimum ($\mathbb{E}[Y^\star|\text{do}\,(\mathbf{X}_s = \mathbf{x})]$) when our algorithm is used as a surrogate model with (DAG-GP$^+$) and without (DAG-GP) the causal prior. See Fig. 7.3 for a summary of the compared methods. See the supplement for standard deviations across replicates.

and solves the problem sequentially by selecting, at every step j, a point $\mathbf{x}_s^j = \text{argmax}_{\mathbf{x}_s^j \in D_s \setminus A^{j-1}} H(t_s(\mathbf{x})|A^{j-1}) - H(t_s(\mathbf{x})|D_s \setminus (A^{j-1} \cup \mathbf{x}_s^j))$ where A^{j-1} denotes the points collected in the $j-1$ steps. Both $H(t_s(\mathbf{x})|A^{j-1}) = \frac{1}{2}\log(2\pi\sigma^2_{\mathbf{x}_s^j|A^{j-1}})$ and $H(t_s(\mathbf{x})|D_s \setminus (A^{j-1} \cup \mathbf{x}_s^j) = \frac{1}{2}\log(2\pi\sigma^2_{\mathbf{x}_s^j|D_s \setminus (A^{j-1} \cup \mathbf{x}_s^j)})$ do not depend on the observed $\mathbf{T}$ values thus the set $A^\star$ can be selected before any function evaluation is collected. In order to select the next intervention level and set while properly accounting for uncertainty reduction, one can use a probabilistic model for $\mathbf{T}$ and get estimates for the terms $\sigma^2_{\mathbf{x}_s^j|A^{j-1}}$ and $\sigma^2_{\mathbf{x}_s^j|D_s \setminus (A^{j-1} \cup \mathbf{x}_s^j)}$ for every $\mathbf{X}_s$.

We run the AL algorithm proposed by Krause et al. [2008] using DAG-GP as a surrogate model and select observations based on the Mutual Information (MI) criteria extended to a multi-task setting. Fig. 7.5 shows the RMSE performances as more interventional data are collected. Across different N settings, DAG-GP$^+$ converges to the lowest RMSE performance faster than competing methods by collecting evaluations in areas where: (i) $\mathcal{D}^O$ does not provide information and (ii) the predictive variance is not reduced by the experimental information transferred from the other interventions. As mentioned before, $\mathcal{D}^O$ impacts on the causal prior parameters via the *do*-calculus computations. When the latter are less precise, because of lower N or lower coverage of the interventional domains, the model variances for DAG-GP$^+$ or GP$^+$ are inflated. Therefore, when DAG-GP$^+$ or GP$^+$ are used as surrogate models, the interventions are collected mainly in areas where $\mathcal{D}^O$ is not observed thus slowing down the exploration of the interventional domains and the convergence to the minimum RMSE (see Fig. 7.5, DAG2, $N = 100$).

7.5.3 DAG-GP as surrogate model in CBO

Finally, we use DAG-GP as a surrogate model for the CBO algorithm introduced in Chapter 6. Note that we only change the surrogate model and keep the

same EI acquisition function and ϵ-greedy policy. As in Chapter 6, our choice of acquisition function is due to its computational tractability and the possibility to straightforwardly compare improvements over multiple surrogate models. We leave the development of alternative causal acquisition functions to future work. Differently from the standard CBO framework, every time we collect a new data point we update all surrogate models thus sharing interventional information across causal effects models. We compare DAG-GP against the single-task models used in Chapter 6 both with and without causal GP prior. Independently on the prior construction, we found DAG-GP to significantly speed up the convergence of CBO to the global optimum (Fig. 7.6) across different causal graphs. This confirms the benefit of using the DAG-GP model to support decision-makers in finding optimal interventions in real-life scenarios thus avoiding expensive and invasive interventions.

7.6 Conclusions and Discussion

This chapter addresses the problems of modelling the correlation structure of a set of intervention functions defined on the DAG of a causal model. Similarly to the model developed in Chapter 4 in the context of Poisson point processes, we tackle this issue by proposing a multi-task GP framework, called the DAG-GP model, that is inspired by the literature on latent force models (see Section 2.2) and captures the DAG topology via a set of integrating measures. These can be seen as smoothing kernels in convolutional GP models or Green's functions in latent force models. The DAG-GP model is based on a theoretical analysis of the DAG structure and allows to share experimental information across interventions while integrating observational and interventional data via *do*-calculus. As seen when comparing single-task and multi-task models in acausal settings (Chapter 4), we found DAG-GP to outperform competing single-task approaches in terms of fitting performances. In addition, the DAG-GP model significantly increases the performance of sequential causal decision-making algorithms, such as CBO or AL, when used as a surrogate model. This is due to the better uncertainty quantification we obtain in DAG-GP thanks to the transfer of interventional data which in turn drives the exploration of the action space to more promising regions.

It remains an intriguing open question to analyse whether the DAG-GP model can be used to transfer experimental information *across* environments whose DAGs are partially different. Developing a joint probabilistic model for all intervention functions across different systems would allow us to infer causal effects for environments where no data is available and only mechanistic models would be used at the moment. In addition, it would allow performing

experiments in systems where the cost of intervening is lower and then transfer the results thereby lowering the overall cost of experimentation.

Chapter 8

Dynamic Causal Bayesian Optimization

As discussed in the previous chapters, solving decision-making problems in a variety of domains requires understanding cause-effect relationships in a system. This can be obtained by experimenting in the system, selecting interventions based on the Causal Bayesian Optimization (CBO) decision-making framework introduced in Chapter 6 and, when this is possible, using the DAG-GP model of Chapter 7 as a surrogate. However, both CBO and DAG-GP focus on static settings where the variables are treated as i.i.d. over time, and their time evolution is disregarded. Deciding how to intervene at every point in time is particularly complex in dynamical systems, due to the evolving nature of causal effects. For instance, companies need to decide how to allocate scarce resources across different quarters. Alternatively, in healthcare, doctors need to select an optimal sequence of treatments over a given time horizon. This chapter describes a probabilistic framework that can be used to find optimal interventions over time.

Focusing on a specific example, consider a setting in which Y_t denotes the unemployment rate of an economy at time t, Z_t is the economic growth and X_t is the inflation rate. Fig. 8.1(a) depicts the causal graph representing an agent's understanding of the causal links between these variables. The agent aims at determining, at each time step $t \in \{0, 1, 2\}$, the optimal action to perform in order to minimize the *current* unemployment rate Y_t while accounting for the intervention cost. The investigator could frame this setting as a sequence of global optimization problems and find the solutions by resorting to CBO. However, CBO does not account for the system's temporal evolution thus breaking the time dependency structure existing among variables, see Fig. 8.1(b). This might lead to sub-optimal solutions, especially in non-stationary scenarios. The same would happen when using Adaptive Bayesian Optimization [ABO, Nyikosa et al., 2018] which is represented in Fig. 8.1(c) or BO which is given in

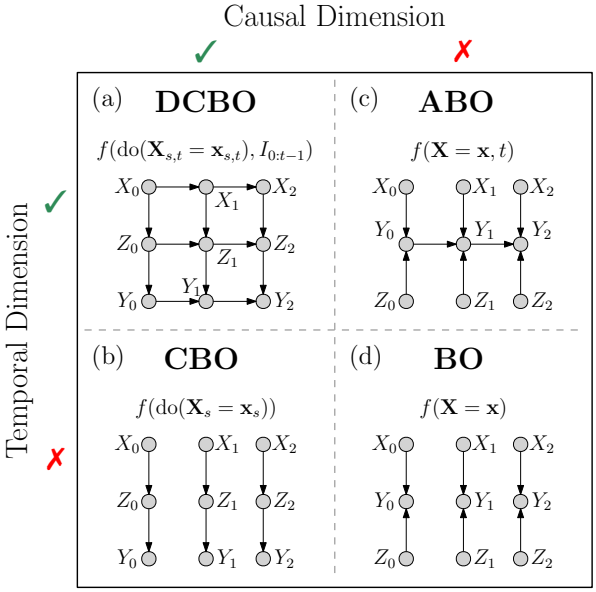


Figure 8.1: DAG representation of a dynamic causal global optimisation (DCGO) problem (a) and the DAG considered when using CBO, ABO or BO to address the same problem. Shaded nodes gives observed variables while the arrows represent causal effects.

Fig. 8.1(d). Indeed, ABO captures the time dependency of the objective function but neither considers the causal structure among inputs nor their temporal evolution. BO disregards both the temporal and the causal structure. Note that neither BO nor ABO was developed to deal with causal settings. Therefore, the causal graphs in Fig. 8.1 associated with these two methods only aim at helping the reader link different existing approaches that are then tested in the experimental comparison. In other words, the goal of Fig. 8.1 is to give the reader some perspective of the methods she can adopt when facing the problem described above and detailed later in the chapter.

Furthermore, the setting we consider in this chapter differs from both reinforcement learning (RL) and multi-armed bandits settings (MAB). Differently from MAB, we consider interventions on continuous variables where the dynamic target variable has a non-stationary interventional distribution. In addition, compared to RL, we do not model the state dynamics explicitly and allow the agent to perform a number of explorative interventions, which do not change the underlying state of the system, before selecting the optimal action. We discuss these points further in Section 8.2.

Dynamic Causal Bayesian Optimization, henceforth referred to as DCBO, accounts for both the causal relationships among input variables and the causality between inputs and outputs which might evolve over time. This allows DCBO to determine e.g. how the level of inflation rate should be manipulated

in order minimize the unemployment rate at every time step. DCBO integrates CBO with dynamic Bayesian networks (DBN), offering a novel approach for decision making under uncertainty within dynamical systems. DBN [Koller and Friedman, 2009] are commonly used in time-series modelling and carry dependence assumptions that do not imply causation. Instead, in probabilistic causal models [Pearl, 2009b], which form the basis for the CBO framework, graphs are built around causal information and allow us to reason about the effects of different interventions. By combining CBO with DBNs, the methodology proposed in this chapter finds an optimal *sequence* of interventions that accounts for the causal temporal dynamics of the system. In addition, DCBO takes into account past optimal interventions and transfers this information across time, thus identifying the optimal intervention faster than competing approaches and at a lower cost. We make the following contributions:

Dynamic Causal Global Optimization We formulate a new class of optimization problems called Dynamic Causal Global Optimization (DCGO) where the objective functions account for the temporal causal dynamics among the variables and generalises the global optimization problem defined in Chapter 6.

Theoretical results on causal dynamic graphs We give theoretical results demonstrating how interventional information can be transferred across time-steps depending on the topology of the causal graph. We provide a recursion formula that can be used to express the causal effects at one time step in terms of previous causal effects for a general causal graph structure.

Dynamic Causal Bayesian Optimization algorithm Exploiting our theoretical results, we solve the optimization problem with DCBO. At every time step, DCBO constructs surrogate models for different intervention sets by integrating various sources of data while accounting for past interventions.

Experimental comparison across eight different settings We analyse the performance of DCBO in a variety of settings comparing against CBO, ABO, and BO. Specifically, we compare all methods on three synthetic graphs specifying alternative stationary and non-stationary structural causal models. We then evaluate the performances in two real-world settings.

8.1 Problem Setup

We consider a structural causal model (SCM) as defined in Definition 3.1 of Section 3.2 and the associated causal directed acyclic graph (DAG) denoted by $\mathcal{G}$[1].

[1] As in the previous chapters, here we assume $\mathcal{G}$ to be known. However, one could run a causal discovery algorithm as a pre-processing step or use interventional data to discriminate

Within the complete set of variables in the SCM, we distinguish between three different types of variables: treatment variables $\mathbf{X}$ that can be manipulated and set to specific values, non-manipulative variables $\mathbf{C}$, which cannot be modified, and the output variable Y that represents the agent's outcome of interest. As done in Chapter 6 and Chapter 7, we denote the *interventional distribution* for two disjoint sets in $\mathbf{V}$, say $\mathbf{X}$ and Y, as $P(Y|\text{do}\,(\mathbf{X}=\mathbf{x}))$. This is the distribution of Y obtained by intervening on $\mathbf{X}$ and fixing its value to $\mathbf{x}$ in the data generating mechanism, irrespective of the values of its parents. The interventional distribution differs from the *observational distribution* which is denoted by $P(Y|\mathbf{X}=\mathbf{x})$. In this chapter, we assume the causal effect for $\mathbf{X}$ on Y to be identifiable $\forall \mathbf{X} \in \mathcal{P}(\mathbf{X})$ with $\mathcal{P}(\mathbf{X})$ denoting the power set of $\mathbf{X}$. When this is the case (see Galles and Pearl [1995] for the set of identifiability conditions given a causal graph), *do*-calculus allows the estimation of interventional distributions and thus causal effects from observational distributions [Pearl, 1995]. However, the *do*-calculus involves computing integrals which are generally not tractable. When this is the case, observational data can be used to get a Monte Carlo estimate, e.g. $\hat{P}(\mathbf{Y}|\text{do}\,(\mathbf{X}=\mathbf{x})) \approx P(\mathbf{Y}|\text{do}\,(\mathbf{X}=\mathbf{x}))$, which is consistent when the number of samples drawn from $P(\mathbf{V})$ is sufficiently large. $\mathcal{D}^O$ and $\mathcal{D}^I$ denote observational and interventional datasets respectively.

Causality in time One can encode the existence of causal mechanisms across time steps by explicitly representing these relationships with edges in an extended graph denoted by $\mathcal{G}_{0:T}$. For instance, the DAG in Fig. 8.1(a) can be seen as one of the DAGs in Fig. 8.1(b) propagated in time. The DAG in Fig. 8.1(a) captures both the causal structure existing across time steps and the causal mechanism within every "time-slice" t [Koller and Friedman, 2009]. Alternatively, in order to reason about interventions that are implemented in a sequential manner, that is *at time t we decide which intervention to perform in the system at the current time step*, we define the following sub-graph $\mathcal{G}_t$ and sub-model M_t:

Definition 8.1. (Sub-SCM M_t) M_t is the SCM at time step t defined as $M_t = \langle \mathbf{U}_{0:t}, \mathbf{V}_{0:t}, \mathbf{F}_{0:t}, P(\mathbf{U}_{0:t}) \rangle$ where $0:t$ denotes the union of the corresponding variables or functions up to time t (see Fig. 8.2). $\mathbf{V}_{0:t}$ includes $\mathbf{X}_{0:t} = \mathbf{X}_t$, $\mathbf{Y}_{0:t} = Y_t$ and $\mathbf{C}_{0:t} = \mathbf{C}_t \cup \mathbf{C}_{0:t-1} \cup \mathbf{Y}_{t-1} \cup \mathbf{X}_{t-1}$. The functions in $\mathbf{F}_{0:t}$ corresponding to intervened variables are replaced by constant values while the exogenous variables related to them are excluded from $\mathbf{U}_{0:t}$.

Definition 8.2. (Sub-graph $\mathcal{G}_t$) $\mathcal{G}_t$ is the causal graph associated to M_t. In $\mathcal{G}_t$, the incoming edges in variables intervened at $0:t-1$ are mutilated while

among graphs within the Markov equivalence class. This is an open challenge.

intervened variables are represented by deterministic nodes (squares) – see Fig. 8.2 for an example with $t \in \{0, 1, 2\}$.

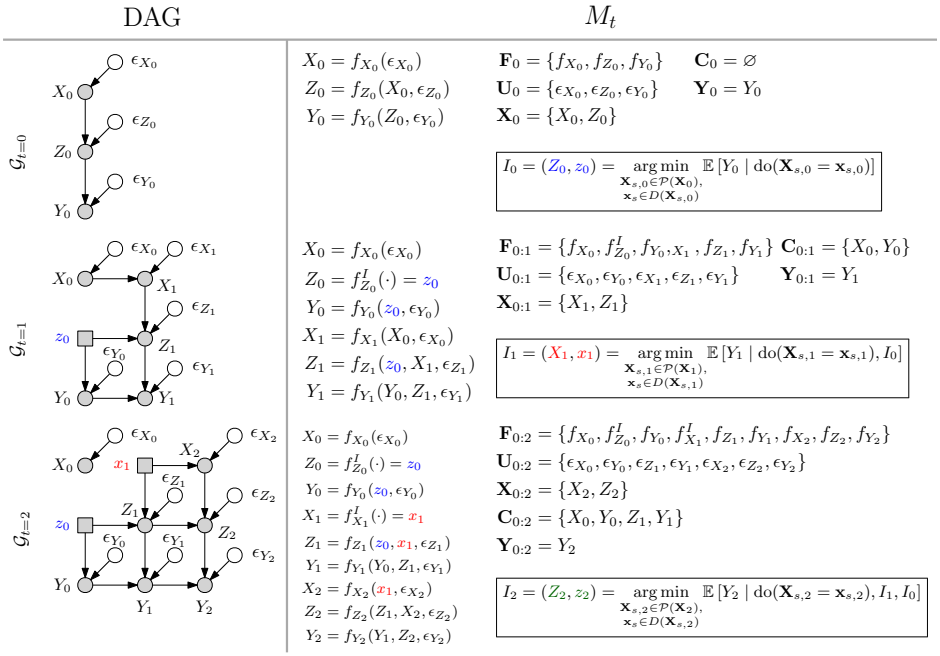


Figure 8.2: Structural equation models considered by DCBO at every time step $t \in \{0, 1, 2\}$. Exogenous noise variables ϵ_i are depicted here but are omitted in the remainder of the paper, to avoid clutter. For every t, $\mathcal{G}_t$ is a mutilated version of $\mathcal{G}_{t-1}$ reflecting the optimal intervention implemented in the system at $0 : t-1$ which are represented by squares. The SCM functions in $\mathbf{F}_{0:t}$ corresponding to the intervened variables are set to constant values. The exogenous variables that only relate to the intervened variables are excluded from U_t. The set of non manipulative variables at every time step denoted by $\mathbf{C}_{0:t}$ is given by the union of the non manipulative variables up to time t, the previous target variables and the previous manipulative variables that is $\{\mathbf{C}_t \cup \mathbf{C}_{0:t-1} \cup \mathbf{Y}_{t-1} \cup \mathbf{X}_{t-1}\}$.

Dynamic Causal Global Optimization (DCGO) The goal of the methodology proposed in this chapter is to find a sequence of interventions to implement in a causal DAG so as to optimize a target variable *at each time step*. Given $\mathcal{G}_t$ and M_t, at every time step t, we wish to optimize Y_t by intervening on a subset of the manipulative variables $\mathbf{X}_t$. The optimal intervention variables $\mathbf{X}^\star_{s,t}$ and intervention levels $\mathbf{x}^\star_{s,t}$ are given by:

$$\mathbf{X}^\star_{s,t}, \mathbf{x}^\star_{s,t} = \underset{\substack{\mathbf{X}_{s,t} \in \mathcal{P}(\mathbf{X}_t) \\ \mathbf{x}_{s,t} \in D(\mathbf{X}_{s,t})}}{\arg\min} \mathbb{E}[Y_t \mid \mathrm{do}\left(\mathbf{X}_{s,t} = \mathbf{x}_{s,t}\right), \mathbb{1}_{t>0} \cdot I_{0:t-1}] \tag{8.1}$$

where $I_{0:t-1} = \bigcup_{i=0}^{t-1} \mathrm{do}\left(\mathbf{X}^\star_{s,i} = \mathbf{x}^\star_{s,i}\right)$ denotes previous interventions, $\mathbb{1}_{t>0}$ is the indicator function and $\mathcal{P}(\mathbf{X}_t)$ is the power set of $\mathbf{X}_t$. $D(\mathbf{X}_{s,t})$ represents

the interventional domain of $\mathbf{X}_{s,t}$. In the sequel we denote the previously intervened variables by $I^V_{0:t-1} = \bigcup_{i=0}^{t-1} \mathbf{X}^\star_{s,i}$ and implemented intervention levels by $I^L_{0:t-1} = \bigcup_{i=0}^{t-1} \mathbf{x}^\star_{s,i}$. The cost of each intervention is given by $Co(\mathbf{X}_{s,t}, \mathbf{x}_{s,t})$. In order to solve the problem in Eq. (8.1) we make the following assumptions :

Assumptions 1. *Denote by $\mathcal{G}(t)$ the causal graph including variables at time t in $\mathcal{G}_{0:T}$ and let $Y_t^{\mathrm{PT}} = Pa(Y_t) \cap Y_{0:t-1}$ be the set of variables in $\mathcal{G}_{0:T}$ that are both parents of Y_t and targets at previous time step. Let the set $Y_t^{\mathrm{PNT}} = Pa(Y_t) \setminus Y_t^{\mathrm{PT}}$ be the complement and denote by $f_{Y_t}(\cdot)$ the functional mapping for Y_t in M_t. We make the following assumptions:*

1. *Invariance of causal structure: $\mathcal{G}(t) = \mathcal{G}(0), \forall t > 0$.*
2. *Additivity of $f_{Y_t}(\cdot)$ that is $Y_t = f_{Y_t}(Pa(Y_t)) + \epsilon$ with $f_{Y_t}(Pa(Y_t)) = f_Y^Y(Y_t^{\mathrm{PT}}) + f_Y^{NY}(Y_t^{\mathrm{PNT}})$ where f_Y^Y and f_Y^{NY} are two generic unknown functions and $\epsilon \sim \mathcal{N}(0, \sigma^2)$.*
3. *Absence of unobserved confounders in $\mathcal{G}_{0:T}$.*

Assumption (3) implies the absence of unobserved confounders at every time step. For instance, this is the case in Fig. 8.1(a). Still in the DAG of Fig. 8.1(a), Assumption (2) implies $f_{Y_t}(\mathrm{Pa}\,(Y_t)) = f_Y^Y(Y_{t-1}) + f_Y^{NY}(Z_t) + \epsilon_{Y_t}, \forall t > 0$. Finally, Assumption (1) implies the existence of the same variables at every time step and a constant orientation of the edges among them for $t > 0$.

Challenges The problem given in Eq. (8.1) is challenging for multiple reasons. As in CBO, finding the optimal intervention involves both exploring $\mathcal{P}(\mathbf{X}_t)$, which grows exponentially with $|\mathbf{X}_t|$, and performing costly intervention to select the optimal level $\mathbf{x}^*_{s,t}$. The time dimension introduces additional challenges. Indeed, the objective function in Eq. (8.1) changes at every time step depending on past decisions. The search space at each time step might also evolve depending on $\mathcal{P}(\mathbf{X}_t)$. Finally, due to the evolving nature of the system, it is reasonable to assume that only a few number of interventional data points can be selected for every t. We thus need to develop an algorithm capable of exploiting and reusing all information, both interventional and observational, collected over time. Importantly, in selecting experiments, we need to account for how the system has been modified by the interventions implemented at previous time steps so as to speed up the identification of the optimal configuration and keep the total intervention cost low.

8.2 Related Work

As seen in the previous chapters, there exists an extensive literature on causal methods focusing on both causal effects estimation and causal discovery methods

[Guo et al., 2020]. The same holds for the literature on dynamic Bayesian networks [Murphy, 2002] that have been used to represent rich dependency structures between variables in a probabilistic model and have been applied in a variety of settings ranging from gene networks modelling [Perrin et al., 2003; Zou and Conzen, 2005] to speech analysis [Jain et al., 2012] and other application areas [Wang et al., 2004; Yao et al., 2008]. The literature on causal dynamic graphs and optimization of variables within them is instead very limited. Here we will review the relevant works within the three fields that are more closely related to the approach introduced in this chapter: dynamic optimization, causal optimization, and decision-making algorithms.

Dynamic Optimization Optimization in dynamic environments has been studied in the context of evolutionary algorithms [Fogel et al., 1966; Goldberg and Smith, 1987]. More recently, other optimization techniques [De et al., 2006; Pelta et al., 2009; Trojanowski and Wierzchoń, 2009] have been adapted to dynamic settings, see e.g. Cruz et al. [2011] for a review. Focusing on BO, the literature on dynamic settings is limited. Azimi et al. [2011] performed batch BO in a dynamic setting where the batch sizes are dynamically determined. Bogunovic et al. [2016] introduced a BO algorithm with bandit feedback and a reward function that varies with time. More recently, Nyikosa et al. [2018] developed ABO, a framework for solving BO on continuous spaces when the function evolution follows a more complex behaviour than a simple Markov model. ABO treats the inputs as fixed and not as random variables, thereby disregarding their temporal evolution and, more importantly, breaking their causal dependencies. In addition, ABO requires a slow rate of change of the objective function so as to gather enough samples to learn the function evolution over space and time. All these dynamic optimization methods tackle the dynamic dimension of the problems we address but do not account for the causal relationships among variables.

Causal Optimization The CBO framework introduced in Chapter 6 focuses instead on the causal aspect of optimization and solves the problem of finding an optimal intervention in a DAG by modelling the intervention functions with single GPs or using the DAG-GP model introduced in Chapter 7. CBO disregards the existence of a temporal evolution in both the inputs and the output variable, treating them as i.i.d. overtime. In many practical applications, the i.i.d. assumption does not provide an adequate description for the data and different causal methodologies have been adapted to deal with longitudinal studies [Granger, 1969; Hyttinen et al., 2013; Peters et al., 2013; Pfister et al., 2019]. While disregarding time significantly simplifies the problem, it prevents the identification of an optimal intervention at every time step t.

Bandits, RL and dynamic treatment regimes In the broader decision-making literature, causal relationships have been considered in the context of bandits [Bareinboim et al., 2015; Lattimore et al., 2016; Lee and Bareinboim, 2018, 2019] and reinforcement learning [Buesing et al., 2019; Foerster et al., 2018; Lu et al., 2018; Madumal et al., 2020; Zhang and Bareinboim, 2019a]. In these cases, actions or arms correspond to interventions on a causal graph where there exist complex links between the agent's decisions and the received rewards. While dynamic settings have been considered in acausal bandit algorithms [Besbes et al., 2014; Villar et al., 2015; Wu et al., 2018], causal MAB have focused on static settings. Dynamic settings are instead considered by RL algorithms and formalized through Markov decision processes (MDP). In the current formulation, DCBO does not consider a MDP as we do not have a notion of *state* and therefore do not require an explicit model of its dynamics. The system is fully specified by the causal model. As in BO, we focus on identifying a set of time-indexed optimal actions rather than an optimal policy. We allow the agent to perform explorative interventions that do not lead to state transitions. More importantly, differently from both MAB and RL, *we allow for the integration of both observational and interventional data.* In the next subsection, we provide an expanded discussion on the links between DCBO, CBO, and ABO. Linking DCBO to the MDPs used by causal RL algorithms is a challenging open problem.

Finally, our work is related to the literature on Dynamic Treatment Regimes (DTRs). DTRs [Murphy, 2003] provide an attractive framework for identifying personalized treatments in longitudinal settings. Specifically, DTRs give us a decision rule that dictates what treatments to provide at each time step given time-varying covariates and treatments' history with the final goal of optimizing a target outcome. These decision rules are also known as adaptive treatment strategies [Lavori and Dawson, 2000, 2008; Murphy, 2005] or treatment policies [Lunceford et al., 2002; Wahed and Tsiatis, 2006]. In the causality literature, Zhang [2020] and Zhang and Bareinboim [2019b] have recently studied the online learning of optimal DTRs in settings where there exists confounded observations and we are given a causal diagram representing the underlying unknown environment. As in RL, but differently from our settings, in DTRs the goal is to identify the sequence of treatments, also called trajectory, optimizing a cumulative regret. On the contrary, DCBO is a myopic algorithm that, at each time step, selects the intervention optimizing the target variable at the current time step. In addition, being a BO algorithm, it does not learn a policy but only identifies a set of actions. Extending DCBO to consider a unique outcome variable possibly delayed in time, but also a cumulative regret function and non-myopic acquisition functions would further shed light on the connection to

RL and DTRs.

8.2.1 Connections

The settings we focus on in this chapter differ from those considered by both CBO and ABO. Here we discuss the main differences and highlight the reasons why DCBO is needed to solve the problem in Eq. (8.1).

CBO algoritm The CBO algorithm can be used to find optimal interventions to perform in a causal graph so as to optimize a single target node Y. CBO addresses static settings where variables in $\mathcal{G}$ are i.i.d. across time steps, i.e. $p(\mathbf{V}_t) = p(\mathbf{V}), \forall t$, and only one static target variable exists. For instance, CBO can be used to find the optimal intervention for Y in the DAG of Fig. 8.1(b). In order to use CBO for the DAG of Fig. 8.1(a), one would need to identify a unique target among $Y_{0:T}$, e.g. Y_T. However, optimizing Y_T might lead to chose interventions that are sub-optimal for $Y_{0:T-1}$ thus not solving the problem in Eq. (8.1). In addition, to find the optimal intervention for Y_T, CBO explores all interventions in $\mathcal{P}(\mathbf{X}_{0:T})$ which results in a large search space and requires performing a high number of interventions. This slows down the convergence of the algorithm and increases the optimization cost. One can alternatively run CBO T times optimizing Y_t at each time step. Doing that would require re-initializing the surrogate models for the objective functions at every t and would thus imply losing all the information collected from previous interventions. Indeed, when optimizing Y_t, CBO does not account for how the previously taken interventions have changed the system again slowing down the convergence of the algorithm. In order to recursively optimise intermediate outputs given the previously taken decisions, one needs to resort to DCBO. By changing the objective function at every time step, incorporating prior interventional information in the objective function, and limiting the search space at every time step based on the topology of the $\mathcal{G}$, DCBO addresses the CBO issues mentioned above making it a framework that can be practically used for sequential decision-making in a variety of applications.

ABO algorithm While CBO tackles the causal dimension of the DCGO problem but not the temporal dimension, the ABO algorithm also addresses dynamic settings but does not account for the causal relationships among variables, see Fig. 8.1 for a graphical representation of the relationship between these methods. As for BO, one could use ABO to solve a DCGO problem by breaking the causal dependencies between the inputs and intervening simultaneously on all of them thus setting $\mathbf{X}_{s,t} = \mathbf{X}_t$ for all t. Additionally, as ABO was originally developed for acausal settings, it considers the inputs as fixed and not as random variables therefore disregarding their temporal evolution. This

is reflected in the DAG of Fig. 8.1(c) where both the horizontal links between the inputs and the edges amongst the input variables are missing.

In solving the problem in Eq. (8.1) for the DAG in Fig. 8.1(a), BO would disregard both the temporal dependencies in Y and the input dependencies (DAG in Fig. 8.1(d)) while ABO would keep the former but ignore the latter. In addition, differently from our approach, ABO considers a continuous time-space and places a surrogate model on $Y_t = f(\mathbf{x}, t)$. $f(\mathbf{x}, t)$ is then modelled via a spatio-temporal GP with a separable kernel. The ABO acquisition function for $f(\mathbf{x}, t)$ is then restricted to avoid collecting points in the past or too far ahead in the future where the GP predictions have high uncertainty. The spatio-temporal GP allows ABO to predict the objective function ahead in time and track the evolution of the optimum. However, in order for ABO to work, the objective function rate of change over time must be slow enough to gather enough samples to learn the relationships in space and time. In our discrete time setting this condition is equivalent to ask that, at every time step, it is possible to perform different interventions with an underlying true function that does not change.

Note that, also in DCBO, Assumptions 1 imply a certain level of regularity in the objective functions. For instance, in the DAG of Fig. 8.1(a), given that $\mathrm{Pa}\,(Y_t) = \{Z_t, Y_{t-1}\}, \forall t > 0$, the objective functions have a constant shape and are only shifted vertically by the performed interventions. While some regularity is also required in DCBO, through the causal graph we impose more structure on the objective function and its input thus lowering the need for exploration. The more accurate the estimation of the functions in the SCM is, the more we can track the dynamics of the objective function and we can deal with sharp changes in the objectives.

One additional important difference between ABO and DCBO is in the exploration of different intervention sets. Indeed, by intervening on all variables, ABO can lead to a sub-optimal solution. As mentioned for BO in Chapter 6, depending on the structural relationships between variables, intervening on a subgroup might lead to a propagation of effects in the causal graph and a higher final target. Finally, intervening on all variables is cost-ineffective in cases when the same target can be obtained by setting only a subgroup of them. This is particularly true in dynamic settings as the optimal intervention set might not only be a subset of $\mathcal{P}(\mathbf{X}_t)$ but might also evolve over time.

8.3 Methodology

In this section, we introduce Dynamic Causal Bayesian Optimization (DCBO), a novel methodology addressing the problem in Eq. (8.1). We first study the correlation among objective functions for two consecutive time steps and use it to derive a recursion formula that, based on the topology of the graph,

expresses the causal effects at time t as a function of previously implemented interventions (see square nodes in Fig. 8.2). Exploiting these results, we develop a new surrogate model for the objective functions that can be used within a CBO framework to find the optimal sequence of interventions. This model enables the integration of observational data, interventional data collected at previous time-steps, and interventional data collected at time t thereby speeding up the identification of the present optimal intervention.

8.3.1 Characterization of the time structure in a DAG with time dependent variables

The following result provides a theoretical foundation for the dynamic causal GP model introduced later. In particular, it derives a recursion formula allowing us to express the objective function at time t as a function of the objective functions corresponding to the optimal interventions at previous time steps. The proof is given in Appendix E.1.

Definition 8.3. Consider a DAG for time steps 0 to T denoted by $\mathcal{G}_{0:T}$ and the objective function $\mathbb{E}[Y_t \mid \mathrm{do}\left(\mathbf{X}_{s,t} = \mathbf{x}_{s,t}\right), I_{0:t-1}]$ for a generic time step $t \in \{0, \dots, T\}$. Denote by $Y_t^{\mathrm{PT}} = (\mathrm{Pa}\left(Y_t\right) \cap Y_{0:t-1})$ the parents of Y_t that are targets at previous time steps and by $Y_t^{\mathrm{PNT}} = \mathrm{Pa}\left(Y_t\right) \setminus Y_t^{\mathrm{PT}}$ the remaining parents. For any $\mathbf{X}_{s,t} \in \mathcal{P}(\mathbf{X}_t)$ and $I_{0:t-1}^V \subseteq \mathbf{X}_{0:t-1}$ we define the following sets:

- $\mathbf{X}_{s,t}^{\mathrm{PY}} = \mathbf{X}_{s,t} \cap \mathrm{Pa}\left(Y_t\right)$ includes the variables in the intervention set $\mathbf{X}_{s,t}$ that are also parents of Y_t.
- $I_{0:t-1}^{\mathrm{PY}} = I_{0:t-1}^V \cap \mathrm{Pa}\left(Y_t\right)$ includes the variables in the intervention set $I_{0:t-1}^V$ that are also parents of Y_t.
- $W \subset \mathrm{Pa}\left(Y_t\right)$ is a set such that $\mathrm{Pa}\left(Y_t\right) = (\mathrm{Pa}\left(Y_t\right) \cap Y_{0:t-1}) \cup \mathbf{X}_{s,t}^{\mathrm{PY}} \cup I_{0:t-1}^{\mathrm{PY}} \cup W$. W includes variables that are parents of Y_t but are not targets nor intervened variables.

The values of the sets $I_{0:t-1}$, $\mathbf{X}_{s,t}^{\mathrm{PY}}$, $I_{0:t-1}^{\mathrm{PY}}$ and W will be denoted by $\mathbf{i}$, $\mathbf{x}^{\mathrm{PY}}$, $\mathbf{i}^{\mathrm{PY}}$ and $\mathbf{w}$ respectively.

Theorem 8.1. Time operator. *Consider a* DAG $\mathcal{G}_{0:T}$ *and the related* SCM *satisfying Assumptions (1). It is possible to prove that,* $\forall \mathbf{X}_{s,t} \in \mathcal{P}(\mathbf{X}_t)$*, the intervention function* $f_{s,t}(\mathbf{x}) = \mathbb{E}[Y_t \mid do\left(\mathbf{X}_{s,t} = \mathbf{x}\right), \mathbb{1}_{t>0} \cdot I_{0:t-1}]$ *with* $f_{s,t}(\mathbf{x}) : D(\mathbf{X}_{s,t}) \to \mathbb{R}$ *can be written as:*

$$f_{s,t}(\mathbf{x}) = f_Y^Y(\mathbf{f}^\star) + \mathbb{E}_{p(\mathbf{w}|do(\mathbf{X}_{s,t}=\mathbf{x}),\mathbf{i})}\left[f_Y^{NY}(\mathbf{x}^{\mathrm{PY}}, \mathbf{i}^{\mathrm{PY}}, \mathbf{w})\right] \tag{8.2}$$

where $\mathbf{f}^\star = \{\mathbb{E}\left[Y_i | do\left(\mathbf{X}_{s,i}^\star = \mathbf{x}_{s,i}^\star\right), I_{0:i-1}\right]\}_{Y_i \in Y_t^{\mathrm{PT}}}$ *that is the set of previously observed optimal targets that are parents of* Y_t*.* f_Y^Y *denotes the function mapping* Y_t^{PT} *to* Y_t *and* f_Y^{NY} *represents the function mapping* Y_t^{PNT} *to* Y_t*.*

Eq. (8.2) reduces to $\mathbb{E}_{p(\mathbf{w}|\text{do}(\mathbf{X}_{s,t}=\mathbf{x}),\mathbf{i})}\left[f_Y^{NY}(\mathbf{x}^{\text{PY}},\mathbf{i}^{\text{PY}},\mathbf{w})\right]$ when Y_t does not depend on previous targets. This is the setting considered in CBO (Chapter 6) that can be thus seen as a particular instance of DCBO. Exploiting Assumptions (1), it is possible to further expand the second term in Eq. (8.2) to get the following expression:

$$\mathbb{E}_{p(\mathbf{w}|\text{do}(\mathbf{X}_{s,t}=\mathbf{x}),\mathbf{i})}\left[f_Y^{NY}(\mathbf{x}^{\text{PY}},\mathbf{i}^{\text{PY}},\mathbf{w})\right]=\mathbb{E}_{p(\mathbf{U}_{0:t})}\left[f_Y^{NY}(\mathbf{x}^{\text{PY}},\mathbf{i}^{\text{PY}},\{C(W)\}_{W\in\mathbf{W}})\right] \tag{8.3}$$

where $p(\mathbf{U}_{0:t})$ is the distribution for the exogenous variables up to time t and $C(W)$ is given by:

$$C(W)=\begin{cases} f_W(\mathbf{u}_W,\mathbf{x}^{\text{PW}},\mathbf{i}^{\text{PW}}) & \text{if} \quad R=\varnothing \\ f_W(\mathbf{u}_W,\mathbf{x}^{\text{PW}},\mathbf{i}^{\text{PW}},r) & \text{if} \quad R\subseteq\mathbf{X}_{s,t}\cup I_{0:t-1}^V \\ f_W(\mathbf{u}_W,\mathbf{x}^{\text{PW}},\mathbf{i}^{\text{PW}},C(R)) & \text{if} \quad R\not\subseteq\mathbf{X}_{s,t}\cup I_{0:t-1}^V \end{cases}$$

where f_W represents the functional mapping for W in the SEM and $\mathbf{u}_W$ is the set of exogenous variables with edges into W. $\mathbf{x}^{\text{PW}}$ and $\mathbf{i}^{\text{PW}}$ are the values corresponding to $\mathbf{X}_{s,t}^{\text{PW}}$ and $I_{0:t-1}^{\text{PW}}$ which in turn represent the subset of variables in $\mathbf{X}_{s,t}$ and $I_{0:t-1}^V$ that are parents of W. Finally r is the value of $R=\text{Pa}\left(W\right)\setminus(\mathbf{X}_{s,t}^{\text{PY}}\cup I_{0:t-1}^{\text{PW}})$.

Examples for Eq. (8.2): For the DAG in Fig. 8.1(a), at time $t=1$ and with $I_{0:t-1}^V=\{Z_0\}$, we have $\mathbb{E}[Y|\text{do}\left(Z_1=z\right),I_0]=f_Y^Y(y_0^\star)+f_Y^{NY}(z)$. Indeed in this case $\mathbf{W}=\varnothing$, $\mathbf{x}^{\text{PY}}=z$ and $\mathbf{f}^\star=\{y_0^\star=\mathbb{E}[Y_0|\text{do}\left(Z_0=z_0\right)]\}$. Still at $t=1$ and with $I_{0:t-1}^V=\{Z_0\}$, the objective function for $\mathbf{X}_{s,t}=\{X_1\}$ can be written as $f_Y^Y(y_0^\star)+\mathbb{E}_{p(z_1|\text{do}(X_1=x),I_0)}\left[f_Y^{NY}(z_1)\right]$ as $\mathbf{W}=\{Z_1\}$. All derivations for these expressions and alternative graphs are given in Appendix E.2.

8.3.2 Restricting the search space

The search space for the problem in Eq. (8.1) grows exponentially with $|\mathbf{X}_t|$ thus slowing down the identification of the optimal intervention when $\mathcal{G}_t$ includes more than a few nodes. Indeed, a naive approach to find $\mathbf{X}_{s,t}^\star$ at $t=0,\dots,T$ would be to explore the $2^{|\mathbf{X}_t|}$ sets in $\mathcal{P}(\mathbf{X}_t)$ at every t and keep $2^{|\mathbf{X}_t|}$ models for the objective functions. In the static setting, CBO reduces the search space by exploiting invariances in the interventional space [Lee and Bareinboim, 2018] to identify a subset of intervention sets $\mathbb{M}\subseteq\mathcal{P}(\mathbf{X})$ worth exploring (see Section 6.3.1). In our dynamic setting, the objective functions change at every time step depending on the previously implemented interventions and one would need to recompute $\mathbb{M}$ at every t. However, it is possible to show that, given Assumptions (1), the search space remains constant over time. Denote by $\mathbb{M}_t$

the set $\mathbb{M}$ at time t and let $\mathbb{M}_0$ represent the set at $t = 0$ which corresponds to the exploration set computed in CBO. For $t > 0$ it is possible to prove that:

Proposition 8.3.1. MIS in time. If Assumptions (1) are satisfied, the search space is constant overtime that is $\mathbb{M}_t = \mathbb{M}_0$ for $t > 0$.

A proof is given in Appendix E.3.

8.3.3 Dynamic Causal GP model

Here we introduce the Dynamic Causal GP model that is used as a surrogate model for the objective functions in Eq. (8.1). The prior parameters are constructed by exploiting the recursion in Eq. (8.2). At each time step t, the agent explores the sets in $\mathbb{M}_t \subseteq \mathcal{P}(\mathbf{X}_t)$ by selecting the next intervention to be the one maximizing a given acquisition function.

Prior Surrogate Model At each time step t and for each $\mathbf{X}_{s,t} \in \mathbb{M}_t$, we place a GP prior on the objective function $f_{s,t}(\mathbf{x}) = \mathbb{E}[Y_t | \text{do}\left(\mathbf{X}_{s,t} = \mathbf{x}\right), \mathbb{1}_{t>0} \cdot I_{0:t-1}]$. We construct the prior parameters exploiting the recursive expression given in Eq. (8.2):

$$
\begin{aligned}
f_{s,t}(\mathbf{x}) &\sim \mathcal{GP}(m_{s,t}(\mathbf{x}), k_{s,t}(\mathbf{x}, \mathbf{x}')) \\
m_{s,t}(\mathbf{x}) &= \mathbb{E}\Big[f_Y^Y(\mathbf{f}^\star) + \widehat{\mathbb{E}}[f_Y^{NY}(\mathbf{x}^{\text{PY}}, \mathbf{i}^{\text{PY}}, \mathbf{w})]\Big] \\
k_{s,t}(\mathbf{x}, \mathbf{x}') &= k_{\text{RBF}}(\mathbf{x}, \mathbf{x}') + \sigma_{s,t}(\mathbf{x})\sigma_{s,t}(\mathbf{x}')
\end{aligned}
$$

with $\sigma_{s,t}(\mathbf{x}) = \sqrt{\mathbb{V}[f_Y^Y(\mathbf{f}^\star) + \hat{\mathbb{E}}[f_Y^{NY}(\mathbf{x}^{\text{PY}}, \mathbf{i}^{\text{PY}}, \mathbf{w})]}$ and $k_{\text{RBF}}(\mathbf{x}, \mathbf{x}')$ representing the radial basis function kernel. The inner expectation in $m_{s,t}(\mathbf{x})$ is equal to $\hat{\mathbb{E}}_{p(\mathbf{w}|\text{do}(\mathbf{X}_{s,t}=\mathbf{x}),\mathbf{i})}[f_Y^{NY}(\mathbf{x}^{\text{PY}}, \mathbf{i}^{\text{PY}}, \mathbf{w})]$ and represents the expected value of $f_Y^{NY}(\mathbf{x}^{\text{PY}}, \mathbf{i}^{\text{PY}}, \mathbf{w})$ with respect to $p(\mathbf{w} \mid \text{do}\left(\mathbf{X}_{s,t} = \mathbf{x}\right), \mathbf{i})$ which is estimated via the *do*-calculus using observational data. The outer expectation in $m_{s,t}(\mathbf{x})$ and the variance in $\sigma_{s,t}(\mathbf{x})$ are computed with respect to $p(f_Y^Y, f_Y^{NY})$ which is also estimated using observational data. In this work we model f_Y^Y, f_Y^{NY} and all functions in the SEM by independent GPs. However, any alternative probabilistic model can be used to learn these functions. We give estimation details for all experiments in Appendix E.4.

Both $m_{s,t}(\mathbf{x})$ and $\sigma_{s,t}(\mathbf{x})$ can be equivalently written by exploiting the equivalence in Eq. (8.3). In both cases, this prior construction allows the integration of three different types of data: observational data, interventional data collected at time t, and the optimal interventional data points collected in the past. The former is used to estimate the SCM model and $p(\mathbf{w}|\text{do}\left(\mathbf{X}_{s,t} = \mathbf{x}\right), \mathbf{i})$ via the rules of do-calculus. The optimal interventional data points at $0 : t-1$ determine the shift $f_Y^Y(\mathbf{f}^\star)$ while the interventional data collected at time t are

used to update the prior distribution on $f_{s,t}(\mathbf{x})$. Similar prior constructions were considered in the statistic settings of Chapter 6 and Chapter 7 where only observational and interventional data at the current (and unique) time step were used. The additional shift term appears here as there exists a causal dynamic in the target variables and the objective function is affected by previous decisions. Fig. E.2 shows a synthetic example in which accounting for the dynamic aspect in the prior formulation leads to a more accurate GP posterior compared to the baselines, especially when the optimum location changes across time steps.

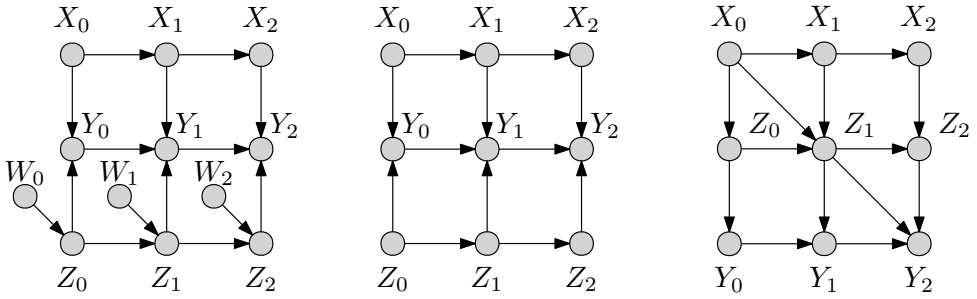
Likelihood Let $\mathcal{D}^I_{s,t} = (\mathbf{X}^I, \mathbf{Y}^I_{s,t})$ be the set of interventional datapoints collected for $\mathbf{X}_{s,t}$ with $\mathbf{X}^I$ being a vector of intervention values and $\mathbf{Y}^I_{s,t}$ representing the corresponding vector of observed target values. As in standard BO we assume each $y_{s,t}$ in $\mathbf{Y}^I_{s,t}$ to be a noisy observation of the function $f_{s,t}(\mathbf{x})$ that is $y_{s,t}(\mathbf{x}) = f_{s,t}(\mathbf{x}) + \epsilon_{s,t}$ with $\epsilon_{s,t} \sim \mathcal{N}(0, \sigma^2)$ for $s \in \{1, \dots, |\mathbb{M}_t|\}$ and $t \in \{0, \dots, T\}$. In compact form, the joint likelihood function for $\mathcal{D}^I_{s,t}$ is $p(\mathbf{Y}^I_{s,t} \mid f_{s,t}, \sigma^2) = \mathcal{N}(f_{s,t}(\mathbf{X}^I), \sigma^2\mathbf{I})$.

Acquisition Function Given our surrogate models at time t, the agent selects the interventions to implement resorting to Causal Bayesian Optimization. Recall from Chapter 6 that the agent explores the sets in $\mathbb{M}_t$ and decides where to intervene by maximizing the Causal Expected Improvement (EI). Denote by $y^\star_t$ the optimal observed target value in $\{\mathbf{Y}^I_{s,t}\}_{s=1}^{|\mathbb{M}_t|}$ that is the optimal observed target across all intervention sets at time t. The Causal EI is given by $\text{EI}_{s,t}(\mathbf{x}) = \mathbb{E}_{p(y_{s,t})}[\max(y_{s,t} - y^\star_t, 0)]/Co(\mathbf{X}_{s,t}, \mathbf{x}_{s,t})$. Let $\alpha_1, \dots, \alpha_{|\mathbb{M}_t|}$ be solutions of the optimization of $\text{EI}_{s,t}(\mathbf{x})$ for each set in $\mathbb{M}_t$ and $\alpha^\star := \max\{\alpha_1, \dots, \alpha_{|\mathbb{M}_t|}\}$. The next best intervention to explore at time t is given by $s^\star = \text{argmax}_{s \in \{1, \cdots, |\mathbb{M}_t|\}} \alpha_s$. Therefore, the set-value pair to intervene on is $(s^\star, \alpha^\star)$. At every t, the agents implement H *explorative* interventions in the system which are selected by maximizing the Causal EI. Once the budget H is exhausted, the agent implements what we call the *decision* intervention I_t, that is the optimal intervention found at the current time step, and move forward to a new optimization at $t+1$ carrying the information in $y^\star_{0:t-1}$. The parameter H determines the level of exploration of the system and acts as a budget for the CBO algorithm. Its value is determined by the agent and is generally problem specific. Note that, in settings where H is low, the exploration of the algorithm at each time step is limited thus the convergence at every t is not guaranteed when moving to $t+1$. In turn, this affects the optimum value that the algorithm can reach at every subsequent step, see Appendix E.4.9 for some experimental results on settings when DCBO is allowed to perform a lower number of trials.

Posterior Surrogate Model For any set $\mathbf{X}_{s,t} \in \mathbb{M}_t$, the posterior distribution $p(f_{s,t} \mid \mathcal{D}^I_{s,t})$ can be derived analytically via standard GP updates. $p(f_{s,t}|\mathcal{D}^I_{s,t})$ will also be a GP with parameters $m_{s,t}(\mathbf{x} \mid \mathcal{D}^I_{s,t}) = m_{s,t}(\mathbf{x}) + k_{s,t}(\mathbf{x}, \mathbf{X}^I)[k_{s,t}(\mathbf{X}^I, \mathbf{X}^I)+\sigma^2\mathbf{I}])(\mathbf{Y}^I_{s,t}-m_{s,t}(\mathbf{X}^I)$ and $k_{s,t}(\mathbf{x}, \mathbf{x}' \mid \mathcal{D}^I_{s,t}) = k_{s,t}(\mathbf{x}, \mathbf{x}')- k_{s,t}(\mathbf{x}, \mathbf{X}^I)[k_{s,t}(\mathbf{X}^I, \mathbf{X}^I) + \sigma^2\mathbf{I}])k_{s,t}(\mathbf{X}^I, \mathbf{x}')$. We give the complete DCBO algorithm in Algorithm 3. The time complexity of DCBO is dominated by algebraic operations on $k_{s,t}(\mathbf{X}^I, \mathbf{X}^I)$ which are $\mathcal{O}(H^3)$ where H denotes the number of collected interventional data points. The space complexity is $\mathcal{O}(H^2)$.

Algorithm 3 DCBO

Inputs: $\mathcal{D}^O$, $\{\mathcal{D}^I_{s,t=0}\}_{s\in\{0,\dots,|\mathbb{M}_0|\}}$, $\mathcal{G}_{0:T}$, H.
Output: Optimal intervention path $\{\mathbf{X}^\star_{s,t}, \mathbf{x}^\star_{s,t}, y^\star_t\}^T_{t=1}$.
Initialize: $\mathbb{M}$, $\mathcal{D}^I_0$ and initial optimal $\mathcal{D}^I_\star = \varnothing$.
for `t=1 to T` **do**
    1. Initialise dynamic causal GP models using $\mathcal{D}^I_{\star,t-1}$ if $t > 0$.
    2. Initialise interventional dataset $\{\mathcal{D}^I_{s,t}\}_{s\in\{0,\dots,|\mathbb{M}_t|\}}$.
    for `h=1 to H` **do**
        1. Compute $\text{EI}_{s,t}(\mathbf{x})$ for each $\mathbf{X}_{s,t} \in \mathbb{M}_t$.
        2. Obtain $(s^\star, \alpha^\star)$.
        3. Intervene and augment $\mathcal{D}^I_{s=s^\star,t}$.
        4. Update posterior for $f_{s=s^\star,t}$.
    end for
    3. Return the optimal intervention $(\mathbf{X}^\star_{s,t}, \mathbf{x}^\star_{s,t})$.
    4. Append optimal interventional data $\mathcal{D}^I_{\star,t} = \mathcal{D}^I_{\star,t-1} \cup ((\mathbf{X}^\star_{s,t}, \mathbf{x}^\star_{s,t}), y^\star_t)$.
end for



(a) MULTIV. (b) IND. (c) NONSTAT.

Figure 8.3: DAGs used in the experimental sections for the synthetic data.

8.4 Experiments

We evaluate the performance of DCBO in a variety of synthetic and real-world settings with DAGs given in Fig. 8.3 and Fig. 8.4. We first run the

algorithm for a stationary setting where both the graph structure and the SEM do not change over time (STAT.). We then consider a scenario characterised by increased observation noise (NOISY) for the manipulative variables and a setting where observational data are missing at some time steps (MISS.). Still assuming stationarity, we then test the algorithm in a DAG where multivariate interventions are included in $\mathbb{M}_t$ (MULTIV.). Lastly, we run DCBO for a non-stationary graph where both the SCM and the DAG change over time (NONSTAT.). To conclude, we use DCBO to optimize the unemployment rate of a closed economy (ECON.) and to find the optimal intervention in a system of ordinary differential equations modelling a real predator-prey system (EVOL.). All implementation details are given in the supplement. Code and data for all the experiments are provided at `https://github.com/VirgiAgl/DCBO`.

Baselines We compare against the algorithms in Fig. 8.1. Note that, by constructions, ABO and BO intervene on all manipulative variables while DCBO and CBO explore only $\mathbb{M}_t$ at every t. In addition, both DCBO and ABO reduce to CBO and BO at the first time step. We assume the availability of an observational dataset $\mathcal{D}^O$ and set a unit intervention cost for all variables.

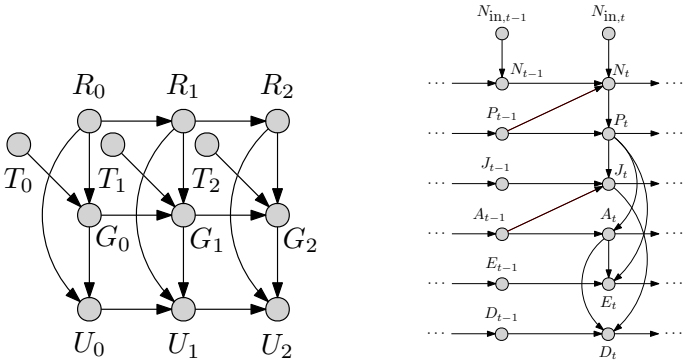
Performance metric We run all experiments for 10 replicates and show the average convergence path at every time step. We then compute the values of a modified "gap" metric across time steps and with standard errors across replicates. This metric is a modified version of the one used in Huang et al. [2006] and is defined as:

$$\mathrm{G}_t = \left(\frac{y(\mathbf{x}^\star_{s,t}) - y(\mathbf{x}_{\text{init}})}{y^\star - y(\mathbf{x}_{\text{init}})} + \frac{H - H(\mathbf{x}^\star_{s,t})}{H} \right) \Big/ \left(1 + \frac{H-1}{H} \right)$$

where $y(\cdot)$ represents the evaluation of the objective function, $y^\star$ is the global minimum, and $\mathbf{x}_{\text{init}}$ and $\mathbf{x}^\star_{s,t}$ are the first and best evaluated point, respectively. The term $\frac{H - H(\mathbf{x}^\star_{s,t})}{H}$ with $H(\mathbf{x}^\star_{s,t})$ denoting the number of explorative trials needed to reach $\mathbf{x}^\star_{s,t}$ captures the speed of the optimization. This term is equal to zero when the algorithm is not converged and equal to $(H-1)/H$ when the algorithm converges at the first trial. We have $0 \leq \mathrm{G}_t \leq 1$ with higher values denoting better performances. For each method, we also show the average percentage of replicates where the optimal intervention set $\mathbf{X}^\star_{s,t}$ is identified.

8.4.1 Synthetic Experiments

Stationary DAG and SEM (STAT.) We run the algorithms for the DAG in Fig. 8.1(a) with $T = 3$ and $N = 10$. *For $t > 0$,* DCBO *converges to the optimal value faster than competing approaches* (see Fig. E.2 in the supplement, right



(a) Econ. (b) Evol.

Figure 8.4: DAGs used in the experimental sections for the real data.

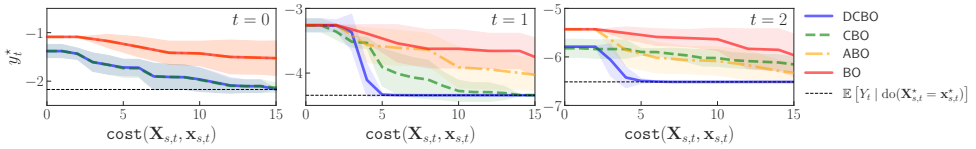


Figure 8.5: Experiment Noisy. Convergence of DCBO and competing methods across replicates. The dashed black line (- - -) gives the optimal outcome $y_t^*, \forall t$. Shaded areas are $\pm$ one standard deviation.

panel, 3rd row). DCBO identifies the optimal intervention set in 93% of the replicates (Table 8.2) and reaches the highest average gap metric (Table 8.1). In this experiment the location of the optimum changes significantly both in terms of optimal set and intervention value when going from $t = 0$ to $t = 1$. This information is incorporated by DCBO through the prior dependency on $y^\star_{0:t-1}$. In addition, ABO performance improves over time as it accumulates interventional data and uses them to fit the temporal dimension of the surrogate model. This benefits ABO in a stationary settings but might penalise it in non stationary settings where the objective functions change significantly.

Noisy manipulative variables (Noisy): *The benefit of using* DCBO *becomes more apparent when the manipulative variables observations are noisy* while the evolution of the target variable is more accurately detected. In this case, both the convergence of DCBO and CBO are slowed down by noisy observations which are diluting the information provided by the do-calculus making the priors less informative. However, the DCBO prior dependency on $y^\star_{0:t-1}$ allows it to correctly identify the shift in the target variable thus improving the prior accuracy and speeding up the algorithm (Fig. 8.5).

Missing observational data (Miss.) *Incorporating dynamic information in the surrogate model allows us to efficiently optimise a target variable even in settings where observational data are missing.* We consider the DAG in Fig. 8.1(a)

with $T = 6$, $N = 10$ for the first three time steps and $N = 0$ afterwards. DCBO uses the observational distributions learned with data from the first three time steps to construct the prior for $t > 3$. On the contrary, CBO uses the standard prior for $t > 3$. In this setting DCBO consistently outperforms CBO at every time step. However, notice how ABO performance improves over time and outperforms DCBO starting from $t = 4$ (see Fig. E.3 in the supplement). As mentioned above, this is due to the ability of ABO to learn the time dynamic of the objective function and exploit all interventional data collected over time to predict at the next time step. While this benefits ABO in stationary settings, it penalises it in nonstationary settings where the objective functions change significantly over time (see NonStat. experiment below).

Table 8.1: Average G_t across 10 replicates and time steps. See Fig. 8.1 for a summary of the baselines. Higher values are better. The best result for each experiment in bold. Standard errors in brackets.

	Synthetic data						Real data	
	Stat.	Miss.	Noisy	Multiv.	Ind.	NonStat.	Econ.	Evol.
DCBO	**0.88** (0.00)	**0.84** (0.01)	**0.75** (0.00)	**0.49** (0.01)	0.48 (0.04)	**0.69** (0.00)	**0.64** (0.01)	**0.67** (0.00)
CBO	0.70 (0.01)	0.70 (0.02)	0.51 (0.02)	0.48 (0.09)	0.47 (0.07)	0.61 (0.00)	0.61 (0.01)	0.65 (0.00)
ABO	0.56 (0.01)	0.49 (0.02)	0.49 (0.04)	0.39 (0.21)	**0.54** (0.01)	0.38 (0.02)	0.57 (0.02)	0.48 (0.01)
BO	0.54 (0.02)	0.48 (0.03)	0.38 (0.05)	0.35 (0.08)	0.50 (0.01)	0.38 (0.03)	0.50 (0.01)	0.44 (0.03)

Multivariate intervention sets (Multiv.) *When the optimal intervention set is multivariate, both* DCBO *and* CBO *convergence speed worsen.* For instance, for the DAG in Fig. 8.3(a), $|\mathbb{M}| = 5$ thus both CBO and DCBO will have to perform more explorative interventions before finding the optimum. At the same time, ABO and BO consider interventions only on $\{W_t, X_t, Z_t\}, \forall t$ and need to explore an even higher intervention space. The performance of all methods decreases in this case (Table 8.1) but DCBO still identifies the optimal intervention set in 93% of the replicates (Table 8.2).

Independent manipulative variables (Ind.): *Having to explore multiple intervention sets significantly penalises* DCBO *and* CBO *when there is no causal relationship among manipulative variables* which are also the only parents of the target. This is the case for the DAG in Fig. 8.3(b) where the optimal intervention is $\{X_t, Z_t\}$ at every time step. In this case, exploring $\mathbb{M}$ and propagating uncertainty in the causal prior slows down DCBO convergence and decreases both its performance (Table 8.1) and capability to identify the optimal intervention set (Table 8.2).

Table 8.2: Average % of replicates across time steps for which $\mathbf{X}^{\star}_{s,t}$ is identified. See Fig. 8.1 for a summary of the baselines. Higher values are better. The best result for each experiment in bold.

	Synthetic data						Real data	
	STAT.	MISS.	NOISY	MULTIV.	IND.	NONSTAT.	ECON.	EVOL.
DCBO	**93.00**	58.00	**100.00**	**93.00**	93.00	**100.00**	86.67	**33.3**
CBO	90.00	**85.00**	90.00	90.0	90.00	**100.00**	**93.33**	**33.3**
ABO	0.00	0.00	0.00	0.00	**100.00**	0.00	66.67	0.00
BO	0.00	0.00	0.00	0.00	**100.00**	0.00	66.67	0.00

Non-stationary DAG and SEM (NONSTAT.): Finally, DCBO *outperforms all approaches in non-stationary settings where both the* DAG *and the* SEM *change overtime* – see Fig. 8.3(c). Indeed, DCBO can timely incorporate changes in the system via the dynamic causal prior construction while CBO, BO and ABO need to perform several interventions before accurately learning the new objective functions and identifying the optimum.

8.4.2 Real experiments

Real-World Economic data (ECON.) We use DCBO to minimize the unemployment rate U_t of a closed economy. We consider its causal relationships with economic growth (G_t), inflation rate (R_t) and fiscal policy[2] (T_t). Inspired by the economic example in Huang et al. [2019] we consider the DAG in Fig. 8.4(a) where R_t and T_t are considered manipulative variables we need to intervene on to minimize $log(U_t)$ at every time step. Time series data for 10 countries[3] are used to construct a non parametric simulator and to compute the causal prior for both DCBO and CBO. DCBO convergences to the optimal intervention faster than competing approaches (see Table 8.1 and Fig. E.6 in the appendix). The optimal sequence of interventions found in this experiment is equal to $\{(T_0, R_0) = (9.38, -2.00), (T_1, R_1) = (0.53, 6.00), (T_2) = (0.012)\}$ which is consistent with domain knowledge.

Planktonic predator-prey community in a chemostat (EVOL.) We investigate a biological system in which two species interact, one as a predator and the other as prey, with the goal of identifying the intervention reducing the concentration of dead animals in the chemostat – see D_t in Fig. 8.4(b). We use the system of ordinary differential equations (ODE) given by Blasius et al. [2020] as our SCM and construct the DAG by rolling out the temporal variable

[2]The causality between economic variables is oversimplified in this example thus the results cannot be used to guide public policy and are only meant to showcase how DCBO can be used within a real application.

[3]Data were downloaded from `https://www.data.oecd.org/ [Accessed: 01/04/2021]`. Details are given in Appendix E.4.7

dependencies in the ODE while removing graph cycles. Observational data are provided in Blasius et al. [2020] and are use to compute the dynamic causal prior. DCBO outperforms competing methods in terms of average gap metric and identifies the optimum faster (Table 8.1). Additional details about this experiment can be found in Appendix E.4.8.

8.5 Conclusions and Discussion

In this chapter, we consider the problem of finding a sequence of optimal interventions in a causal graph where causal temporal dependencies exist between variables. We propose the Dynamic Causal Bayesian Optimization (DCBO) algorithm which finds the optimal intervention at every time step by intervening in the system according to a causal acquisition function. Importantly, for each possible intervention, we developed a surrogate model that incorporates information from previous interventions implemented in the system. This is constructed by exploiting theoretical results establishing the correlation structure among objective functions for two consecutive time steps as a function of the topology of the causal graph. We discuss DCBO performance in a variety of setting characterized by different DAG properties and stationarity assumptions.

Extending our theoretical results to more general DAG structures remains an open problem. In particular, allowing for unobserved confounders and a changing DAG topology within each time step are two important challenges. Finally, as in the previous two chapters, DCBO assumes full knowledge of the DAG. As mentioned earlier, we tackle this issue in Branchini et al. [2022] by developing a framework for joint optimization and causal discovery. This method accounts for uncertainty in the graph structure via a structured surrogate model similar to those seen so far and offers an acquisition function capable of selecting interventions that are useful in jointly identifying the optimal intervention and the true underlying graph.

Chapter 9

Conclusions and Future Work

This thesis addressed the problem of developing an integrated framework for accurate estimation and selection of actions in a causal system. The problem was tackled through the Bayesian framework which allows probabilistic reasoning while handling uncertainty in a principled manner. We considered Gaussian process (GP) models for both inference and causal decision-making and answered two specific research questions within the higher-level goal:

- how to develop scalable probabilistic models for point data that incorporate structure in the model likelihood and posterior and can be thus used as surrogate models;
- why and how to incorporate causality into sequential decision-making algorithms so as to enable the selection of actions.

We first focused on how to construct flexible and meaningful representations of a system. Particularly, we investigated models for point data as many real-world problems involve events and these types of models present significant methodological and computational challenges. We then studied decision-making algorithms and investigated how, based on complex surrogate models such as those developed in the first part of the thesis, an agent can select actions to perform in a causal system. More specifically, we generalised Bayesian Optimization (BO), Active Learning (AL), and multi-task GP models to deal with causal information. We showed why sequential decision-making algorithms should be equipped with causal knowledge and how one can develop such frameworks integrating different types of data.

More specifically, in Chapter 4 and Chapter 5 we proposed two novel GP modulated Poisson point processes (PPP). Chapter 4 tackled the issue of developing a multi-task model capable of capturing the correlation across different processes while correctly quantifying uncertainty in the presence of missing data. The proposed framework allowed the development of an efficient variational

inference algorithm that is orders of magnitude faster than competing methods and offers the current state-of-the-art performance in modelling multivariate point processes. Still focusing on PPP, Chapter 5 investigated the problem of developing a continuous model that avoids further approximation in the likelihood function and captures the existence of highly dependent variables a-posteriori. Exploiting the properties of the superposition of PPPs, in this chapter we developed a structured variational approximation scheme in the continuous space directly that, avoiding the need for accurate numerical integration over the input space, can be used for problems with higher input dimensionality. The approaches developed in these chapters provided flexible GP models capable of capturing complex data distributions, quantifying uncertainty in a principled way, and enabling fast approximate inference. These are all crucial properties of surrogate models used within decision-making algorithms.

In the second part of the thesis, we studied how probabilistic models, such as those developed in Chapter 4 and Chapter 5, can be combined with an acquisition function to obtain sequential decision-making algorithms. We saw how a causation structure rather than a correlation structure can be incorporated in GP surrogate models allowing us to select actions based on cause-effect relationships. In particular, we developed a causal formulation for BO (Chapter 6 and Chapter 8), AL (Chapter 7) and multi-task GP models (Chapter 7). Chapter 6 offered the first Causal Bayesian Optimization (CBO) framework solving the problem of finding an optimal intervention when there exist causal relationships between the inputs and the output of a target function. Chapter 7 extended multi-task GP models to capture the correlation across functions defined on a causal graph and characterised by different input dimensionality. Using multi-task causal GP models we improved the performance of both AL and the proposed CBO algorithm thus further demonstrating the benefit of using an accurate surrogate model that properly quantifies uncertainty over unknown functions when selecting actions. Finally, Chapter 8 extended CBO to deal with dynamical systems allowing for the selection of a sequence of optimal actions implemented over time. The Dynamic Causal Bayesian Optimization (DCBO) framework generalised the problem of causal global optimization introduced in Chapter 6 to settings where causal effects evolve over time. The performance of DCBO in several stationary and non-stationary scenarios demonstrated his flexibility and applicability to various real-world problems.

9.1 Future Research Directions

There are a variety of directions in which future work could build upon the ideas introduced in this thesis.

A good starting point for extending the model developed in Chapter 4 would be to consider different prior distributions for the mixing weights of the latent functions. For instance, sparse prior constructions would induce a model selection mechanism for the number of latent functions and increase the model interpretability. While the discretization problem of the model proposed in Chapter 4 has been addressed in Chapter 5, the framework in Chapter 5 can only be used for a single task. Extending it to deal with a higher number of tasks remains an open challenge together with the relaxation of the factorization assumption introduced in the approximate posterior distribution.

Multiple open research directions steam from the chapters included in the second part of this thesis. Firstly, there are many other variants of BO that were not tackled in this thesis and could be extended to integrate causal information. For instance, multi-objective BO could be used to jointly maximize different interventional functions or deal with multi-dimensional outputs. An alternative direction would be to extend the framework in Chapter 8 to be non-myopic. Indeed, the current formulation selects actions based on the highest one-step ahead reward. A non-myopic causal BO would be particularly useful in dynamical systems where interventions performed at one time step affect the rewards an agent can obtain at future time steps. Extending DCBO to use a non-myopic acquisition function such as the one proposed in González et al. [2016b] or Jiang et al. [2020] would allow the algorithm to select interventions in terms of multiple-steps ahead reward thus avoiding potential sub-optimal solutions. This direction would require a further study on the connections between different sequential causal decision-making algorithms such as causal Reinforcement Learning, causal Bandits, CBO, and DCBO. For instance, one could link DCBO to causal RL by writing the expected utility in the form of a Bellman equation (as done for non-myopic BO in Jiang et al. [2020]).

Extending causal decision-making frameworks to deal with discrete outputs and more generally non-Gaussian likelihoods is another important challenge. When dealing with non-Gaussian likelihoods, posterior inference is not closed-form and sequential approximation schemes need to be used to update the surrogate models and select actions. For instance, sequential variational schemes could be used within CBO or DCBO to select interventions minimizing the crime counts analysed in Chapter 4 or maximizing the number of taxi trips considered in Chapter 5. Further work is required to combine Poisson point process models, both single-task and multi-task, with BO or AL. Another open question is the extension of these frameworks to deal with data coming from different populations which are potentially associated with different graph structures. Specifically, how can we exploit the intervention functions learned for one

graph to infer those defined on an alternative graph and characterised by a different structural causal model? Last but not least, all the proposed causal decision-making algorithms are based on the assumption of full knowledge of the causal graph. This is often unrealistic, especially in real-world applications. Extending the proposed frameworks to account for uncertainty in the causal structure is a challenging open problem that we are currently investigating in Branchini et al. [2022] and we briefly introduce it here.

9.2 Active Research Direction

As mentioned above, the causal decision-making frameworks proposed in this thesis are based on the assumption of full knowledge of the causal graph which is often unrealistic. In Branchini et al. [2022] we focus on CBO and develop a framework for joint optimization and causal discovery that properly accounts for uncertainty in the graph structure. In particular, we further generalise the optimization problem defined in Chapter 6 and extended in Chapter 7 to settings where the uncertainty on the graph structure is represented by a prior distribution. Specifically, in Branchini et al. [2022], we aim at identifying the interventional variables and values ($\mathbf{X}_s^\star$, and $\mathbf{x}_s^\star$) optimizing the target while learning the underlying true causal graph, denoted by $\mathcal{G}$. Using the notation adopted Chapter 6, we can formally write this goal as:

$$\begin{cases} \mathbf{X}_s^\star, \mathbf{x}_s^\star & = \underset{\mathbf{X}_s \in \mathcal{P}(\mathbf{X}), \mathbf{x} \in D(\mathbf{X}_s)}{\arg\min} \mathbb{E}[Y \mid \mathrm{do}\left(\mathbf{X}_s = \mathbf{x}\right), G = \mathcal{G}] \\ P(G \mid \mathcal{D}^I) & \propto p(\mathcal{D}^I | G) P(G) \end{cases}$$

where $\mathcal{D}^I$ represents the interventional dataset. Even when restricting the attention to a limited number of potential graphs, solving this optimization problem is challenging. Indeed, similarly to the settings discussed in this thesis, evaluating $\mathbb{E}[Y \mid \mathrm{do}\left(\mathbf{X}_s = \mathbf{x}\right), \mathcal{G}]$ requires intervening in the system at a cost and observing its output. In addition, every time we intervene in the system we need to update $P(G \mid \mathcal{D}^I)$ which might be computationally challenging. We thus want to carefully select the interventions in such a way that we identify the optimum faster. In turn, this requires properly accounting for uncertainty in the graph structure. We solve the problem by developing a structured surrogate model, similar to those seen so far, where the prior distribution is a modified version of the causal GP prior defined in Chapter 6 and used for both the DAG-GP model and the DCBO framework.

For each set $\mathbf{X}_s$ we place a GP prior on $f_s(\mathbf{x}) = \mathbb{E}[Y \mid \mathrm{do}\left(\mathbf{X}_s = \mathbf{x}\right), \mathcal{G}]$ and construct an empirical Bayes prior that incorporates the current belief about the graph together with the observational and interventional data. Specifically

we have $f_s(\mathbf{x}) \sim \mathcal{GP}(m_s(\mathbf{x}), k_s(\mathbf{x}, \mathbf{x}'))$ with $m_s(\mathbf{x}) = \hat{\mathbb{E}}[Y \mid \mathrm{do}\,(\mathbf{X}_s = \mathbf{x})]$ and $k_s(\mathbf{x}, \mathbf{x}') = k_{\text{RBF}}(\mathbf{x}, \mathbf{x}') + \hat{\mathbb{V}}[Y \mid \mathrm{do}\,(\mathbf{X}_s = \mathbf{x})]$. More importantly, the mean function $m_s(\mathbf{x})$ approximates $f_s(\mathbf{x})$ and is computed by accounting for the uncertainty in G:

$$\begin{aligned}\hat{\mathbb{E}}[Y \mid \mathrm{do}\,(\mathbf{X}_s = \mathbf{x})] &= \mathbb{E}_{P(G)}\left[\mathbb{E}_{\hat{p}(Y|\mathrm{do}(\mathbf{X}_s=\mathbf{x}),G)}[Y]\right] \\ &= \sum_{g \in R_G} \int Y \hat{p}(Y \mid \mathrm{do}\,(\mathbf{X}_s = \mathbf{x}), G = g) P(G = g) \mathrm{d}Y\end{aligned}$$

where R_G denotes the support of the distribution on G and the interventional distribution $\hat{p}(Y \mid \mathrm{do}\,(\mathbf{X}_s = \mathbf{x}), G = g)$ can be reduced to do-free expression through the rules of do-calculus. Differently from the approaches seen in this thesis, we use the interventional dataset $\mathcal{D}^I$ to update not only the surrogate models on the interventional outputs but also the prior distribution on G so as to compute $P(G|\mathcal{D}^I)$. In turn, this requires defining a prior and a likelihood function for the graph and computing $P(G|\mathcal{D}^I)$ which could present significant computational challenges. Updating $P(G|\mathcal{D}^I)$ affects the prior construction for the surrogate model and consequently the convergence of the algorithm. In order to select appropriate interventions, both in terms of variable and value and for both causal discovery *and* causal global optimization, we develop an entropy-based acquisition function. Inspired by Wang and Jegelka [2017], we define a joint distribution for the graph structure and the optimal target value and we aim at reducing its uncertainty. Notice that here the optimal target value is defined across different intervention sets, potentially of different dimensionality, and different graph structures. We thus factorise the joint distribution into the distributions for the optimal target value associated to each intervention set. This allows us to write a mixture distribution where each component is equivalent to the distribution targeted by Wang and Jegelka [2017] and the weights are probability reflecting our current belief about the best intervention set. These probability terms are similar to those used to represent the current belief about the best arm to pull in a MAB setting. Using this acquisition function and jointly solving the two problems, global optimization and causal discovery, would enable the application of causal sequential decision-making algorithms to various settings where the causal graph is not known, and assuming a wrong causal graph would prevent the identification of the optimal intervention.

Addressing the unknown causal graph settings together with all the important research questions mentioned above would ultimately lead to an integrated framework for accurate estimation and sequential decision-making in a causal system.

Appendix A

Supplementary Material for MCPM

A.1 Derivation of the KL-divergence Term

The KL-divergence terms composing the ELBO can be written as $\mathcal{L}_{\text{kl}}(\boldsymbol{\nu}) = \mathcal{L}^u_{\text{ent}}(\boldsymbol{\nu}_u) + \mathcal{L}^u_{\text{cross}}(\boldsymbol{\nu}_u) + \mathcal{L}^w_{\text{ent}}(\boldsymbol{\nu}_w) + \mathcal{L}^w_{\text{cross}}(\boldsymbol{\nu}_w)$ where each term is given by:

$$\mathcal{L}^u_{\text{cross}}(\boldsymbol{\nu}_u) = \sum_{q=1}^{Q} \left[\log \mathcal{N}(\mathbf{m}_q; \mathbf{0}, \mathbf{K}^q_{zz}) - \frac{1}{2} \text{ tr } (\mathbf{K}^q_{zz})^{-1} \mathbf{S}_q \right] \tag{A.1}$$

$$\mathcal{L}^u_{\text{ent}}(\boldsymbol{\nu}_u) = \frac{1}{2} \sum_{q=1}^{Q} \left[M \log 2\pi + \log |\mathbf{S}_q| + M \right] \tag{A.2}$$

$$\mathcal{L}^w_{\text{cross}}(\boldsymbol{\nu}_w) = \sum_{q=1}^{Q} \left[\log \mathcal{N}(\boldsymbol{\omega}_q; \mathbf{0}, \mathbf{K}^q_{w}) - \frac{1}{2} \text{ tr } (\mathbf{K}^q_{w})^{-1} \boldsymbol{\Omega}_q \right] \tag{A.3}$$

$$\mathcal{L}^w_{\text{ent}}(\boldsymbol{\nu}_w) = \frac{1}{2} \sum_{q=1}^{Q} \left[\, P \log 2\pi + \log |\boldsymbol{\Omega}_q| + P \right], \tag{A.4}$$

When placing an independent prior and approximate posterior over $\mathbf{W}$, the terms $\mathcal{L}^w_{\text{ent}}$ and $\mathcal{L}^w_{\text{cross}}$ get simplified further, reducing the computational cost significantly when a large number of tasks is considered. Here we derive the expressions for Eqs. (A.1)–(A.4). The negative cross-entropy term for $\mathbf{u}$ (Eq. (A.1)) is given by:

$$\begin{aligned} \mathcal{L}^u_{\text{cross}}(\boldsymbol{\nu}_u) &= \mathbb{E}_{q(\mathbf{u}|\boldsymbol{\nu}_u)}[\log p(\mathbf{u})] = \int q(\mathbf{u}|\boldsymbol{\nu}_u) \log p(\mathbf{u}) \mathrm{d}\mathbf{u} \\ &= \sum_{q=1}^{Q} \int q(\mathbf{u}_{\bullet q}|\boldsymbol{\nu}_u) \log p(\mathbf{u}_{\bullet q}) \mathrm{d}\mathbf{u}_{\bullet q} \end{aligned}$$

$$= \sum_{q=1}^{Q} [\mathcal{N}(\mathbf{u}_{\bullet q}; \mathbf{m}_q, \mathbf{S}_q) \log \mathcal{N}(\mathbf{u}_{\bullet q}; \mathbf{0}, \mathbf{K}_{zz}^q)]$$
$$= \sum_{q=1}^{Q} \left[\log \mathcal{N}(\mathbf{m}_q; \mathbf{0}, \mathbf{K}_{zz}^q) - \frac{1}{2} \text{ tr } (\mathbf{K}_{zz}^q)^{-1} \mathbf{S}_q \right].$$

The entropy term for $\mathbf{u}$ (Eq. (A.2)) is given by:

$$\mathcal{L}_{\text{ent}}^u(\boldsymbol{\nu}_u) = -\mathbb{E}_{q(\mathbf{u}|\boldsymbol{\nu}_u)}[\log q(\mathbf{u}|\boldsymbol{\nu}_u)] = -\int q(\mathbf{u}|\boldsymbol{\nu}_u) \log q(\mathbf{u}|\boldsymbol{\nu}_u) \mathrm{d}\mathbf{u}$$
$$= -\sum_{q=1}^{Q} \int q(\mathbf{u}_{\bullet q}|\boldsymbol{\nu}_u) \log q(\mathbf{u}_{\bullet q}|\boldsymbol{\nu}_u) \mathrm{d}\mathbf{u}_{\bullet q}$$
$$= -\sum_{q=1}^{Q} \int \mathcal{N}(\mathbf{u}_{\bullet q}; \mathbf{m}_q, \mathbf{S}_q) \log \mathcal{N}(\mathbf{u}_{\bullet q}; \mathbf{m}_q, \mathbf{S}_q) \mathrm{d}\mathbf{u}_{\bullet q}$$
$$= -\sum_{q=1}^{Q} \left[\mathcal{N}(\mathbf{m}_q; \mathbf{m}_q, \mathbf{S}_q) - \frac{1}{2} \text{ tr } (\mathbf{S}_q)^{-1} \mathbf{S}_q \right]$$
$$= \frac{1}{2} \sum_{q=1}^{Q} [M \log 2\pi + \log |\mathbf{S}_q| + M].$$

When placing a coupled prior on the mixing weights, the negative cross-entropy term for $\mathbf{W}$ (Eq. (A.3)) is given by:

$$\mathcal{L}_{\text{cross}}^w(\boldsymbol{\nu}_w) = \mathbb{E}_{q(\mathbf{W}|\boldsymbol{\nu}_w)}[\log p(\mathbf{W})] = \int q(\mathbf{W}|\boldsymbol{\nu}_w) \log p(\mathbf{W}) \mathrm{d}\mathbf{W}$$
$$= \sum_{q=1}^{Q} \int q(\mathbf{W}_{\bullet q}|\boldsymbol{\nu}_w) \log p(\mathbf{W}_{\bullet q}) \mathrm{d}\mathbf{W}_{\bullet q}$$
$$= \sum_{q=1}^{Q} \int \mathcal{N}(\boldsymbol{\omega}_q, \boldsymbol{\Omega}_q) \log \mathcal{N}(\mathbf{0}, \mathbf{K}_w^q)) \mathrm{d}\mathbf{W}_{\bullet q}$$
$$= \sum_{q=1}^{Q} \left[\log \mathcal{N}(\boldsymbol{\omega}_q; \mathbf{0}, \mathbf{K}_w^q) - \frac{1}{2} \text{ tr } (\mathbf{K}_w^q)^{-1} \boldsymbol{\Omega}_q \right].$$

The entropy term for $\mathbf{W}$ (Eq. (A.4)) is given by:

$$\mathcal{L}_{\text{ent}}^w(\boldsymbol{\nu}_w) = -\int q(\mathbf{W}|\boldsymbol{\nu}_w) \log q(\mathbf{W}|\boldsymbol{\nu}_w) \mathrm{d}\mathbf{W}$$
$$= -\sum_{q=1}^{Q} \int \mathcal{N}(\mathbf{W}_{\bullet q}; \boldsymbol{\omega}_q, \boldsymbol{\Omega}_q) \log \mathcal{N}(\mathbf{W}_{\bullet q}; \boldsymbol{\omega}_q, \boldsymbol{\Omega}_q) \mathrm{d}\mathbf{W}_{\bullet q}$$

$$= -\sum_{q=1}^{Q}\left[\mathcal{N}(\boldsymbol{\omega}_q;\boldsymbol{\omega}_q,\boldsymbol{\Omega}_q) - \frac{1}{2}\ \mathrm{tr}\ (\boldsymbol{\Omega}_q)^{-1}\boldsymbol{\Omega}_q\right]$$
$$= \frac{1}{2}\sum_{q=1}^{Q}\left[P\log 2\pi + \log|\boldsymbol{\Omega}_q| + P\right].$$

When placing an independent prior and approximate posterior over $\mathbf{W}$, the terms $\mathcal{L}_{\text{ent}}^{w}$ and $\mathcal{L}_{\text{cross}}^{w}$ get further simplified as follow:

$$\begin{aligned}\mathcal{L}_{\text{ent}}^{w}(\boldsymbol{\nu}_w) &= -\int q(\mathbf{W}|\boldsymbol{\nu}_w)\log q(\mathbf{W}|\boldsymbol{\nu}_w)\mathrm{d}\mathbf{W}\\ &= -\sum_{q=1}^{Q}\sum_{p=1}^{P}\int \mathcal{N}(\omega_{pq},\Omega_{pq})\log\mathcal{N}(\omega_{pq},\Omega_{pq})\mathrm{d}w_{pq}\\ &= \frac{1}{2}\sum_{q=1}^{Q}\sum_{p=1}^{P}\left[\log 2\pi + \log\Omega_{pq} + 1\right],\end{aligned} \tag{A.5}$$

$$\begin{aligned}\mathcal{L}_{\text{cross}}^{w}(\boldsymbol{\nu}_w) &= \int q(\mathbf{W}|\boldsymbol{\nu}_w)\log p(\mathbf{W})\mathrm{d}\mathbf{W}\\ &= \sum_{q=1}^{Q}\sum_{p=1}^{P}\int q(w_{pq}|\boldsymbol{\nu}_w)\log p(w_{pq})\mathrm{d}w_{pq}\\ &= \sum_{q=1}^{Q}\sum_{p=1}^{P}\int \mathcal{N}(\omega_{pq},\Omega_{pq})\log\mathcal{N}(0,\sigma_{pq}^2)\mathrm{d}w_{pq}\\ &= \sum_{q=1}^{Q}\sum_{p=1}^{P}\left[\log\mathcal{N}(\omega_{pq};\mathbf{0},\Omega_{pq}) - \frac{\Omega_{pq}}{2\sigma_{pq}^2}\right],\end{aligned} \tag{A.6}$$

where Ω_{pq} represents the p-th diagonal term of $\boldsymbol{\Omega}_q$.

A.2 Closed form evaluation of $\mathcal{L}_{\text{ell}}$

The MCPM model formulation allows deriving a closed-form expression for the moments of the intensity function. Here we provide details about the derivations and obtain an expression for the first moment of $\exp(\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet})$ which is used in the closed-form evaluation of $\mathcal{L}_{\text{ell}}$. In order to compute the moments of λ, we can exploit the moment generating function (MGF) of the product of two normal random variables. Denote by X and Y two independent and normally

distributed random variables. The variable $Z = XY$ has $\text{MGF}_Z(t)$ defined as:

$$\text{MGF}_Z(t) = \frac{\exp\left[\frac{t\mu_X\mu_Y + 1/2(\mu_Y^2\sigma_X^2 + \mu_X^2\sigma_Y^2)t^2}{1-t^2\sigma_X^2\sigma_Y^2}\right]}{\sqrt{1-t^2\sigma_X^2\sigma_Y^2}}. \tag{A.7}$$

Now define the random variable $V = \sum_{q=1}^{Q} X_q Y_q$ where $X_q \perp\!\!\!\perp Y_q, \forall q$, $X_q \perp\!\!\!\perp X_{q'}$, $\forall q, q'$ and $Y_q \perp\!\!\!\perp Y'_{q'}$, $\forall q, q'$. Given these assumptions, the MGF for V is defined as the product of Q MGF of the form given in Eq. (A.7). We have $\text{MGF}_V(t) = \prod_{q=1}^{Q} \text{MGF}_{Z_q}(t)$. This implies that:

$$\mathbb{E}(\lambda_p) = \mathbb{E}\left[\exp(\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet})\right] = \text{MGF}_V(1) \tag{A.8}$$

where $X_q = \omega_{pq}$ and $Y_q = f_{nq}$. Exploiting Eq. (A.8) we can derive a closed form expression for $\mathcal{L}_{\text{ell}}$:

$$\begin{aligned}
&\mathbb{E}_{q(\mathbf{f}),q(\mathbf{W})}\left[\log(p(\mathbf{Y}|\mathbf{f},\mathbf{W}))\right] = \\
&= -\sum_{n=1}^{N}\sum_{p=1}^{P}\mathbb{E}\left[\exp(\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet}+\phi_p) + y_{np}\log(\exp(\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet}+\phi_p)\right. \\
&\left.+\log\Gamma(y_{np}+1)\right] \\
&= \sum_{n=1}^{N}\sum_{p=1}^{P}\mathbb{E}\left[-\exp(\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet}+\phi_p)y_{np}\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet} + y_{np}\phi_p\right. \\
&\left.+\log\Gamma(y_{np}+1)\right] \\
&= -\sum_{n=1}^{N}\sum_{p=1}^{P}\mathbb{E}\left[\exp(\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet}+\phi_p)\right] + \sum_{n=1}^{N}\sum_{p=1}^{P}\left[y_{np}\mathbb{E}(\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet})+\right. \\
&\left.+y_{np}\phi_p + \log\Gamma(y_{np}+1)\right] \\
&= -\sum_{n=1}^{N}\sum_{p=1}^{P}\exp(\phi_p)\text{MGF}_V(1)+ \\
&\sum_{n=1}^{N}\sum_{p=1}^{P}\sum_{q=1}^{Q}(y_{np}\omega_{pq}\mu_q(\mathbf{x}_n) + y_{np}\phi_p + \log\Gamma(y_{np}+1))
\end{aligned} \tag{A.9}$$

Given the moments of $q(\mathbf{W}_{p\bullet})$ and $q(\mathbf{f}_{n\bullet})$ we can write:

$$\mathbb{E}_{q(\mathbf{f}_{n\bullet})q(\mathbf{W}_{p\bullet})}\left[\exp(\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet})\right] = \prod_{q=1}^{Q}\frac{\exp\left[\frac{\omega_{pq}\mu_{nq} + 1/2(\mu_{nq}^2\Omega_{pq} + \omega_{pq}^2\Sigma_{nn}^q)}{1-\Omega_{pq}\Sigma_{nn}^q}\right]}{\sqrt{1-\Omega_{pq}\Sigma_{nn}^q}} \tag{A.10}$$

Defining $\delta_X = \mu_X/\sigma_X$ in Eq. (A.7) we can rewrite $\text{MGF}_Z(t)$ as:

$$\text{MGF}_Z(t) = \frac{\exp\left[\frac{t\mu_X\mu_Y + 1/2(\mu_Y^2\sigma_X^2 + \mu_X^2\sigma_Y^2)t^2}{1 - t^2\frac{\mu_X^2\sigma_Y^2}{\delta_X^2}}\right]}{\sqrt{1 - t^2\frac{\mu_X^2\sigma_Y^2}{\delta_X^2}}} \tag{A.11}$$

As δ_X increases, $\text{MGF}_Z(t)$ converges to the form:

$$\text{MGF}_Z(t) = \exp\left[t\mu_X\mu_Y + 1/2(\mu_Y^2\sigma_X^2 + \mu_X^2\sigma_Y^2)t^2\right] \tag{A.12}$$

which is the MGF of a Gaussian distribution with mean and variance given by $\mu_X\mu_Y$ and $\mu_Y^2\sigma_X^2 + \mu_X^2\sigma_Y^2$ respectively [Seijas-Macías and Oliveira, 2012]. This implies that for increasing values of δ_{X_q} the sum of the products of Gaussians tends to a Gaussian distribution.

A.3 Relationship to existing literature

As mentioned in Appendix A.2, when $\mathbf{\Omega}_q \to 0$, $\mathbf{W}_{p\bullet}\mathbf{f}_{n\bullet}$ converges to a Gaussian distribution. Depending on the number of latent GPs included in the model (Q) and the moments of $q(\mathbf{W}_{p\bullet})$, MCPM will thus converge either to an ICM (or LCM) or to a MLGCP or to an LGCP. When $Q \neq P$, we have $\log\lambda_p(\mathbf{x}^{(n)}) = \sum_{q=1}^{Q}\omega_{pq}\mathbf{f}_{n\bullet}$ for each n and p. We can thus write:

$$\lim_{\mathbf{K}_w \to 0}\text{Cov}(\log\lambda_p(\mathbf{x}), \log\lambda_{p'}(\mathbf{x}')) = \sum_{q,q'}\omega_{pq}\omega_{p'q'}\text{Cov}(\mathbf{f}_{\bullet q}, \mathbf{f}_{\bullet q'}) = \sum_{q,q'}\underbrace{\omega_{pq}\omega_{p'q'}}_{\boldsymbol{B}_q}\widetilde{\mathbf{K}}^q_{\mathbf{x}\mathbf{x}'}$$

where we have exploited the independence assumption between $\mathbf{f}_{\bullet q}$ and $\mathbf{f}_{\bullet q'}$ for $q \neq q'$. When $Q = P + 1$ and $\mathbf{W}_{P\times(P+1)} = [\boldsymbol{I}_P \ \mathbb{1}_P]$, the intensity for each task is determined by the $(P+1)$-th common GP and by the p-th task specific GP. We thus recover the MLGCP formulation. Finally, when $Q = P$ and $\mathbf{W}_{P\times P} = \boldsymbol{I}_P$, the intensity for each task is determined only by the p-th task specific GP. We thus recover the LGCP formulation. We summarize these results in the following theorem:

Theorem A.1. MCPM *generalizes* ICM, MLGCP *and* LGCP. *As* $Cov(w_{pq}, w_{p'q'}) \to 0, \forall p, q, p', q'$, *for* $Q \neq P$ *we have* $\hat{\lambda}_{\text{MCPM}} \to \hat{\lambda}_{\text{ICM}}$ *(or a* $\hat{\lambda}_{\text{MCPM}} \to \hat{\lambda}_{\text{LCM}}$ *depending on the assumed covariance functions for the latent* GP*s) where the intensity parameters are jointly determined by the moments of* $\mathbf{f}$ *and* $\mathbf{W}$*:*

$$\lim_{\substack{Cov(w_{pq}, w_{p'q'})\to 0 \\ \forall p,q,p',q'}}\text{Cov}(\log\lambda_p(\mathbf{x}), \log\lambda_{p'}(\mathbf{x}')) = \sum_{q=1}^{Q}\underbrace{\gamma_{pq}\gamma_{p'q'}}_{\boldsymbol{B}_{q_{(p,p')}}}\widetilde{\mathbf{K}}^q_{\mathbf{x}\mathbf{x}'}$$

where $\boldsymbol{B}_q \in \mathbb{R}^{P\times P}$ *is known as coregionalisation matrix. For* $Q = P + 1$ *and*

$\mathbf{W}_{P\times(P+1)} = [\boldsymbol{I}_P \ \mathbb{1}_P]$ *we have* $\hat{\lambda}_{\text{MCPM}} \to \hat{\lambda}_{\text{MLGCP}}$. *Finally, for* $Q = P$ *and* $\mathbf{W}_{P\times P} = \boldsymbol{I}_P$ *we have* $\hat{\lambda}_{\text{MCPM}} \to \hat{\lambda}_{\text{LGCP}}$.

When considering the task descriptors $\mathbf{h}$, we can view the log intensity as a function of the joint space of input features and task descriptors *i.e.* $\log \lambda(\mathbf{x}, \mathbf{h})$. It is possible to show that under our independence prior assumption between weights ($\mathbf{W}$) and latent functions ($\mathbf{f}$), the prior covariance over the log intensities (evaluated at inputs $\mathbf{x}$ and $\mathbf{x}'$ and tasks p and p') is given by:

$$\mathbb{C}\text{ov}[\log \lambda_p(\mathbf{x}), \lambda_{p\prime}(\mathbf{x}')] = \sum_{q=1}^{Q} \kappa_w^q(\mathbf{h}^p, \mathbf{h}^{p'})\kappa_f^q(\mathbf{x}, \mathbf{x}')$$

where $\mathbf{h}^p$ denotes the p-th task descriptors. At the observed data $\{\mathbf{X}, \mathbf{H}\}$, assuming a regular grid, the MCPM prior covariance over the log intensities is $\mathbb{C}\text{ov}[\log \boldsymbol{\lambda}(\mathbf{X}), \log \boldsymbol{\lambda}(\mathbf{X})] = \sum_{q=1}^{Q} \mathbf{K}_w^q \otimes \mathbf{K}_f^q$. This is effectively the LCM prior with $\mathbf{K}_w^q$ denoting the coregionalization matrix. Importantly, the two methods differ substantially in terms of inference. While in LCM a point estimate of $\mathbf{K}_w^q$ is generally obtained, MCPM proceeds by optimizing the hyperparameters for $\mathbf{K}_w^q$ and doing full posterior estimation for both $\mathbf{W}$ and $\mathbf{f}$. By adopting a process view on $\mathbf{W}$, we increase the model flexibility and accuracy by capturing additional correlations across tasks while being able to generalize over unseen task descriptors. Finally, by considering our priors and approximate posteriors over $\mathbf{W}$ and $\mathbf{f}$ separately, instead of a single joint prior over the log intensities, we can exploit state-of-the-art inducing variable approximations [Titsias, 2009b] over each $\mathbf{W}_{\bullet q}$ and $\mathbf{f}_{\bullet q}$ separately, instead of dealing with a sum of Q Kronecker products for which there is not an efficient decomposition when $Q > 2$ [Rakitsch et al., 2013].

A.4 Algorithmic efficiency

Evaluating $\mathcal{L}_{\text{ell}}$ in closed form, we are able to significanlty speed up the algorithm by getting rid of the Monte Carlo approximations in the ELBO evaluations, see Fig. A.1 and Fig. A.2.

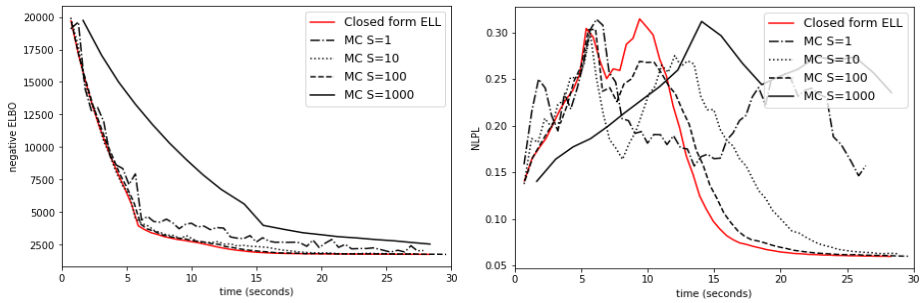


Figure A.1: Synthetic data. Monte Carlo approximation vs. closed form evaluation of $\mathcal{L}_{\text{ell}}$. *Left:* Negative ELBO values over time. *Right:* NLPL values for one task over time. S denotes the number of samples used in the Monte Carlo evaluation.

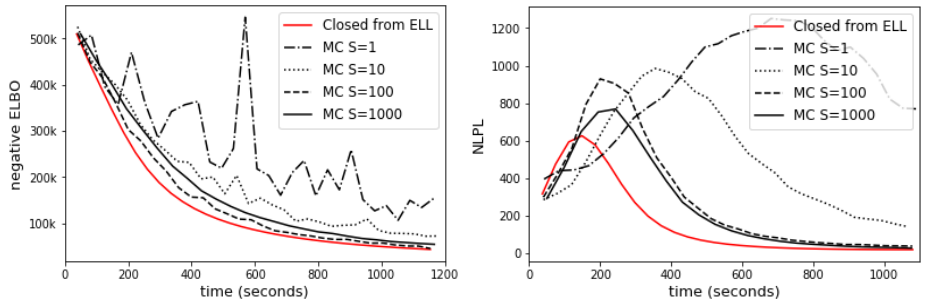


Figure A.2: CRIME data. Monte Carlo approximation vs. closed form evaluation of $\mathcal{L}_{\text{ell}}$. *Left:* Negative ELBO values over time. *Right:* NLPL values for one task over time. S denotes the number of samples used in the Monte Carlo evaluation.

A.5 Additional experimental results

Synthetic experiments We report additional performance metrics for the two synthetic experiments included in the text. Table A.1 gives the coverage numbers for the first synthetic experiment (S1) while Table A.2 and Table A.3 display the RMSE and coverage performances for the second synthetic dataset (S2). Finally, Fig. A.3 gives the predicted counts distributions for S2.

Crime data experiments We report the RMSE values for MCPM and competing models on the CRIME dataset (Table A.4). In Fig. A.4 and Fig. A.5 we give the estimated intensities and conditional probabilities for the CRIME complete data experiment. Finally, in Fig. A.6 we show the conditional probabilities for the missing data experiment.

Table A.1: S1 dataset. In-sample/Out-of-sample 90% CI coverage for the predicted event counts distributions.

	Empirical Coverage (EC)			
	1	**2**	**3**	**4**
MCPM-N	0.80/0.12	**0.99**/0.58	0.92/0.57	**0.94**/**0.83**
MCPM-GP	**0.95**/**0.19**	0.72/**0.67**	**1.00**/**0.78**	0.92/0.75
ICM	0.75/0.03	0.66/0.60	0.62/0.50	0.93/0.42

Table A.2: S2 dataset. RMSE performance when making predictions on the interval $[80, 100]$.

	RMSE									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
MCPM-N	**1.10**	**1.15**	**0.89**	0.17	0.95	0.99	1.10	0.63	1.50	0.55
MCPM-GP	1.15	1.43	0.91	**0.13**	0.94	**0.97**	1.19	**0.58**	**1.43**	0.70
MTPP	1.20	1.70	1.12	0.17	**0.91**	1.05	**1.05**	1.11	1.61	**0.49**

BTB data experiments In Fig. A.9 we show the estimated conditional probabilities on the origin color scale used by Diggle et al. [2013]. In Fig. A.7 we give the estimated intensity surfaces for the complete data experiment. Finally, in Fig. A.8 we show the estimated intensity surfaces for the missing data experiment.

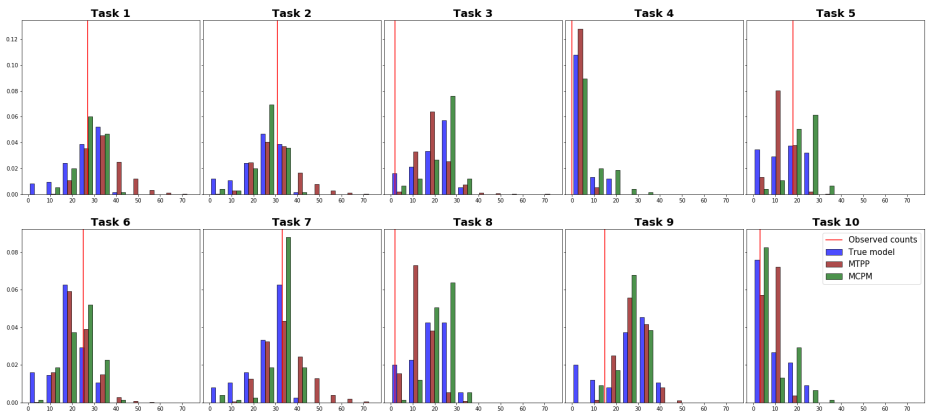


Figure A.3: Predicted empirical distributions of event counts in $[80, 100]$ for S2.

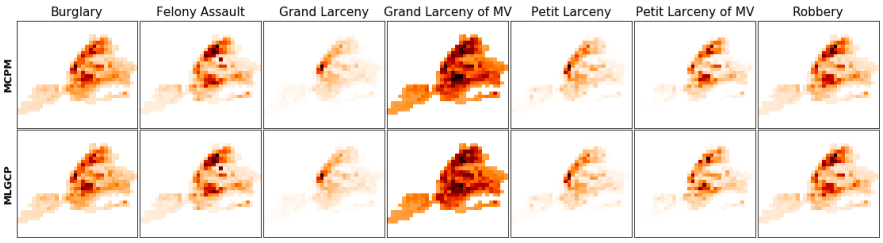


Figure A.4: CRIME dataset. Estimated intensity surface with MCPM (first row) and MLGCP (second row). The color scale used is given in Fig. (5).

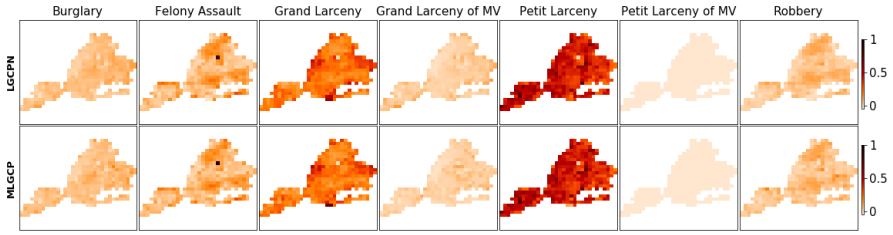


Figure A.5: CRIME dataset. Estimated conditional probabilities in the complete data setting. *Row 1:* MCPM *Row 2:* MLGCP.

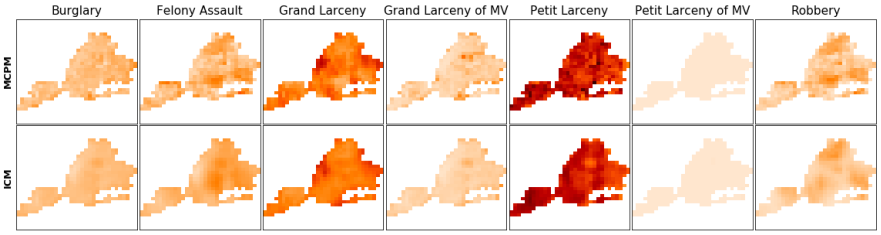


Figure A.6: CRIME dataset. Estimated conditional probabilities when introducing missing data regions. *Row 1:* MCPM *Row 2:* LGCP.

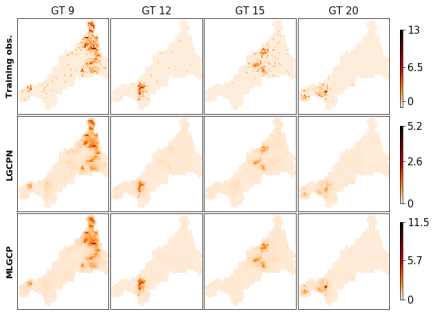


Figure A.7: Estimated intensity surfaces in the complete data setting. *First row:* Training data. *Second row:* MCPM *Third row:* MLGCP

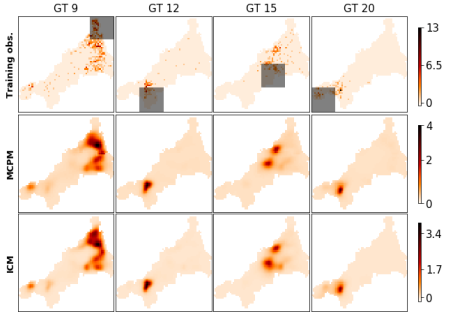


Figure A.8: Estimated intensity surfaces in the missing data (shaded regions) setting. *First row:* Training data. *Second row:* MCPM *Third row:* ICM

Table A.3: S2 dataset. In-sample/Out-of-sample 90% CI coverage for the predicted event counts distributions.

	Empirical Coverage									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
MCPM-N	**1.00**/**1.00**	**1.00**/**1.00**	0.95/0.99	0.66/**1.00**	**1.00**/0.86	0.97/**1.00**	**0.99**/**1.00**	0.88/**1.00**	0.92/0.95	**1.00**/**1.00**
MCPM-GP	0.99/**1.00**	**1.00**/**1.00**	**1.00**/**1.00**	**1.00**/**1.00**	**1.00**/**1.00**	**1.00**/**1.00**	**0.99**/**1.00**	**1.00**/**1.00**	**1.00**/**1.00**	**1.00**/**1.00**
MTPP	0.77/0.77	0.82/0.73	0.86/**1.00**	0.93/**1.00**	0.75/0.83	0.96/0.84	0.78/0.54	0.99/**1.00**	0.66/0.88	0.74/0.95

Table A.4: CRIME dataset. Performance on the missing regions. Standard errors in parentheses.

	Standardized RMSE						
	1	**2**	**3**	**4**	**5**	**6**	**7**
MCPM	1.74	2.91	**3.00**	**2.75**	3.57	**11.70**	**1.54**
	(0.42)	(1.06)	(1.22)	(0.82)	(1.99)	(2.32)	(0.29)
MCPM-GP	**1.71**	**1.91**	3.40	2.96	**2.00**	12.18	1.62
	(0.39)	(0.33)	(1.80)	(1.03)	(0.47)	(2.76)	(0.33)
LGCP	5.16	4.68	8.93	3.09	7.69	36.96	5.19
	(1.81)	(0.99)	(5.22)	(0.50)	(3.68)	(5.43)	(1.21)
ICM	3.36	3.64	3.70	2.97	3.05	12.36	2.82
	(1.04)	(0.83)	(1.89)	(1.22)	(0.97)	(1.99)	(0.62)

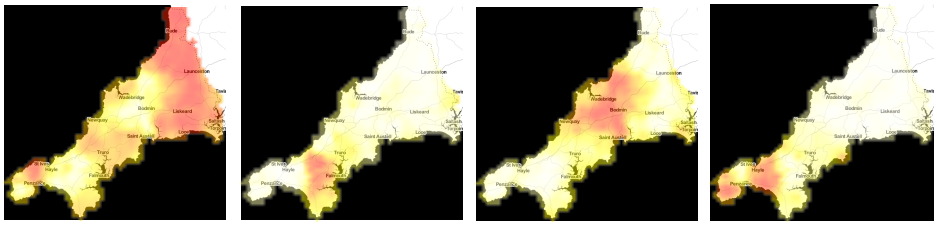


Figure A.9: MLGCP- BTB dataset. Estimated conditional probabilities plotted on the color scale used by Diggle et al. [2013] and Taylor et al. [2015]. The first plots corresponds to GT 9, the second to GT 12, the third to GT 15 and the fourth to GT 20.

Appendix B

Supplementary Material for STVB

B.1 ELBO derivations

Here we derive the expressions given in Eq. (5.7) and Eqs. (5.8)–(5.12). Starting with Eq. (5.7), the evidence lower bound ($\mathcal{L}_{\text{elbo}}$) can be written as:

$$
\begin{aligned}
\mathcal{L}_{\text{elbo}} &= \mathbb{E}_Q\left[\log\left[\frac{p(\{\mathbf{x}_n\}_{n=1}^N, \{\mathbf{y}_k\}_{k=1}^K, K, \mathbf{f}, \mathbf{u}, \lambda^\star|\tau, \boldsymbol{\theta})}{p(\mathbf{f}|\mathbf{u})q(\{\mathbf{y}_k\}_{k=1}^K|K)q(K|\mathbf{u}, \lambda^\star)q(\mathbf{u})q(\lambda^\star)}\right]\right] \\
&= \mathbb{E}_Q\left[\log p(\{\mathbf{x}_n\}_{n=1}^N, \{\mathbf{y}_k\}_{k=1}^K, K, \mathbf{u}, \lambda^\star|\tau, \boldsymbol{\theta})\right] \\
&\quad - \mathbb{E}_Q\left[\log q(\{\mathbf{y}_k\}_{k=1}^K|K)q(K|\mathbf{u}, \lambda^\star)q(\mathbf{u})q(\lambda^\star)\right] \\
&= \mathbb{E}_Q\left[(N+K)\log(\lambda^\star) - \lambda^\star\mu(\tau) - \log(K!) - \log(N!) + \sum_{n=1}^N \log(\sigma(\mathbf{f}(\mathbf{x}_n)))\right] \\
&\quad + \mathbb{E}_Q\left[\sum_{k=1}^K \log(\sigma(-\mathbf{f}(\mathbf{y}_k))) + \log(p(\mathbf{u})) + \log(p(\lambda^\star))\right] \\
&\quad - \mathbb{E}_Q\left[\log(q(\mathbf{u})) - \log(q(K|\mathbf{u}, \lambda^\star)) - \log(q(\lambda^\star)) - \log(q(\{\mathbf{y}_k\}_{k=1}^K|K))\right] \\
&= N(\psi(\alpha) - \log(\beta)) - V\frac{\alpha}{\beta} - \log(\log N!) + \underbrace{\mathbb{E}_Q[K\log(\lambda^\star)]}_{T_1} - \underbrace{\mathbb{E}_Q[\log K!]}_{T_2} + \\
&\quad + \sum_{n=1}^N \mathbb{E}_{q(\mathbf{u})}[\log(\sigma(\mathbf{f}(\mathbf{x}_n)))] + \underbrace{\mathbb{E}_Q\left[\sum_{k=1}^K \log(\sigma(-\mathbf{f}(\mathbf{y}_k)))\right]}_{T_3} + \\
&\quad - KL(q(\mathbf{u})||p(\mathbf{u})) - KL(q(\lambda^\star)||p(\lambda^\star)) \\
&\quad - \underbrace{\mathbb{E}_Q[\log q(K|\mathbf{u}, \lambda^\star)]}_{T_4} - \underbrace{\mathbb{E}_Q\left[\log q(\{\mathbf{y}_k\}_{k=1}^K|K)\right]}_{T_5}
\end{aligned}
$$

Let's now focus on the terms T_i for $i = 1, ..., 5$.

The term T_1 (Eq. (5.8)) is given by:

$$\begin{aligned}
T_1 &= \mathbb{E}_{q(\mathbf{u})q(\lambda^\star)}\big[\mathbb{E}_{q(K|\mathbf{u},\lambda^\star)}[K\log(\lambda^\star)]\big] \\
&= \mathbb{E}_{q(\mathbf{u})q(\lambda^\star)}\big[\log(\lambda^\star)\mathbb{E}_{q(K|\mathbf{u},\lambda^\star)}[K]\big] \\
&= \mathbb{E}_{q(\mathbf{u})q(\lambda^\star)}\left[\log(\lambda^\star)\lambda^\star\int_\tau \sigma(-\mathbf{u}(x))dx\right] \\
&= \mathbb{E}_{q(\lambda^\star)}[\lambda^\star\log(\lambda^\star)]\mathbb{E}_{q(\mathbf{u})}[\mu(\mathbf{u})]
\end{aligned}$$

The term T_3 (Eq. (5.9)) is given by:

$$\begin{aligned}
T_3 &= \mathbb{E}_{q(\mathbf{f})q(\mathbf{u})q(\mathbf{y}_k)q(\lambda^\star)}\left[\mathbb{E}_{q(K|\mathbf{f},\mathbf{y}_k)}\left[\sum_{k=1}^{K}\log(\sigma(-\mathbf{f}(\mathbf{y}_k)))\right]\right] \\
&= \mathbb{E}_{q(\mathbf{f})q(\mathbf{u})q(\mathbf{y}_k)q(\lambda^\star)}\left[\log(\sigma(-\mathbf{f}(\mathbf{y}_k)))\mathbb{E}_{q(K|\mathbf{f},\mathbf{y}_k)}\left[\sum_{k=1}^{K}1\right]\right] \\
&= \mathbb{E}_{q(\mathbf{f})q(\mathbf{u})q(\mathbf{y}_k)}[\log(\sigma(-\mathbf{f}(\mathbf{y}_k)))\lambda^\star\mu(\mathbf{u})] \\
&= \frac{\alpha}{\beta}\mathbb{E}_{q(\mathbf{u})}[\mu(\mathbf{u})]\mathbb{E}_{q(\mathbf{f})q(\mathbf{y}_k)}[\log(\sigma(-\mathbf{f}(\mathbf{y}_k)))]
\end{aligned}$$

The term T_4 (Eq. (5.10)) is given by:

$$\begin{aligned}
T_4 &= \mathbb{E}_Q[-\lambda^\star\mu(\mathbf{u})] + \mathbb{E}_Q[K\log(\lambda^\star\mu(\mathbf{u}))] - \mathbb{E}_Q[\log K!] \\
&= -\frac{\alpha}{\beta}\mathbb{E}_{q(\mathbf{u})}[\mu(\mathbf{u})] + \mathbb{E}_{q(\lambda^\star)q(\mathbf{u})}[\lambda^\star\log(\lambda^\star)\mu(\mathbf{u}) + \lambda^\star\mu(\mathbf{u})\log(\mu(\mathbf{u}))] + \\
&\quad - \mathbb{E}_Q[\log K!] \\
&= -\frac{\alpha}{\beta}\mathbb{E}_{q(\mathbf{u})}[\mu(\mathbf{u})] + \mathbb{E}_{q(\lambda^\star)}[\lambda^\star\log(\lambda^\star)]\mathbb{E}_{q(\mathbf{u})}[\mu(\mathbf{u})] + \\
&\quad + \frac{\alpha}{\beta}\mathbb{E}_{q(\mathbf{u})}[\mu(\mathbf{u})\log(\mu(\mathbf{u}))] - \mathbb{E}_Q[\log(K!)]
\end{aligned}$$

Finally, the term T_5 (Eq. (5.12)) is given by:

$$\begin{aligned}
T_5 &= \mathbb{E}_Q\left[\sum_{k=1}^{K}\log q(\mathbf{y}_k)\right] = \mathbb{E}_{q(\mathbf{u},\lambda^\star,K)}\left[\sum_{k=1}^{K}\mathbb{E}_{q(\mathbf{y}_k)}[\log q(\mathbf{y}_k)]\right] \\
&= \mathbb{E}_{q(\mathbf{u},\lambda^\star,K)}[K]\mathbb{E}_{q(\mathbf{y}_k)}[\log q(\mathbf{y}_k)] \\
&= \mathbb{E}_{q(\mathbf{u})q(\lambda^\star)}[\lambda^\star\mu(\mathbf{u})]\mathbb{E}_{q(\mathbf{y}_k)}[\log q(\mathbf{y}_k)] \\
&= \frac{\alpha}{\beta}\mathbb{E}_{q(\mathbf{y}_k)}[\log q(\mathbf{y}_k)]\mathbb{E}_{q(\mathbf{u})}[\mu(\mathbf{u})]
\end{aligned}$$

Notice how the last term in T_4 that is $-\mathbb{E}_Q[\log(K!)]$, appears with opposite sign in $T_2 = \mathbb{E}_Q[\log(K!)]$. This term is thus cancelling out in the computation of the ELBO.

B.2 Performance metrics

We test the algorithms evaluating the l_2 norm to the true intensity function (in the synthetic settings), the test log likelihood (ℓ_{test}) on the test set and the negative log predicted likelihood (NLPL) on the training set. The l_2 metric is computed as follow:

$$l_2 = \int_{\mathcal{X}} (\lambda(\mathbf{x}) - \bar{\lambda}(\mathbf{x}))^2 \mathrm{d}\mathbf{x} \tag{B.1}$$

where $\lambda(\mathbf{x})$ is the true intensity function, $\bar{\lambda}(\mathbf{x})$ is the posterior mean intensity and the integral is evaluated numerically. The ℓ_{test} values are given by:

$$\ell_{test} = \mathbb{E}_{q(\lambda^\star)q(\mathbf{f})}\left[\log\left[\exp\left(-\int_{\mathcal{X}} \lambda(\mathbf{x}) d\mathbf{x}\right) \prod_{\mathbf{x}\in\mathcal{D}_{\text{test}}} \lambda(\mathbf{x})\right]\right] \tag{B.2}$$

where again the integral is computed via numerical integration. The NLPL metric is computed as:

$$\text{NLPL} = -\frac{1}{S}\sum_{s=1}^{S} \log p(N_{\text{train}}| \int_{\mathcal{X}} \lambda^s(\mathbf{x}) \mathrm{d}(\mathbf{x})) \tag{B.3}$$

where S denotes the number of samples from the variational distributions $q(\mathbf{f})$ and $q(\lambda^\star)$. Finally, the EC is computed by evaluating the coverage of the CIs of the posterior ($p(N|\mathcal{D})$) and predictive ($p(N^*|\mathcal{D})$) distributions. To construct the empirical count distribution we sample from the variational distributions $q(\mathbf{f})$ and $q(\mathbf{W})$, obtain samples of $\lambda(\mathbf{x})$ and simulate the number of events N or N^* from Poisson($\lambda^\star \int_{\mathcal{X}} \sigma(\mathbf{f}(\mathbf{x}))d\mathbf{x}$).

B.3 Additional experimental results

For all comparisons we consider a GP with squared-exponential covariance function with equally set hyperparameters. Denote by $\boldsymbol{\theta}_i = (l, \sigma_f^2)$ the values of the hyperameters for the kernel function $K(\mathbf{x}, \mathbf{x}')$ on $\lambda_i(\mathbf{x})$ where l indicates the lenghtscale. For the synthetic experiments we set:

- $\boldsymbol{\theta}_1 = (10, 1)$
- $\boldsymbol{\theta}_2 = (0.25, 1)$
- $\boldsymbol{\theta}_3 = (15, 1)$

For the real-world settings we set the following kernel hyperparameters:

- $\boldsymbol{\theta}_{\text{neuronal data}} = (10, 1)$
- $\boldsymbol{\theta}_{\text{taxi data}} = (0.3, 1)$

Table B.1: $\lambda_1(\mathbf{x})$ - EC performance on training and test dataset. Higher values are better. Standard errors in brackets.

	$\lambda_1(\mathbf{x})$ - In-sample EC					$\lambda_1(\mathbf{x})$ - Out-of-sample EC				
	10% CI	20% CI	30% CI	40% CI	50% CI	10% CI	20% CI	30% CI	40% CI	50% CI
STVB	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	0.96	0.88	**0.81**	**0.72**	**0.60**
	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.24)	(0.24)	(0.23)	(0.29)	(0.29)
MFVB	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	0.95	0.80	0.76	0.61	0.52
	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)
VBPP	**1.00**	**1.00**	**1.00**	**1.00**	0.10	**1.00**	**0.97**	0.75	0.41	0.04
	(0.00)	(0.00)	(0.00)	(0.00)	(0.30)	(0.00)	(0.05)	(0.21)	(0.25)	(0.09)
SGCP	**1.00**	**1.00**	**1.00**	**1.00**	0.60	0.75	0.60	0.39	0.27	0.08
	(0.00)	(0.00)	(0.00)	(0.00)	(0.49)	(0.29)	(0.33)	(0.28)	(0.22)	(0.12)
LGCP	0.70	0.00	0.00	0.00	0.00	0.48	0.22	0.08	0.03	0.01
	(0.46)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)

Table B.2: $\lambda_2(\mathbf{x})$ - EC performance on training and test dataset. Higher values are better. Standard errors in brackets.

	$\lambda_2(\mathbf{x})$ - In-sample EC					$\lambda_2(\mathbf{x})$ - Out-of-sample EC				
	10% CI	20% CI	30% CI	40% CI	50% CI	10% CI	20% CI	30% CI	40% CI	50% CI
STVB	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**0.97**	**0.91**	**0.88**	**0.86**
	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.09)	(0.24)	(0.23)	(0.22)
MFVB	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	0.92	0.92	0.89	0.84	0.82
	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)
VBPP	**1.00**	**1.00**	**1.00**	**1.00**	0.10	0.92	0.86	0.76	0.45	0.05
	(0.00)	(0.00)	(0.00)	(0.00)	(0.30)	(0.24)	(0.23)	(0.26)	(0.26)	(0.05)
SGCP	**1.00**	0.90	0.70	0.40	0.30	0.90	0.90	0.64	0.14	0.00
	(0.00)	(0.30)	(0.46)	(0.49)	(0.46)	(0.00)	(0.00)	(0.09)	(0.05)	(0.00)
LGCP	0.10	0.00	0.00	0.00	0.00	0.80	0.22	0.04	0.00	0.00
	(0.30)	(0.00)	(0.00)	(0.00)	(0.00)	(0.24)	(0.16)	(0.08)	(0.00)	(0.00)

Synthetic data experiments In Table B.1, Table B.2 and Table B.3 we report the values of EC for different CIs and on both the training and test set.

Real data experiments In Table B.4 we report the values of EC for different CIs and on both the training and test set.

Table B.3: $\lambda_3(\mathbf{x})$ - EC performance on training and test dataset. Higher values are better. Standard errors in brackets.

	$\lambda_3(\mathbf{x})$ - In-sample EC					$\lambda_3(\mathbf{x})$ - Out-of-sample EC				
	10% CI	20% CI	30% CI	40% CI	50% CI	10% CI	20% CI	30% CI	40% CI	50% CI
STVB	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	0.99	0.97	0.92
	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.12)
MFVB	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	0.97	0.91	0.78
	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)
VBPP	**1.00**	**1.00**	**1.00**	**1.00**	0.10	0.97	0.94	0.83	0.43	0.03
	(0.00)	(0.00)	(0.00)	(0.00)	(0.30)	(0.09)	(0.15)	(0.19)	(0.14)	(0.05)
SGCP	0.80	0.70	0.50	0.40	0.00	0.82	0.54	0.49	0.34	0.02
	(0.40)	(0.46)	(0.50)	(0.49)	(0.00)	(0.12)	(0.05)	(0.03)	(0.07)	(0.04)
LGCP	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**0.95**
	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)

Table B.4: Real data. Values are given as In-sample - Out-of-sample EC. Mean and standard errors (in parenthesis) are computed across different seeds.

	Neuronal data				
	10% EC	20% EC	30% EC	40% EC	50%EC
STVB	**1.00**-**1.00**	**1.00**-**1.00**	**1.00**-**1.00**	**0.99**-**0.56**	**0.01**- 0.00
	(0.00)-(0.00)	(0.00)- (0.00)	(0.00)-(0.00)	(0.10)-(0.50)	(0.10)-(0.00)
MFVB	**1.00**-**1.00**	**1.00**-0.62	**1.00**-0.03	0.78-0.00	0.00 - 0.00
	(0.00) - (0.00)	(0.00)-(0.49)	(0.00)-(0.17)	(0.41)-(0.00)	(0.00) - (0.00)
VBPP	**1.00**-0.53	**1.00**-0.00	**1.00**-0.00	0.83-0.00	**0.01**-0.00
	(0.00)-(0.50)	(0.00)- (0.00)	(0.00)- (0.00)	(0.38)-(0.00)	(0.10)-(0.00)
	Taxi data				
	10% EC	20%EC	30% EC	40% EC	50% EC
STVB	**1.00**-1.00	**1.00**-**1.00**	0.81-**0.37**	0.09-**0.01**	0.00-0.00
	(0.00)-(0.00)	(0.00)-(0.00)	(0.39)-(0.48)	(0.29)-(0.10)	(0.00)-(0.00)
MFVB	0.49-0.93	0.00-0.13	0.00-0.00	0.00-0.00	0.00-0.00
	(0.50)-(0.26)	(0.00)-(0.34)	(0.00)-(0.00)	(0.00)-(0.00)	(0.00)-(0.00)
VBPP	**1.00**-0.00	**1.00**-0.00	**0.98**-0.00	**0.48**-0.00	0.00- 0.00
	(0.00)- (0.00)	(0.00)-(0.00)	(0.14)-(0.00)	(0.50)-(0.00)	(0.00)-(0.00)

Appendix C

Supplementary Material for CBO

C.1 *Do*-calculus derivations for the toy experiment

In this section we derive the *do*-calculus expressions for all interventions one can implement in the DAG of Fig. 6.3 (left).

$$p(y|\mathrm{do}(X=x)) = p(y|X=x) \qquad \text{by } (Y \perp\!\!\!\perp X) \text{ in } \mathcal{G}_{\underline{X}}$$

$$p(y|\mathrm{do}(Z=z)) = p(y|Z=z) \qquad \text{by } (Y \perp\!\!\!\perp Z) \text{ in } \mathcal{G}_{\underline{Z}}$$

$$p(y|\mathrm{do}(X=x),\mathrm{do}(Z=z)) = p(y|\mathrm{do}(Z=z)) \qquad \text{by } (Y \perp\!\!\!\perp X|Z) \text{ in } \mathcal{G}_{\overline{XZ}}$$

C.2 *Do*-calculus derivations for the synthetic experiment

In this section we derive the *do*-calculus expressions for all interventions one can implement in the DAG of Fig. 6.2 (a).

$$\begin{aligned}
p(y|\mathrm{do}(B=b)) &= \int p(y|c,\mathrm{do}(B=b))p(c|B=b)\mathrm{d}c \\
&= \int p(y|\mathrm{do}(C=c),\mathrm{do}(B=b))p(C=c|B=b)\mathrm{d}c \\
&\quad \text{by } (Y \perp\!\!\!\perp C|B) \text{ in } \mathcal{G}_{\overline{B}\underline{C}} \\
&= \int p(y|\mathrm{do}(C=c))p(c|B=b)\mathrm{d}c \quad \text{by } (Y \perp\!\!\!\perp B|C) \text{ in } \mathcal{G}_{\overline{BC}} \\
&= \int p(y|b',\mathrm{do}(C=c))p(b'|\mathrm{do}(C=c))p(c|B=b)\mathrm{d}b'\mathrm{d}c \\
&= \int p(y|b',C=c)p(b')p(c|B=b)\mathrm{d}b'\mathrm{d}c \quad \text{by } (Y \perp\!\!\!\perp C|B) \text{ in } \mathcal{G}_{\overline{B}\underline{C}}
\end{aligned}$$

$$
\begin{aligned}
p(y|\mathrm{do}(D=d)) &= \int p(y|c,\mathrm{do}(D=d))p(c|\mathrm{do}(D=d))\mathrm{d}c \\
&= \int p(y|c,D=d)p(c)\mathrm{d}c \quad \text{by } (Y \perp\!\!\!\perp D|C) \text{ in } \mathcal{G}_{\underline{D}}
\end{aligned}
$$

$$
\begin{aligned}
p(y|\mathrm{do}(E=e)) &= \int p(y|a,c,\mathrm{do}(E=e))p(a,c|\mathrm{do}(E=e))\mathrm{d}a\mathrm{d}c \\
&= \int p(y|a,c,E=e)p(a)p(c)\mathrm{d}a\mathrm{d}c \quad \text{by } (Y \perp\!\!\!\perp E|A,C) \text{ in } \mathcal{G}_{\underline{E}}
\end{aligned}
$$

$$
\begin{aligned}
p(y|\mathrm{do}(B=b),\mathrm{do}(D=d)) &= \int p(y|\mathrm{do}(B=b),c,\mathrm{do}(D=d))p(c|\mathrm{do}(B=b),\mathrm{do}(D=d))\mathrm{d}c \\
&= \int p(y|\mathrm{do}(B=b),\mathrm{do}(C=c),\mathrm{do}(D=d))p(c|B=b)\mathrm{d}c \\
&\quad \text{by } (Y \perp\!\!\!\perp C|B,D) \text{ in } \mathcal{G}_{\underline{C}\overline{BD}} \\
&= \int p(y|\mathrm{do}(C=c),\mathrm{do}(D=d))p(c|B=b)\mathrm{d}c \\
&\quad \text{by } (Y \perp\!\!\!\perp B|C,D) \text{ in } \mathcal{G}_{\overline{BCD}} \\
&= \int p(y|b',\mathrm{do}(C=c),\mathrm{do}(D=d)) \\
&\qquad \times p(b'|\mathrm{do}(C=c),\mathrm{do}(D=d))p(c|B=b)\mathrm{d}c\mathrm{d}b' \\
&= \int p(y|b',C=c,\mathrm{do}(D=d))p(b')p(c|B=b)\mathrm{d}c\mathrm{d}b' \\
&\quad \text{by } (Y \perp\!\!\!\perp C|B,D) \text{ in } \mathcal{G}_{\overline{BD}\underline{C}} \\
&= \int p(y|b',C=c,D=d)p(b')p(c|B=b)\mathrm{d}c\mathrm{d}b' \\
&\quad \text{by } (Y \perp\!\!\!\perp D|B,C) \text{ in } \mathcal{G}_{\underline{D}}
\end{aligned}
$$

$$
\begin{aligned}
p(y|\mathrm{do}(D=d),\mathrm{do}(E=e)) &= \int p(y|a,c,\mathrm{do}(D=d),\mathrm{do}(E=e)) \\
&\qquad \times p(a,c|\mathrm{do}(D=d),\mathrm{do}(E=e))\mathrm{d}a\mathrm{d}c \\
&= \int p(y|a,c,D=d,E=e)p(a)p(c)\mathrm{d}a\mathrm{d}c \\
&\quad \text{by } (Y \perp\!\!\!\perp (D,E)|A,C) \text{ in } \mathcal{G}_{\underline{DE}}
\end{aligned}
$$

$$
\begin{aligned}
p(y|\mathrm{do}(B=b),\mathrm{do}(E=e)) &= \int p(y|\mathrm{do}(B=b),c,\mathrm{do}(E=e))p(c|B=b)\mathrm{d}c \\
&= \int p(y|\mathrm{do}(B=b),\mathrm{do}(C=c),\mathrm{do}(E=e))p(c|B=b)\mathrm{d}c \\
&\quad \text{by } (Y \perp\!\!\!\perp C|B,E) \text{ in } \mathcal{G}_{\overline{BE}\underline{C}} \\
&= \int p(y|\mathrm{do}(C=c),\mathrm{do}(E=e))p(c|B=b)\mathrm{d}c \\
&\quad \text{by } (Y \perp\!\!\!\perp B|C,E) \text{ in } \mathcal{G}_{\overline{CEB}} \\
&= \int p(y|\mathrm{do}(C=c),\mathrm{do}(E=e),b') \\
&\quad \times p(b'|\mathrm{do}(C=c),\mathrm{do}(E=e))p(c|B=b)\mathrm{d}b'\mathrm{d}c \\
&= \int p(y|C=c,\mathrm{do}(E=e),b')p(b')p(c|B=b)\mathrm{d}b'\mathrm{d}c \\
&\quad \text{by } (Y \perp\!\!\!\perp C|B,E) \text{ in } \mathcal{G}_{\overline{E}\underline{C}} \\
&= \int p(y|a,C=c,\mathrm{do}(E=e),b')p(a|C=c,\mathrm{do}(E=e),b') \\
&\quad \times p(b')p(c|B=b)\mathrm{d}b'\mathrm{d}c\mathrm{d}a \\
&= \int p(y|a,b',C=c,E=e)p(a)p(b')p(c|B=b)\mathrm{d}b'\mathrm{d}c\mathrm{d}a \\
&\quad \text{by } (Y \perp\!\!\!\perp E|A,B,C) \text{ in } \mathcal{G}_{\underline{E}}
\end{aligned}
$$

$$
\begin{aligned}
p(y|\mathrm{do}(B=b),\mathrm{do}(D=d),\mathrm{do}(E=e)) &= p(y|\mathrm{do}(D=d),\mathrm{do}(E=e)) \\
&\quad \text{by } (Y \perp\!\!\!\perp B|D,E) \text{ in } \mathcal{G}_{\overline{DEB}}
\end{aligned}
$$

C.3 SCM for the synthetic experiment

The SCM for the synthetic example in Section 6.4.2 is given by:

$$
\begin{aligned}
A &= U_1 + \epsilon_A \\
B &= U_2 + \epsilon_B \\
C &= \exp(-B) + \epsilon_C \\
D &= \exp(-C)/10. + \epsilon_D \\
E &= \cos(A) + C/10 + \epsilon_E \\
Y &= \cos(D) + \sin(E) + U_1 + U_2\epsilon_y \\
U_1 &= \epsilon_{YA} \qquad U_2 = \epsilon_{YB}
\end{aligned}
$$

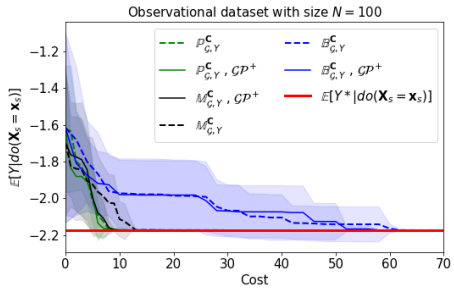


Figure C.1: Toy example. Convergence of CBO and standard BO across different initializations of $\mathcal{D}^I$. The red line gives the optimal Y^* when intervening on sets in $\mathbb{M}^{\mathbf{C}}_{\mathcal{G},Y}$, $\mathbb{P}^{\mathbf{C}}_{\mathcal{G},Y}$ or $\mathbb{B}^{\mathbf{C}}_{\mathcal{G},Y}$. Solid lines give CBO results when using the causal GP model which is denoted by $\mathcal{GP}^+$. Dotted line correspond to CBO with a standard GP prior model $p(f(\mathbf{x}_s)) = \mathcal{GP}(0, k_{\text{RBF}}(\mathbf{x}_s, \mathbf{x}'_s))$. Shaded areas are $\pm$ standard deviation.

with $\epsilon_i \sim \mathcal{N}(0, 1)$ for $i \in \{A, B, C, D, E, YA, YB\}$.

C.4 Cost configurations

Denote by $Co(\mathbf{X}, \mathbf{x})$ the cost of intervening on node $\mathbf{X}$ at the value $\mathbf{x}$. For the toy example (Section 6.4.1) and the real-data examples (Section 6.4.3 and Section 6.4.4) we consider fix unit cost across nodes. For the synthetic example (Section 6.4.2) we consider three possible cost configurations: equal fix costs across nodes, different fix costs across nodes and variable costs across nodes. These are set to:

1. Fix equal costs: $Co(B, b) = Co(D, d) = Co(E, e) = 1$.
2. Fix different costs: $Co(B, b) = 10$, $Co(D, d) = 5$ and $Co(E, e) = 20$.
3. Variable costs: $Co(B, b) = 10 + |b|$, $Co(D, d) = 5 + |d|$ and $Co(E, e) = 20 + |e|$.

C.5 Additional synthetic results

Fig. C.1 shows the results for the toy experiment in Section 6.4.1 across different initialization of $\mathcal{D}^I$. Fig. C.2 shows the results for the synthetic experiment in Section 6.4.2 across different cost structures and values of N.

C.6 Example in Healthcare

The DAG describing the causal relationships between statin drugs and PSA [Ferro et al., 2015; Thompson, 2019] is given in Fig. C.3(a) while the SCM for

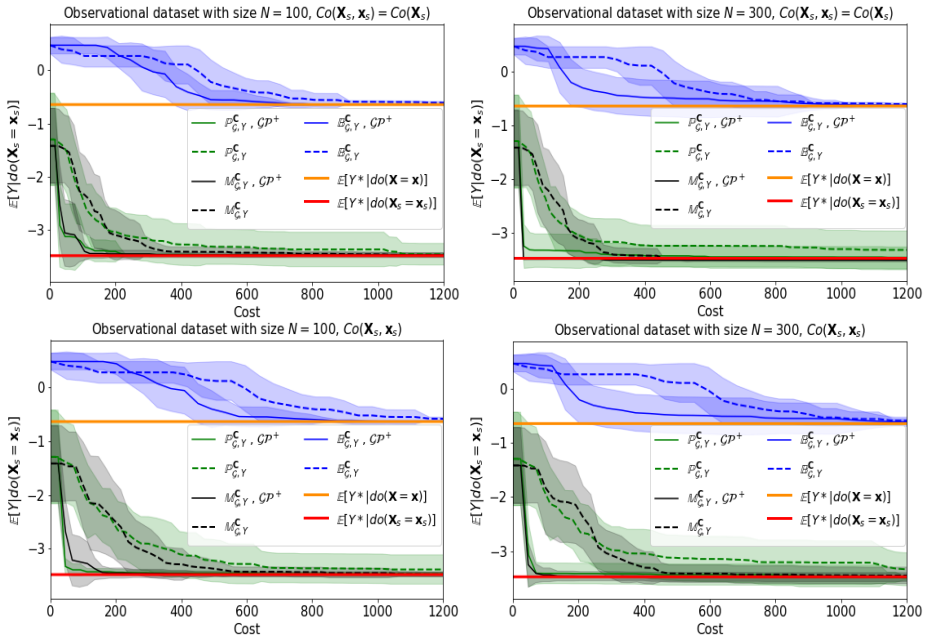


Figure C.2: Synthetic example. Convergence of CBO and standard BO. The orange line gives the optimal Y^* when intervening on $\mathbb{B}^{\mathbf{C}}_{\mathcal{G},Y}$. The red line gives the optimal Y^* when intervening on sets in $\mathbb{M}^{\mathbf{C}}_{\mathcal{G},Y}$ or $\mathbb{P}^{\mathbf{C}}_{\mathcal{G},Y}$. Solid lines give CBO results when using the causal GP model which is denoted by $\mathcal{GP}^+$. Dotted line correspond to CBO with a standard GP prior model. *Upper left*: option (2) in §C.4, $N = 100$. *lower left*: option (3) in §C.4, $N = 100$. *Upper right*: option (2) in §C.4, $N = 300$. *Lower right*: option (3) in §C.4, $N = 300$.

this example is:

$$
\begin{aligned}
\text{age} &= \mathcal{U}(55, 75) \\
\text{bmi} &= \mathcal{N}(27.0 - 0.01 \times \text{age}, 0.7) \\
\text{aspirin} &= \sigma(-8.0 + 0.10 \times \text{age} + 0.03 \times \text{bmi}) \\
\text{statin} &= \sigma(-13.0 + 0.10 \times \text{age} + 0.20 \times \text{bmi}) \\
\text{cancer} &= \sigma(2.2 - 0.05 \times \text{age} + 0.01 \times \text{bmi} - 0.04 \times \text{statin} + 0.02 \times \text{aspirin}) \\
Y &= \mathcal{N}(6.8 + 0.04 \times \text{age} - 0.15 \times \text{bmi} - 0.60 \times \text{statin} \\
&\qquad + 0.55 \times \text{aspirin} + 1.00 \times \text{cancer}, 0.4)
\end{aligned}
$$

where $\mathcal{U}(a, b)$ denotes a uniform random variable with parameters a and b, $\mathcal{N}(m, s)$ represents a normal random variable with mean m and standard deviation s and σ denotes the sigmoidal function computed as $\sigma(x) = \frac{1}{1+e^{-x}}$.

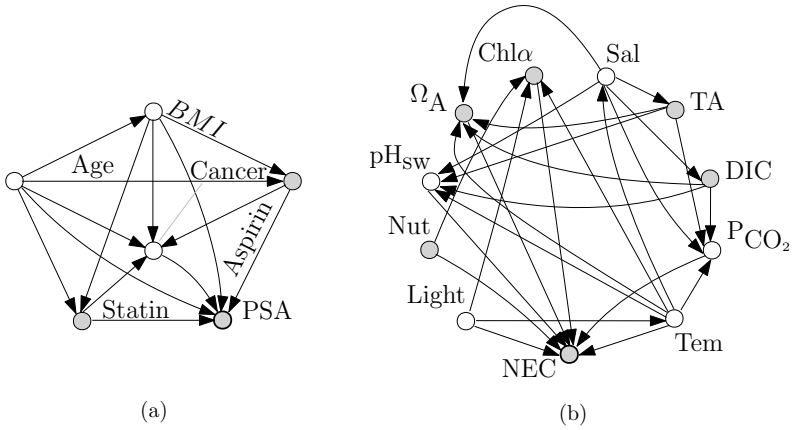


Figure C.3: *(a):* Causal graph of PSA level. Shaded nodes represent variables which can be intervened while empty nodes represent non-manipulative variables. The target variable is PSA. *(b):* DAG of NEC level. Shaded nodes represent manipulative variables. Empty nodes represent non-manipulative variables. The target variable is NEC.

C.7 Example in Ecology

The DAG describing the causal relationships between a set of environmental variables and NEC [Courtney et al., 2017] is given in Fig. C.3(b). The variables included in the DAG are the following:

- $Chl\alpha$: sea surface chlorophyll a;
- Sal: sea surface salinity;
- TA: seawater total alkalinity;
- DIC: seawater dissolved inorganic carbon;
- P_{CO_2}: seawater P_{CO_2};
- Tem: bottom temperature;
- NEC: net ecosystem calcification;
- Light: bottom light levels;
- Nut: PC1 of NH4, NiO2+NiO3, SiO4;
- pH_{SW}: seawater pH;
- Ω_A: seawater saturation with respect to aragonite.

See Andersson and Bates [2018] for more details about the included variables.

Appendix D

Supplementary Material for DAG-GP

D.1 Proofs of theorems and corollaries

In this section we give the proofs for the theoretical results in Chapter 7.

D.1.1 Proof of Theorem 7.1

Proof. Consider a generic $\mathbf{X}_s \in \mathcal{P}(\mathbf{X})$. $\mathbf{v}_s^I$ and $\mathbf{v}_s^N$ denote the values for the sets $\mathbf{I}_s^I$ and $\mathbf{I}_s^N$ respectively. $\mathbf{l} = (\mathbf{l}_s^I \cup \mathbf{l}_s^N)$ represents the values for the set $\mathbf{L}^N$, $\mathbf{l}_s^N$ is the value of $\mathbf{L}_s^N$ and $\mathbf{l}_s^I$ gives the value for $\mathbf{L}_s^I$. Notice that we can write the intervention on $\mathbf{X}_s$, that is $\text{do}\left(\mathbf{X}_s = \mathbf{x}\right)$, as $\text{do}\left(\mathbf{I}_s^I = \mathbf{v}_s^I\right) \cup \text{do}\left(\mathbf{X}_s \backslash \mathbf{I}_s^I = \mathbf{x} \backslash \mathbf{v}_s^I\right)$. Any function $t_s(\mathbf{x}) \in \mathbf{T}$ can be written as:

$$
\begin{aligned}
t_s(\mathbf{x}) &= \mathbb{E}[Y|\text{do}\left(\mathbf{X}_s = \mathbf{x}\right)] \\
&= \int \cdots \int \mathbb{E}\left[Y|\text{do}\left(\mathbf{I}_s^I = \mathbf{v}_s^I\right), \text{do}\left(\mathbf{X}_s \backslash \mathbf{I}_s^I = \mathbf{x} \backslash \mathbf{v}_s^I\right), \mathbf{I}_s^N = \mathbf{v}_s^N, \mathbf{L}_s^N = \mathbf{l}_s^N\right] \times \\
&\quad \times p(\mathbf{v}_s^N, \mathbf{l}_s^N|\text{do}\left(\mathbf{X}_s = \mathbf{x}\right)) \text{d}\mathbf{v}_s^N \text{d}\mathbf{l}_s^N \\
&= \int \cdots \int \mathbb{E}\left[Y|\text{do}\left(\mathbf{I}_s^I = \mathbf{v}_s^I\right), \text{do}\left(\mathbf{X}_s \backslash \mathbf{I}_s^I = \mathbf{x} \backslash \mathbf{v}_s^I\right), \text{do}\left(\mathbf{I}_s^N = \mathbf{v}_s^N\right), \mathbf{L}_s^N = \mathbf{l}_s^N\right] \times \\
&\quad \times p(\mathbf{v}_s^N, \mathbf{l}_s^N|\text{do}\left(\mathbf{X}_s = \mathbf{x}\right)) \text{d}\mathbf{v}_s^N \text{d}\mathbf{l}_s^N \qquad \text{by} \quad Y \perp\!\!\!\perp \mathbf{I}_s^N | \mathbf{X}_s, \mathbf{L}_s^N \ \text{ in } \ \mathcal{G}_{\overline{\mathbf{X}_s}\underline{\mathbf{I}_s^N}} \\
&= \int \cdots \int \mathbb{E}\left[Y|\text{do}\left(\mathbf{I}_s^I = \mathbf{v}_s^I\right), \text{do}\left(\mathbf{I}_s^N = \mathbf{v}_s^N\right), \mathbf{L}^N = \mathbf{l}_s^N\right] \times \\
&\quad \times p(\mathbf{v}_s^N, \mathbf{l}_s^N|\text{do}\left(\mathbf{X}_s = \mathbf{x}\right)) \text{d}\mathbf{v}_s^N \text{d}\mathbf{l}_s^N \\
&\qquad \text{by} \quad Y \perp\!\!\!\perp (\mathbf{X}_s \backslash \mathbf{I}_s^I) | \mathbf{I}, \mathbf{L}_s^N \ \text{ in } \ \mathcal{G}_{\overline{\mathbf{I}(\mathbf{X}_s \backslash \mathbf{I}_s^I)(\mathbf{L}_s^N)}}
\end{aligned}
\tag{D.1}
\tag{D.2}
$$

$$= \int \cdots \int \mathbb{E}\big[Y|\text{do}\left(\mathbf{I}=\mathbf{v}\right), \mathbf{L}_s^N = \mathbf{l}_s^N\big] p(\mathbf{v}_s^N, \mathbf{l}_s^N|\text{do}\left(\mathbf{X}_s=\mathbf{x}\right))\text{d}\mathbf{v}_s^N\text{d}\mathbf{l}_s^N$$

$$= \int \cdots \int \mathbb{E}\big[Y|\text{do}\left(\mathbf{I}=\mathbf{v}\right), \mathbf{L}_s^N = \mathbf{l}_s^N, \mathbf{L}_s^I = \mathbf{l}_s^I\big] \times$$
$$\times\, p(\mathbf{l}_s^I|\text{do}\left(\mathbf{I}=\mathbf{v}\right), \mathbf{L}_s^N = \mathbf{l}_s^N) p(\mathbf{v}_s^N, \mathbf{l}_s^N|\text{do}\left(\mathbf{X}_s=\mathbf{x}\right))\text{d}\mathbf{v}_s^N\text{d}\mathbf{l}_s^N\text{d}\mathbf{l}_s^I$$

$$= \int \cdots \int \mathbb{E}\big[Y|\text{do}\left(\mathbf{I}=\mathbf{v}\right), \mathbf{L}^N = \mathbf{l}\big] p(\mathbf{l}_s^I|\mathbf{L}_s^N = \mathbf{l}_s^N)$$
$$\times\, p(\mathbf{v}_s^N, \mathbf{l}_s^N|\text{do}\left(\mathbf{X}_s=\mathbf{x}\right))\text{d}\mathbf{v}_s^N\text{d}\mathbf{l}_s^N\text{d}\mathbf{l}_s^I \qquad \text{by} \quad \mathbf{L}_s^I \perp\!\!\!\perp \mathbf{I}|\mathbf{L}_s^N \ \text{ in } \ \mathcal{G}_{\overline{\mathbf{I}}} \quad \text{(D.3)}$$

$$= \int \cdots \int \mathbb{E}\big[Y|\text{do}\left(\mathbf{I}=\mathbf{v}\right), \mathbf{L}^N = \mathbf{l}\big] p(\mathbf{l}_s^I|\mathbf{l}_s^N) p(\mathbf{v}_s^N, \mathbf{l}_s^N|\text{do}\left(\mathbf{X}_s=\mathbf{x}\right))\text{d}\mathbf{v}_s^N\text{d}\mathbf{l}$$
$$= \int \cdots \int f(\mathbf{v}, \mathbf{l}) p(\mathbf{l}_s^I|\mathbf{l}_s^N) p(\mathbf{v}_s^N, \mathbf{l}_s^N|\text{do}\left(\mathbf{X}_s=\mathbf{x}\right))\text{d}\mathbf{v}_s^N\text{d}\mathbf{l} \qquad \text{(D.4)}$$

where Eq. (D.1) follows from Rule 2 of *do*-calculus while Eq. (D.2) and Eq. (D.3) follow from Rule 3 of *do*-calculus [Pearl, 2009b]. Eq. (D.4) gives the causal operator. □

D.1.2 Proof of Corollary 7.1

Proof. Suppose there exists another set $\mathbf{A}$, different from $\text{Pa}(Y)$ and defined as $\mathbf{A} = \text{Pa}(Y)\backslash\text{Pa}(Y)_i$, where $\text{Pa}(Y)_i$ represents a single variable in $\text{Pa}(Y)$, such that Eq. (7.2) holds for every set $\mathbf{X}_s$. This means that $\mathbf{A}$ blocks the front-door paths from all $\mathbf{X}_s \in \mathcal{P}(\mathbf{X})$ to Y. That is, $\mathbf{A}$ also blocks the directed path from $\text{Pa}(Y) \in \mathcal{P}(\mathbf{X})$ to Y thus including descendants of $\text{Pa}(Y)$ which are ancestors of Y. This contradicts the definition of a parent as a variable connected to Y through a direct arrow. The same reasoning hold for every set non containing all parents of Y thus $\text{Pa}(Y)$ is the smallest set such that Eq. (7.2) holds. □

D.1.3 Proof of Theorem 7.2

Proof. Suppose that $\mathbf{L}$ includes a node, say L_i, that has both an incoming and an outcoming unconfounded edge. The unconfounded incoming edge implies the existence of a set $\mathbf{X}_s$ for which L_i is a collider on the confounded path from $\mathbf{X}_s$ to Y. At the same time, the unconfounded outcoming edge implies the existence of a set $\mathbf{X}_{s'}$ such that L_i is an ancestor that we need to condition on in order to block the back-door paths from $\mathbf{X}_{s'}$ to Y. Consequently, the conditions $Y \perp\!\!\!\perp \mathbf{I}_s^N|\mathbf{X}_s, \mathbf{L}_s^N$ in $\mathcal{G}_{\overline{\mathbf{X}_s}\underline{\mathbf{I}_s^N}}$ and $Y \perp\!\!\!\perp (\mathbf{X}_s\backslash\mathbf{I}_s^I)|\mathbf{I}, \mathbf{L}_s^N$ in $\mathcal{G}_{\overline{\mathbf{I}(\mathbf{X}_s\backslash\mathbf{I}_s^I)(\mathbf{L}_s^N)}}$ in Theorem 7.1 cannot hold, at the same time, for both $\mathbf{X}_s$ and $\mathbf{X}_{s'}$. Indeed,

these independence conditions would be verified for X_s when excluding L_i from $\mathbf{L}^N$ while they would be verified for $X_{s'}$ when L_i is included in $\mathbf{L}^N$. The same reasoning hold for every node in $\mathbf{L}$ having both incoming and outcoming unconfounded edges. Therefore, if $\mathcal{G}$ has one of such node, it is not possible to find a set $\mathbf{L}$ such that Eq. (7.2) holds from all $\mathbf{X}_s \in \mathcal{P}(\mathbf{X})$. □

D.1.4 Proof of Corollary 7.2

Proof. Suppose there exists another set $\mathbf{A}$, different from $\mathbf{L}$ and defined as $\mathbf{A} = \mathbf{L} \backslash L_i$ where $L_i \in \mathcal{P}(\mathbf{X})$ denotes a single variable in $\mathbf{L}$ that is not a collider. The set $\mathbf{A}$ need to be such that $Y \perp\!\!\!\perp (\mathbf{X}_s \backslash \mathbf{I}_s^I) | \mathbf{I}, \mathbf{A}_s^N$ in $\mathcal{G}_{\overline{\mathbf{I}(\mathbf{X}_s \backslash \mathbf{I}_s^I)(\mathbf{A}_s^N)}}$ $\forall \mathbf{X}_s$ in $\mathcal{P}(\mathbf{X})$. Consider $\mathbf{X}_s = L_i$ and notice that the back door path from L_i to Y is *not* blocked by conditioning on $\mathbf{I}$ or $\mathbf{A}_s^N$. Therefore $Y \not\perp\!\!\!\perp (\mathbf{X}_s \backslash \mathbf{I}_s^I) | \mathbf{I}, \mathbf{A}_s^N$ in $\mathcal{G}_{\overline{\mathbf{I}(\mathbf{X}_s \backslash \mathbf{I}_s^I)(\mathbf{A}_s^N)}}$ and $\mathbf{A}$ is not a valid set. The same reasoning holds for every set not containing all confounders of Y thus $\mathbf{L}$ is the minimal set for $\mathbf{L}$. □

D.2 Partial transfer

The conditions in Theorem 7.1 allow for full transfer across *all* intervention functions in $\mathbf{T}$. As shown in Theorem 7.2, this might not be possible when a subset $\mathbf{L}' \subset \mathbf{L}$ includes nodes directly confounded with Y and with both unconfounded incoming and outcoming edges. However, we might still be interested in transferring information across a subset $\mathbf{T}' \subset \mathbf{T}$ which includes functions defined on $\mathcal{P}(\mathbf{X})' \subset \mathcal{P}(\mathbf{X})$. $\mathcal{P}(\mathbf{X})'$ is defined by excluding from $\mathcal{P}(\mathbf{X})$ those intervention sets including variables that have outcoming edges pointing into $\mathbf{L}'$ making the conditions in Theorem 7.1 satisfied for all sets in $\mathcal{P}(\mathbf{X})'$. For instance, consider Fig. 7.4(b) with the red edge where A is a confounded node that has both unconfounded incoming and outcoming edges. To block the path $E \leftarrow A \dashleftarrow\!\dashrightarrow Y$ we need to condition on A. However, conditioning on A opens the path $F \rightarrow A \dashleftarrow\!\dashrightarrow Y$ making it impossible to define a base function. We can thus focus on a subset $\mathbf{T}'$ in which all functions including $\mathbf{L}' = \{A\}$ as an intervention variable have been excluded. This is equivalent to doing full transfer in Fig. 7.4(b) with no incoming red edge in A.

D.3 Advantages of using the Causal operator

The causal operator allows us to write any $t_s(\mathbf{x})$ as an integral transformation of f. The integrating measure, which differs across $\mathbf{X}_s$, captures the dependency structure between the base set and the intervention set and can be reduced to *do*-free operations via *do*-calculus. Notice how, given our identifiability assumptions, all functions in $\mathbf{T}$ can also be computed by simply applying the

rules of *do*-calculus when observational data are available. However, writing the functions via $L_s(f)(\mathbf{x})$ has several advantages:

- it allows to identify the correlation structure across functions and exploit it to derive a multi-task probabilistic model. In turn, this enables the sharing of experimental information across causal effects;
- it allows to learn those intervention functions for which we cannot run experiments or for which only observational data is available via transfer of experimental information from correlated tasks;
- it allows to efficiently learn the set $\mathbf{T}$ when $\mathcal{P}(\mathbf{X})$ is large. Indeed, the sharing of interventional data reduces the total number of interventions one needs to implement in order to learn all causal effects.

All these aspects are crucial when we have limited observational data or we cannot run experiments on some intervention sets or the cardinality of $\mathcal{P}(\mathbf{X})$ is large. In the last case, specifying a model for each individual intervention function would not only be computationally expensive but might also lead to inconsistent prior specification across functions. Through the causal operator, we can model a system by only making one single assumption on f which is then propagated in the causal graph. When an intervention is performed, the information is propagated in the graph through the base function which links the different interventional functions. Finally, using f we avoid the specification of the correlation structure across every pair of intervention functions which would result in a combinatorial problem.

D.4 Single-task models for intervention functions

With single-task model we refer to the idea of placing an individual probabilistic model on the intervention function corresponding to each set in $\mathcal{P}(\mathbf{X})$. For each $\mathbf{X}_s \in \mathcal{P}(\mathbf{X})$ we have:

$$t_s(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), K(\mathbf{x}, \mathbf{x}'))$$

Depending on the availability of $\mathcal{D}^O$, one can decide to set the prior parameters to standard value, e.g. $m(\mathbf{x}) = 0$ and $K(\mathbf{x}, \mathbf{x}') = K_{\text{RBF}}(\mathbf{x}, \mathbf{x}')$ or adopt the causal prior construction introduced in Chapter 6. In both cases, the experimental information is not shared across functions and learning $\mathbf{T}$ requires intervening on all sets in $\mathcal{P}(\mathbf{X})$.

D.5 Active learning with DAG-GP

In Section 7.5.2 we have seen how, in order to select the next intervention level and intervention set within an AL algorithm while properly accounting for uncertainty reduction, one can use the DAG-GP$^+$ model for $\mathbf{T}$. In this case, for every $\mathbf{X}_s$ and at every step j, both variance terms used in the mutual information computations that is $\sigma^2_{\mathbf{x}_s^j|A^{j-1}}$ and $\sigma^2_{\mathbf{x}_s^j|D_s\setminus(A^{j-1}\cup\mathbf{x}_s^j)}$, which correspond to the variance terms of the kernel on $\mathbf{T}$, are determined by observational and interventional data across all experiments. Therefore, both GP$^+$ and DAG-GP$^+$ avoid collecting data points in areas where the causal GP prior is already providing information thus making the posterior mean equal to the true function. GP$^+$ is spreading the function evaluations on the remaining part of the input space collecting data points across the complete input space. On the contrary, DAG-GP$^+$ drives the data points to be collected where neither observational nor interventional information can be transferred for the remaining tasks thus focusing on the border of the input space. Using DAG-GP$^+$ as a surrogate for AL is thus crucial when designing optimal experiments as it allows to account for the uncertainty reduction obtained by transferring interventional data.

D.6 Bayesian Optimization with DAG-GP

The goal of BO is to optimize a function that is costly to evaluate and for which an explicit functional form is not available by making a series of function evaluations. In Chapter 6 we introduced the CBO algorithm which solves the problem of finding an optimal intervention in a DAG. CBO optimizes a target node by accounting for the causal relationship between the inputs and placing a single-task GP model on the intervention functions. By modelling these functions independently, CBO does not account for their correlation when exploring the intervention space. For each $\mathbf{X}_s \in \mathcal{P}(\mathbf{X})$ we have:

$$\begin{aligned} t_s(\mathbf{x}) &= \mathbb{E}[Y|\mathrm{do}\,(\mathbf{X}_s = \mathbf{x})] \\ t_s(\mathbf{x}) &\sim \mathcal{GP}(m^+(\mathbf{x}), K^+(\mathbf{x}, \mathbf{x}')) \end{aligned} \tag{D.5}$$

where $m^+(\mathbf{x})$ and $K^+(\mathbf{x}, \mathbf{x}')$ are the casual prior parameters. It is possible to improve CBO by considering DAG-GP$^+$ as the surrogate model. For each $\mathbf{X}_s \in \mathcal{P}(\mathbf{X})$, instead of considering a single-task GP model as in Eq. (D.5), one can use $t_s(\mathbf{x}) \sim \mathcal{GP}(m_s(\mathbf{x}), K_s(\mathbf{x}, \mathbf{x}'))$ with $m_s(\mathbf{x})$ and $K_s(\mathbf{x}, \mathbf{x}')$ being the parameters computed as in Eq. (7.4) and Eq. (7.5) in the main paper. This allows CBO to correctly place the next function evaluations thus significantly speeding up the convergence to the global optimum both with and without the causal prior.

D.7 Additional Experimental Results

Implementation details For all experiments, we assume Gaussian distributions for both the integrating measures and the conditional distributions in the DAGs. We optimize the parameters via maximum likelihood estimation. We generate the observational data by sampling from the SEMs given below. In order to generate interventional data, we sample from a modified version of the SCM where the functional relationship corresponding to the intervened variable is substituted with a constant value. This is equivalent to sampling from the mutilated graph. We compute the integrals in Eq. (7.4) and Eq. (7.5) via Monte-Carlo integration with 1000 samples. Finally, we fix the variance in the likelihood of Eq. (7.3) and fix the kernel hyper-parameters for both the RBF and causal kernel to standard values ($l = 1$, $\sigma_f^2 = 1$). Optimizing these parameters might potentially lead to improved performances and is left as an open problem.

Additional results In Table D.1 we report the fitting performances for both the synthetic examples and the health-care application across intervention functions and replicates when $N = 500$. In the following subsections we give

Table D.1: RMSE with $N = 500$

	DAG-GP$^+$	DAG-GP	GP$^+$	GP	*do*-calculus
DAG1	**0.48**	0.57	0.60	0.77	0.55
	(0.07)	(0.08)	(0.15)	(0.27)	-
DAG3	0.50	**0.42**	0.58	1.26	2.87
	(0.11)	(0.13)	(0.10)	(0.11)	-
DAG4	**0.09**	0.44	0.54	0.89	0.22
	(0.05)	(0.12)	(0.08)	(0.23)	-

additional *do*-calculus derivations and SCM details for all DAGs in Fig. 7.4.

D.7.1 DAG1

***Do*-calculus derivations** For DAG1 in Fig. 7.4(a) without the red edge we have $\mathbf{I} = \{Z\}$ and $\mathbf{L} = \varnothing$. The base function is thus given by $f = \mathbb{E}[Y|\mathrm{do}\,(Z = z)]$. In this section we give the expressions for the functions in $\mathbf{T}$ and show each of them can be written as a transformation of f with the

corresponding integrating measure. Notice that in this case $f \in \mathbf{T}$.

$$\begin{aligned}
\mathbb{E}[Y|\mathrm{do}\,(X=x)] &= \int \mathbb{E}[Y|\mathrm{do}\,(X=x)\,, z]p(z|\mathrm{do}\,(X=x))\mathrm{d}z \\
&= \int \mathbb{E}[Y|\mathrm{do}\,(X=x)\,, \mathrm{do}\,(Z=z)]p(z|\mathrm{do}\,(X=x))\mathrm{d}z \\
&\quad \text{by} \quad Y \perp\!\!\!\perp Z|X \text{ in } \mathcal{G}_{\overline{BXZ}} \\
&= \int \mathbb{E}[Y|\mathrm{do}\,(Z=z)]p(z|\mathrm{do}\,(X=x))\mathrm{d}z \\
&\quad \text{by} \quad Y \perp\!\!\!\perp X|Z \text{ in } \mathcal{G}_{\overline{XZ}} \\
&= \int f(z)p(z|\mathrm{do}\,(X=x))\mathrm{d}z
\end{aligned}$$

with $p(z|\mathrm{do}\,(X=x)) = p(z|X=x)$.

$$\mathbb{E}[Y|\mathrm{do}\,(Z=z)] = f(z).$$

$$\begin{aligned}
\mathbb{E}[Y|\mathrm{do}\,(X=x)\,, \mathrm{do}\,(Z=z)] &= \mathbb{E}[Y|\mathrm{do}\,(Z=z)] = f(z) \\
&\quad \text{by} \quad Y \perp\!\!\!\perp X|Z \text{ in } \mathcal{G}_{\overline{XZ}}
\end{aligned}$$

SCM:

$$\begin{aligned}
X &= \epsilon_X \\
Z &= \exp(-X) + \epsilon_Z \\
Y &= \cos(Z) - \exp(-Z/20) + \epsilon_Y
\end{aligned}$$

with $\epsilon_X \sim \mathcal{N}(0,1)$, $\epsilon_Z \sim \mathcal{N}(0,1)$ and $\epsilon_Y \sim \mathcal{N}(0,1)$. We consider the following interventional domains:

- $D(X) = [-5, 5]$
- $D(Z) = [-5, 20]$

D.7.2 DAG2

***Do*-calculus derivations** For DAG2 in Fig. 7.4(b) without the red edge we consider $\{A, C\}$ to be non-manipulative. We have $\mathbf{I} = \{D, E\}$ and $\mathbf{L} = \{A, B\}$. The base function is thus given by $f = \mathbb{E}[Y|\mathrm{do}\,(D=d)\,, \mathrm{do}\,(E=e)\,, a, b]$. In this section we give the expressions for all the functions in $\mathbf{T}$ and show each of them can be written as a transformation of f with the corresponding integrating measure.

Intervention sets of size 1

$$\begin{aligned}
\mathbb{E}[Y|\mathrm{do}\,(D=d)] &= \int \mathbb{E}[Y|\mathrm{do}\,(D=d)\,,e,a,b]p(a,b,e|\mathrm{do}\,(D=d))\mathrm{d}a\mathrm{d}b\mathrm{d}e \\
&= \int \mathbb{E}[Y|\mathrm{do}\,(D=d)\,,\mathrm{do}\,(E=e)\,,a,b]p(a,b,e|\mathrm{do}\,(D=d))\mathrm{d}a\mathrm{d}b\mathrm{d}e \\
&\quad \text{by} \quad Y \perp\!\!\!\perp E|D,A,B \text{ in } \mathcal{G}_{\overline{D}\underline{E}} \\
&= \int f(d,e,a,b)p(a,b,e|\mathrm{do}\,(D=d))\mathrm{d}a\mathrm{d}b\mathrm{d}e
\end{aligned}$$

with $p(a,b,e|\mathrm{do}\,(D=d)) = p(a)p(b)p(e|a,b)$.

$$\begin{aligned}
\mathbb{E}[Y|\mathrm{do}\,(E=e)] &= \int \mathbb{E}[Y|\mathrm{do}\,(E=e)\,,d,a,b]p(d,a,b|\mathrm{do}\,(E=e))\mathrm{d}a\mathrm{d}b\mathrm{d}s \\
&= \int \mathbb{E}[Y|\mathrm{do}\,(E=e)\,,\mathrm{do}\,(D=d)\,,a,b]p(d,a,b|\mathrm{do}\,(E=e))\mathrm{d}a\mathrm{d}b\mathrm{d}d \\
&\quad \text{by} \quad Y \perp\!\!\!\perp D|E,A,B \text{ in } \mathcal{G}_{\overline{E}\underline{D}} \\
&= \int f(d,e,a,b)p(d,a,b|\mathrm{do}\,(E=e))\mathrm{d}a\mathrm{d}b\mathrm{d}d
\end{aligned}$$

with $p(d,a,b|\mathrm{do}\,(E=e)) = p(a)p(b)p(d|b)$.

$$\begin{aligned}
\mathbb{E}[Y|\mathrm{do}\,(B=b)] &= \int \mathbb{E}[Y|\mathrm{do}\,(B=b)\,,d,e,a]p(d,e,a|\mathrm{do}\,(B=b))\mathrm{d}d\mathrm{d}e\mathrm{d}a \\
&= \int \mathbb{E}[Y|\mathrm{do}\,(B=b)\,,\mathrm{do}\,(D=d)\,,\mathrm{do}\,(E=e)\,,a]\times \\
&\qquad \times p(d,e,a|\mathrm{do}\,(B=b))\mathrm{d}d\mathrm{d}e\mathrm{d}a \quad \text{by} \quad Y \perp\!\!\!\perp D,E|B,A \text{ in } \mathcal{G}_{\overline{B}\underline{DE}} \\
&= \int \mathbb{E}[Y|\mathrm{do}\,(D=d)\,,\mathrm{do}\,(E=e)\,,a]p(d,e,a|\mathrm{do}\,(B=b))\mathrm{d}d\mathrm{d}e\mathrm{d}a \\
&\quad \text{by} \quad Y \perp\!\!\!\perp B|D,E,A \text{ in } \mathcal{G}_{\overline{BDE}} \\
&= \int \mathbb{E}\left[Y|\mathrm{do}\,(D=d)\,,\mathrm{do}\,(E=e)\,,a,b'\right]p(b')\times \\
&\qquad \times p(d,e,a|\mathrm{do}\,(B=b))\mathrm{d}d\mathrm{d}e\mathrm{d}a\mathrm{d}b' \\
&= \int f(d,e,a,b')p(b')p(d,e,a|\mathrm{do}\,(B=b))\mathrm{d}d\mathrm{d}e\mathrm{d}a\mathrm{d}b'
\end{aligned}$$

with $p(b')p(d,e,a|\mathrm{do}\,(B=b)) = p(b')p(a)p(d|e,a,B=b)p(e|a,B=b)$.

Intervention sets of size 2

$$\begin{aligned}
\mathbb{E}[Y|\mathrm{do}\,(D=d)\,,\mathrm{do}\,(E=e)] &= \int \mathbb{E}[Y|a,b,\mathrm{do}\,(D=d)\,,\mathrm{do}\,(E=e)]\times \\
&\qquad \times p(a,b|\mathrm{do}\,(D=d)\,,\mathrm{do}\,(E=e))\mathrm{d}a\mathrm{d}b \\
&= \int f(d,e,a,b)p(a,b|\mathrm{do}\,(D=d)\,,\mathrm{do}\,(E=e))\mathrm{d}a\mathrm{d}b
\end{aligned}$$

with $p(a, b|\text{do}\,(D = d)\,, \text{do}\,(E = e)) = p(a)p(b)$.

$$\begin{aligned}
\mathbb{E}[Y|\text{do}\,(B = b)\,, \text{do}\,(D = d)] &= \int \mathbb{E}[Y|\text{do}\,(B = b)\,, \text{do}\,(D = d)\,, a, e]\times \\
&\quad \times p(a, e|\text{do}\,(B = b)\,, \text{do}\,(D = d))\text{d}a\text{d}e \\
&= \int \mathbb{E}[Y|\text{do}\,(B = b)\,, \text{do}\,(D = d)\,, a, \text{do}\,(E = e)]\times \\
&\quad \times p(a, e|\text{do}\,(B = b)\,, \text{do}\,(D = d))\text{d}a\text{d}e \\
&\quad \text{by} \quad Y \perp\!\!\!\perp E|A, B, D \text{ in } \mathcal{G}_{\overline{BD}\underline{E}} \\
&= \int \mathbb{E}[Y|\text{do}\,(D = d)\,, \text{do}\,(E = e)\,, a]\times \\
&\quad \times p(a, e|\text{do}\,(B = b)\,, \text{do}\,(D = d))\text{d}a\text{d}e \\
&\quad \text{by} \quad Y \perp\!\!\!\perp B|A, D, E \text{ in } \mathcal{G}_{\overline{BDE}} \\
&= \int \mathbb{E}\big[Y|\text{do}\,(D = d)\,, \text{do}\,(E = e)\,, a, b'\big]p(b')\times \\
&\quad \times p(a, e|\text{do}\,(B = b)\,, \text{do}\,(D = d))\text{d}a\text{d}b'\text{d}e
\end{aligned}$$

with $p(b')p(a, e|\text{do}\,(B = b)\,, \text{do}\,(D = d)) = p(b')p(a)p(e|a, B = b)$.

$$\begin{aligned}
\mathbb{E}[Y|\text{do}\,(B = b)\,, \text{do}\,(E = e)] &= \int \mathbb{E}[Y|\text{do}\,(B = b)\,, \text{do}\,(E = e)\,, a, d]\times \\
&\quad \times p(a, d|\text{do}\,(B = b)\,, \text{do}\,(E = e))\text{d}a\text{d}d \\
&= \int \mathbb{E}[Y|\text{do}\,(B = b)\,, \text{do}\,(E = e)\,, a, \text{do}\,(D = d)]\times \\
&\quad \times p(a, d|\text{do}\,(B = b)\,, \text{do}\,(E = e))\text{d}a\text{d}d \\
&\quad \text{by} \quad Y \perp\!\!\!\perp D|A, B, E \text{ in } \mathcal{G}_{\overline{BE}\underline{D}} \\
&= \int \mathbb{E}[Y|\text{do}\,(D = d)\,, \text{do}\,(E = e)\,, a]\times \\
&\quad \times p(a, d|\text{do}\,(B = b)\,, \text{do}\,(E = e))\text{d}a\text{d}d \\
&\quad \text{by} \quad Y \perp\!\!\!\perp B|A, D, E \text{ in } \mathcal{G}_{\overline{BDE}} \\
&= \int \mathbb{E}\big[Y|\text{do}\,(D = d)\,, \text{do}\,(E = e)\,, a, b'\big]p(b')\times \\
&\quad \times p(a, d|\text{do}\,(B = b)\,, \text{do}\,(E = e))\text{d}a\text{d}b'\text{d}d \\
&= \int f(d, e, a, b')p(b')p(a, d|\text{do}\,(B = b)\,, \text{do}\,(E = e))\text{d}a\text{d}b'\text{d}d
\end{aligned}$$

with $p(b')p(a, d|\text{do}\,(B = b)\,, \text{do}\,(E = e)) = p(b')p(a)p(d|B = b)$.

Intervention sets of size 3

$$\mathbb{E}[Y|\text{do}\left(B=b\right),\text{do}\left(D=d\right),\text{do}\left(E=e\right)] = \mathbb{E}[Y|\text{do}\left(D=d\right),\text{do}\left(E=e\right)]$$
$$\text{by} \quad (Y \perp\!\!\!\perp B|D,E \text{ in } \mathcal{G}_{\overline{DEB}})$$

SCM:

$$\begin{aligned}
U_1 &= \epsilon_{YA} \\
U_2 &= \epsilon_{YB} \\
A &= U_1 + \epsilon_A \\
B &= U_2 + \epsilon_B \\
C &= \exp(-B) + \epsilon_C \\
D &= \exp(-C)/10. + \epsilon_D \\
E &= \cos(A) + C/10 + \epsilon_E \\
Y &= \cos(D) + \sin(E) + U_1 + U_2 + \epsilon_Y
\end{aligned}$$

with $\epsilon_i \sim \mathcal{N}(0,1), \quad \forall i \in \{YA, YB, A, B, C, D, E, Y\}$. We consider the following interventional domains:

- $D(B) = [-3, 4]$
- $D(D) = [-3, 3]$
- $D(E) = [-3, 3]$

D.7.3 DAG3

***Do*-calculus derivations** For DAG3 in Fig. 7.4(c) we consider the variables $\{\text{age}, \text{BMI}, \text{cancer}\}$ to be non-manipulative. We have no unobserved confounder thus $\mathbf{L} = \varnothing$ and $\mathbf{I} = \{\text{aspirin}, \text{statin}, \text{age}, \text{BMI}, \text{cancer}\}$. In this section we give the expressions for all the functions in $\mathbf{T}$ and show each of them can be written as a transformation of f with the corresponding integrating measure.

$$\begin{aligned}
\mathbb{E}[Y|\text{do}\left(\text{aspirin}=x\right)] = \int \cdots \int & f(\text{aspirin}, \text{statin}, \text{age}, \text{BMI}, \text{cancer}) \times \\
& \times p(\text{statin}, \text{age}, \text{BMI}, \text{cancer}|\text{do}\left(\text{aspirin}=x\right))\text{dstatin} \times \\
& \times \text{dagedBMIdcancer}
\end{aligned}$$

where the distribution $p(\text{statin}, \text{age}, \text{BMI}, \text{cancer}|\text{do}\left(\text{aspirin}=x\right))$ can be factorized as the product $p(\text{cancer}|\text{age}, \text{BMI}, \text{aspirin}, \text{aspirin}) \times p(\text{statin}|\text{age}, \text{BMI}) \times$

$p(\text{BMI}|\text{age}) \times p(\text{age})$.

$$\begin{aligned}\mathbb{E}[Y|\text{do}\,(\text{statin} = x)] = \int \cdots \int & f(\text{aspirin}, \text{statin}, \text{age}, \text{BMI}, \text{cancer}) \times \\ & \times p(\text{aspirin}, \text{age}, \text{BMI}, \text{cancer}|\text{do}\,(\text{statin} = x)) \times \\ & \times \text{daspirindagedBMIdcancer}\end{aligned}$$

where the distribution $p(\text{aspirin}, \text{age}, \text{BMI}, \text{cancer}|\text{do}\,(\text{statin} = x))$ can be factorized as the product $p(\text{cancer}|\text{age}, \text{BMI}, \text{aspirin}, \text{aspirin}) \times p(\text{aspirin}|\text{age}, \text{BMI}) \times p(\text{BMI}|\text{age}) \times p(\text{age})$. Finally we have:

$$\begin{aligned}\mathbb{E}[Y|\text{do}(\text{aspirin} = x, \text{statin} = z)] = \int \cdots \int & f(\text{aspirin}, \text{statin}, \text{age}, \text{BMI}, \text{cancer}) \times \\ & \times p(\text{age}, \text{BMI}, \text{cancer}|\text{do}\,(\text{aspirin} = x)\,, \\ & \qquad \text{do}\,(\text{statin} = z)) \times \\ & \times \text{dagedBMIdcancer}\end{aligned}$$

where the distribution $p(\text{age}, \text{BMI}, \text{cancer}|\text{do}\,(\text{aspirin} = x)\,, \text{do}\,(\text{statin} = z))$ can be factorized as the product $p(\text{cancer}|\text{age}, \text{BMI}, \text{aspirin}, \text{aspirin}) \times p(\text{BMI}|\text{age}) \times p(\text{age})$.

SCM:

$$\begin{aligned}\text{age} &= \mathcal{U}(55, 75) \\ \text{bmi} &= \mathcal{N}(27.0 - 0.01 \times \text{age}, 0.7) \\ \text{aspirin} &= \sigma(-8.0 + 0.10 \times \text{age} + 0.03 \times \text{bmi}) \\ \text{statin} &= \sigma(-13.0 + 0.10 \times \text{age} + 0.20 \times \text{bmi}) \\ \text{cancer} &= \sigma(2.2 - 0.05 \times \text{age} + 0.01 \times \text{bmi} - 0.04 \times \text{statin} + 0.02 \times \text{aspirin}) \\ Y &= \mathcal{N}(6.8 + 0.04 \times \text{age} - 0.15 \times \text{bmi} - 0.60 \times \text{statin} \\ &\quad + 0.55 \times \text{aspirin} + 1.00 \times \text{cancer}, 0.4)\end{aligned}$$

We consider the following interventional domains:

- $D(\text{aspirin}) = [0, 1]$
- $D(\text{statin}) = [0, 1]$

Appendix E

Supplementary Material for DCBO

E.1 Characterization of the time structure in a DAG with time dependent variables

In this section we give the proof for Theorem 8.1 in the main text. Consider the objective function $\mathbb{E}[Y_t|\text{do}\left(\mathbf{X}_{s,t} = \mathbf{x}_{s,t}\right), I_{0:t-1}]$ and define the following sets:

- $\text{Pa}\left(Y_t\right) = Y_t^{\text{PT}} \cup Y_t^{\text{PNT}}$ with $Y_t^{\text{PT}} = \text{Pa}\left(Y_t\right) \cap Y_{0:t-1}$ denoting the parents of Y_t that are target variables at previous time steps and $Y_t^{\text{PNT}} = \text{Pa}\left(Y_t\right) \setminus Y_t^{\text{PT}}$ including the parents of Y_t that are not target variables.

- For any set $\mathbf{X}_{s,t} \in \mathcal{P}(\mathbf{X}_t)$, $\mathbf{X}_{s,t}^{\text{PY}} = \mathbf{X}_{s,t} \cap \text{Pa}\left(Y_t\right)$ includes the variables in $\mathbf{X}_{s,t}$ that are parents of Y_t while $\mathbf{X}_{s,t}^{\text{NPY}} = \mathbf{X}_{s,t} \setminus \mathbf{X}_{s,t}^{\text{PY}}$ so that $\mathbf{X}_{s,t} = \mathbf{X}_{s,t}^{\text{PY}} \cup \mathbf{X}_{s,t}^{\text{NPY}}$.

- For any set $I_{0:t-1}^V \subseteq \mathbf{X}_{0:t-1}$, $I_{0:t-1}^{\text{PY}} = I_{0:t-1}^V \cap \text{Pa}\left(Y_t\right)$ includes the variables in $I_{0:t-1}^V$ that are parents of Y_t and $I_{0:t-1}^{\text{NPY}} = I_{0:t-1}^V \setminus I_{0:t-1}^{\text{PY}}$ so that $I_{0:t-1}^V = I_{0:t-1}^{\text{PY}} \cup I_{0:t-1}^{\text{NPY}}$.

- For any two sets $\mathbf{X}_{s,t} \in \text{Pa}\left(Y_t\right)$ and $I_{0:t-1}^V \subseteq \mathbf{X}_{0:t-1}$, $\mathbf{W}$ is a set such that $\text{Pa}\left(Y_t\right) = Y_t^{\text{PT}} \cup \mathbf{X}_{s,t}^{\text{PY}} \cup I_{0:t-1}^{\text{PY}} \cup \mathbf{W}$. This means that $\mathbf{W}$ includes those variables that are parents of Y_t but are nor target at previous time steps nor intervened variables.

In the following proof the values of $I_{0:t-1}^V$, $\mathbf{X}_{s,t}^{\text{PY}}$, $I_{0:t-1}^{\text{PY}}$ and W are denoted by $\mathbf{i}$, $\mathbf{x}^{\text{PY}}$, $\mathbf{i}^{\text{PY}}$ and $\mathbf{w}$ respectively. The values of Y_t^{PT}, $\mathbf{X}_{s,t}^{\text{NPY}}$ and $I_{0:t-1}^{\text{NPY}}$ are instead represented by $\mathbf{y}_t^{\text{PT}}$, $\mathbf{x}^{\text{NPY}}$ and $\mathbf{i}^{\text{NPY}}$. Finally, f_Y^Y and f_Y^{NY} are the functions in the SEM for Y_t (see Assumptions (1) in the main text).

Proof of Theorem 8.1 Under Assumptions (1) we can write the objective function $f_{s,t}(\mathbf{x}_{s,t}) = \mathbb{E}[Y_t|\text{do}\left(\mathbf{X}_{s,t} = \mathbf{x}_{s,t}\right), I_{0:t-1}]$ as:

$$
\begin{aligned}
f_{s,t}(\mathbf{x}_{s,t}) &= \int y_t p(y_t|\text{do}\left(\mathbf{X}_{s,t} = \mathbf{x}_{s,t}\right), I_{0:t-1})\mathrm{d}y_t \\
&= \int \cdots \int y_t p(y_t|\text{do}\left(\mathbf{X}^{\text{PY}}_{s,t} = \mathbf{x}^{\text{PY}}\right), \text{do}\left(\mathbf{X}^{\text{NPY}}_{s,t} = \mathbf{x}^{\text{NPY}}\right), \\
&\qquad I^{\text{PY}}_{0:t-1}, I^{\text{NPY}}_{0:t-1}, \mathbf{y}^{\text{PT}}_t, \mathbf{w})\times \\
&\qquad \times p(\mathbf{y}^{\text{PT}}_t, \mathbf{w}|\text{do}\left(\mathbf{X}_{s,t} = \mathbf{x}_{s,t}\right), I_{0:t-1})\mathrm{d}y_t\mathrm{d}\mathbf{y}^{\text{PT}}_t\mathrm{d}\mathbf{w} \\
&= \Big/\text{Rule 2 and Rule 1 of do-calculus}\Big/ \\
&= \int \cdots \int y_t p(y_t|\text{do}\left(\mathbf{X}^{\text{PY}}_{s,t} = \mathbf{x}^{\text{PY}}\right), I^{\text{PY}}_{0:t-1}, \mathbf{y}^{\text{PT}}_t, \mathbf{w}) \\
&\qquad \times p(\mathbf{y}^{\text{PT}}_t, \mathbf{w}|\text{do}\left(\mathbf{X}_{s,t} = \mathbf{x}_{s,t}\right), I_{0:t-1})\mathrm{d}y_t\mathrm{d}\mathbf{y}^{\text{PT}}_t\mathrm{d}\mathbf{w} \qquad \text{(E.1)} \\
&= \int \cdots \int \mathbb{E}\left[Y_t|\text{do}\left(\mathbf{X}^{\text{PY}}_{s,t} = \mathbf{x}^{\text{PY}}\right), I^{\text{PY}}_{0:t-1}, \mathbf{y}^{\text{PT}}_t, \mathbf{w}\right] \\
&\qquad \times p(\mathbf{y}^{\text{PT}}_t, \mathbf{w}|\text{do}\left(\mathbf{X}_{s,t} = \mathbf{x}_{s,t}\right), I_{0:t-1})\mathrm{d}\mathbf{y}^{\text{PT}}_t\mathrm{d}\mathbf{w} \\
&= \Big/\text{Assumption (2)}\Big/ \\
&= \int \cdots \int f^Y_Y(\mathbf{y}^{\text{PT}}_t) + f^{NY}_Y(\mathbf{x}^{\text{PY}}, \mathbf{i}^{\text{PY}}, \mathbf{w}) \\
&\qquad \times p(\mathbf{y}^{\text{PT}}_t, \mathbf{w}|\text{do}\left(\mathbf{X}_{s,t} = \mathbf{x}_{s,t}\right), I_{0:t-1})\mathrm{d}\mathbf{y}^{\text{PT}}_t\mathrm{d}\mathbf{w} \qquad \text{(E.2)} \\
&= \int \cdots \int f^Y_Y(\mathbf{y}^{\text{PT}}_t)p(\mathbf{y}^{\text{PT}}_t, \mathbf{w}|\text{do}\left(\mathbf{X}_{s,t} = \mathbf{x}_{s,t}\right), I_{0:t-1})\mathrm{d}\mathbf{y}^{\text{PT}}_t\mathrm{d}\mathbf{w} \\
&+ \int \cdots \int f^{NY}_Y(\mathbf{x}^{\text{PY}}, \mathbf{i}^{\text{PY}}, \mathbf{w})\times \\
&\qquad \times p(\mathbf{y}^{\text{PT}}_t, \mathbf{w}|\text{do}\left(\mathbf{X}_{s,t} = \mathbf{x}_{s,t}\right), I_{0:t-1})\mathrm{d}\mathbf{y}^{\text{PT}}_t\mathrm{d}\mathbf{w} \\
&= \int f^Y_Y(\mathbf{y}^{\text{PT}}_t)p(\mathbf{y}^{\text{PT}}_t|\text{do}\left(\mathbf{X}_{s,t} = \mathbf{x}_{s,t}\right), I_{0:t-1})\mathrm{d}\mathbf{y}^{\text{PT}}_t \\
&+ \int f^{NY}_Y(\mathbf{x}^{\text{PY}}, \mathbf{i}^{\text{PY}}, \mathbf{w})p(\mathbf{w}|\text{do}\left(\mathbf{X}_{s,t} = \mathbf{x}_{s,t}\right), I_{0:t-1})\mathrm{d}\mathbf{w} \\
&= \Big/\text{Time assumption}\Big/
\end{aligned}
$$

$$= \int f_Y^Y(\mathbf{y}_t^{\mathrm{PT}}) p(\mathbf{y}_t^{\mathrm{PT}} | I_{0:t-1}) \mathrm{d}\mathbf{y}_t^{\mathrm{PT}} \tag{E.3}$$

$$+ \int f_Y^{NY}(\mathbf{x}^{\mathrm{PY}}, \mathbf{i}^{\mathrm{PY}}, \mathbf{w}) p(\mathbf{w} | \mathrm{do}\left(\mathbf{X}_{s,t} = \mathbf{x}_{s,t}\right), I_{0:t-1}) \mathrm{d}\mathbf{w} \tag{E.4}$$

$$= \Big/ \text{Observed interventions} \Big/$$

$$= f_Y^Y(\mathbf{f}^\star) + \int f_Y^{NY}(\mathbf{x}^{\mathrm{PY}}, \mathbf{i}^{\mathrm{PY}}, \mathbf{w}) p(\mathbf{w} | \mathrm{do}\left(\mathbf{X}_{s,t} = \mathbf{x}_{s,t}\right), I_{0:t-1}) \mathrm{d}\mathbf{w} \tag{E.5}$$

$$= f_Y^Y(\mathbf{f}^\star) + \mathbb{E}_{p(\mathbf{w}|\mathrm{do}(\mathbf{X}_{s,t}=\mathbf{x}_{s,t}), I_{0:t-1})}\left[f_Y^{NY}(\mathbf{x}^{\mathrm{PY}}, \mathbf{i}^{\mathrm{PY}}, \mathbf{w})\right] \tag{E.6}$$

with $\mathbf{f}^\star = \{\mathbb{E}\Big[Y_i | \mathrm{do}\left(\mathbf{X}_{s,i}^\star = \mathbf{x}_{s,i}^\star\right), I_{0:i-1}\Big]\}_{Y_i \in Y_t^{\mathrm{PT}}}$ denoting the values of Y_t^{PT} corresponding to the optimal interventions implemented at previous time steps. Eq. (E.1) follows from Rule 2 of the *do*-calculus where $Y_t \perp\!\!\!\perp (\mathbf{X}_{s,t}^{\mathrm{NPY}} \cup I_{0:t-1}^{\mathrm{NPY}}) | \mathbf{X}_{s,t}^{\mathrm{PY}}, I_{0:t-1}^{\mathrm{PY}}, \mathbf{W}, Y_t^{\mathrm{PT}}$ in $\mathcal{G}_{\overline{\mathbf{X}_{s,t}^{\mathrm{PY}}, I_{0:t-1}^{\mathrm{PY}}} \underline{\mathbf{X}_{s,t}^{\mathrm{NPY}}, I_{0:t-1}^{\mathrm{NPY}}}}$ and Rule 1 of the *do*-calculus where $Y_t \perp\!\!\!\perp (\mathbf{X}_{s,t}^{\mathrm{NPY}} \cup I_{0:t-1}^{\mathrm{NPY}}) | \mathbf{X}_{s,t}^{\mathrm{PY}}, \overline{I_{0:t-1}^{\mathrm{PY}}, \mathbf{W}}, Y_t^{\mathrm{PT}}$ in $\mathcal{G}_{\overline{\mathbf{X}_{s,t}^{\mathrm{PY}}, I_{0:t-1}^{\mathrm{PY}}}}$. Eq. (E.2) follows from the second assumption in Assumptions (Assumptions 1) in the main text. Eq. (E.4) follows from $Y_t^{\mathrm{PT}} \perp\!\!\!\perp \mathbf{X}_{s,t}$ as interventions at time t cannot affect variables at time steps $0 : t-1$. Finally, noticing that $p(\mathbf{y}_t^{\mathrm{PT}} | I_{0:t-1})$ is the distribution targeted when optimizing the objective function at previous time steps, one can obtain the final expression in Eq. (E.6).

■

The derivations above show how the objective function at time t is given by the expected value of the output of the functional relationship f_Y^{NY} where the expectation is taken with respect to the variables that are not intervened on. This expectation is then shifted to account for the interventions implemented in the system at previous time steps that are affecting the target variable through f_Y^Y. Notice that, given our assumption on the absence of unobserved confounders, the distribution $p(\mathbf{w}|\mathrm{do}\left(\mathbf{X}_{s,t} = \mathbf{x}_{s,t}\right), I_{0:t-1})$ can be further simplified by conditioning on the variables in $\mathcal{G}$ that are on the back-door path between $(\mathbf{X}_{s,t}, I_{0:t-1})$ and Y_t and are not colliders. When the variable Y_t does not depend on the previous target nodes, the function f_Y^Y does not exist and Eq. (E.6) reduces to

$$\mathbb{E}_{p(\mathbf{w}|\mathrm{do}(\mathbf{X}_{s,t}=\mathbf{x}_{s,t}), I_{0:t-1})}\left[f_Y^{NY}(\mathbf{x}^{\mathrm{PY}}, \mathbf{i}^{\mathrm{PY}}, \mathbf{w})\right]. \tag{E.7}$$

In this case, previous interventions impact the target variable at time t by changing the distributions of the parents of Y_t that are not intervened but the information in $\mathbf{f}^\star$ is lost.

Eq. (E.6) can be further manipulated to reduce the second term to a *do*-free expression. Instead of applying the rules of do-calculus, one can expand

$p(\mathbf{w}|\text{do}\left(\mathbf{X}_{s,t}=\mathbf{x}_{s,t}\right), I_{0:t-1})$ by further conditioning on the parents of $\mathbf{W}$ that are not in $(\mathbf{X}_{s,t} \cup I_{0:t-1})$. In this case, $\mathbf{w}$ in $f_Y^{NY}(\mathbf{x}^{\text{PY}}, \mathbf{i}^{\text{PY}}, \mathbf{w})$ is replaced by the functions $\{f_W(\cdot)\}_{W\in\mathbf{W}}$ in the SEM corresponding to the variables in $\mathbf{W}$ and computed in $\mathbf{w}$. This leads to a partial composition of f_Y^{NY} with $\{f_W(\cdot)\}_{W\in\mathbf{W}}$ and can be repeated recursively until the set of variables with respect to which we are taking the expectation is a subset of $\mathbf{X}_{s,t}$ or $I_{0:t-1}^V$ thus making the distribution a delta function. For instance, when $\mathbf{W} \subset \mathbf{X}_{s,t}$ in Eq. (E.6), we have $p(\mathbf{w}|\text{do}\left(\mathbf{X}_{s,t}=\mathbf{x}_{s,t}\right), I_{0:t-1}) = \delta(\mathbf{w}=\mathbf{x}^{\text{W}}))$ where $\mathbf{x}^{\text{W}}$ are the values in $\mathbf{x}_{s,t}$ corresponding to the variables in $\mathbf{W}$. Therefore, Eq. (E.6) reduces to $f_Y^Y(\mathbf{f}^\star) + f_Y^{NY}(\mathbf{x}^{\text{PY}}, \mathbf{i}^{\text{PY}}, \mathbf{x}^{\text{W}})$.

For a generic $W \in \mathbf{W} \not\subseteq (\mathbf{X}_{s,t} \cup I_{0:t-1}^V)$, denote by $\mathbf{X}_{s,t}^{\text{PW}}$ and $I_{0:t-1}^{\text{PW}}$ the subset of variables in $\mathbf{X}_{s,t}$ and $I_{0:t-1}$ that are parents of W with corresponding values $\mathbf{x}^{\text{PW}}$ and $\mathbf{i}^{\text{PW}}$. Let $R = \text{Pa}\left(W\right) \setminus (\mathbf{X}_{s,t}^{\text{PW}} \cup I_{0:t-1}^{\text{PW}})$ and r be the corresponding value. We can define the $C(\cdot)$ function as:

$$C(W) = \begin{cases} f_W(\mathbf{u}_W, \mathbf{x}^{\text{PW}}, \mathbf{i}^{\text{PW}}) & \text{if} \quad R = \varnothing \\ f_W(\mathbf{u}_W, \mathbf{x}^{\text{PW}}, \mathbf{i}^{\text{PW}}, r) & \text{if} \quad R \subseteq \mathbf{X}_{s,t} \cup I_{0:t-1}^V \\ f_W(\mathbf{u}_W, \mathbf{x}^{\text{PW}}, \mathbf{i}^{\text{PW}}, C(R)) & \text{if} \quad R \not\subseteq \mathbf{X}_{s,t} \cup I_{0:t-1}^V \end{cases} \tag{E.8}$$

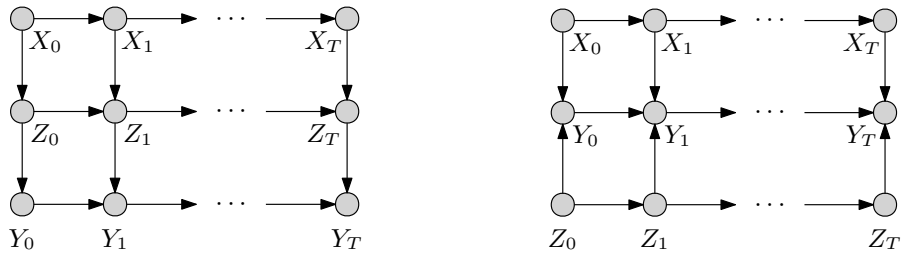
with u_W representing the exogenous variables with edges into W and f_W denoting the functional mapping for W in the SCM. Note that if $R = \varnothing$ and $\mathbf{X}_{s,t}^{\text{PW}}$ and $I_{0:t-1}^{\text{PW}}$ are also empty then $f_W(\mathbf{u}_W, \mathbf{x}^{\text{PW}}, \mathbf{i}^{\text{PW}})$ reduces to $f_W(\mathbf{u}_W)$. The same holds for the other cases that is $f_W(\mathbf{u}_W, \mathbf{x}^{\text{PW}}, \mathbf{i}^{\text{PW}}, r) = f_W(\mathbf{u}_W, r)$ and $f_W(\mathbf{u}_W, \mathbf{x}^{\text{PW}}, \mathbf{i}^{\text{PW}}, C(R)) = f_W(\mathbf{u}_W, C(R))$ when $\mathbf{X}_{s,t}^{\text{PW}}, I_{0:t-1}^{\text{PW}} = \varnothing$. Exploiting Eq. (E.8) we can rewrite Eq. (E.6) as:

$$\mathbb{E}[Y_t|\text{do}\left(\mathbf{X}_{s,t}=\mathbf{x}_{s,t}\right), I_{0:t-1}] = f_Y^Y(\mathbf{f}^\star) + \mathbb{E}_{p(\mathbf{U}_{0:t})}\left[f_Y^{NY}(\mathbf{x}^{\text{PY}}, \mathbf{i}^{\text{PY}}, \{C(W)\}_{W\in\mathbf{W}})\right] \tag{E.9}$$

The distribution $p(\mathbf{U}_{0:t})$ can be further simplified to consider only the exogenous variables with outgoing edges into the variables on the directed paths between $\mathbf{X}_{s,t}$ and Y_t^{PNT} and between $I_{0:t-1}^V$ and Y_t^{PNT}. Notice how the second term in Eq. (E.9) propagates the interventions, both at the present and past time steps, through the SCM so as to express the parents of the target variable as a function of the intervened values. The expected target is then obtained as the propagation of the intervened variables and intervened targets through the function f_{Y_t} in the SCM.

E.2 Example of derivations

Next we show how one can use Theorem 8.1 to derive some of the objective functions used by DCBO for the DAGs in Fig. E.1.



(a) DAG 1 (b) DAG 2

Figure E.1: Dynamic Bayesian networks with different topologies. (a) shows a DAG in which (per time-slice) the manipulative variable X flows through Z, whereas in (b) the manipulative variables are independent of each other (note the direction of the vertical edges).

Derivations for DAG 1 in Fig. E.1(a) Consider the DAG in Fig. E.1(a) and assume that the optimal intervention implemented at time $t = 0$ is given by $I_0 = \mathrm{do}\,(Z_0 = z_0^\star)$ and gives a target value of $y_0^\star$. At $t = 1$ the target variable is Y_1, $Y_t^{\mathrm{PT}} = \{Y_0\}$ and $Y_t^{\mathrm{PNT}} = \{Z_1\}$. Given I_0 we have $I_{0:t-1}^{\mathrm{PY}} = \varnothing$ and $I_{0:t-1}^{\mathrm{NPY}} = Z_0$. We can write the objective functions by noticing that, for $\mathbf{X}_{s,1} = \{Z_1\}$ we have $\mathbf{X}_{s,t}^{\mathrm{PY}} = \{Z_1\}$, $\mathbf{X}_{s,t}^{\mathrm{NPY}} = \varnothing$ and $W = \varnothing$, while for $\mathbf{X}_{s,1} = \{X_1\}$ we have $\mathbf{X}_{s,t}^{\mathrm{PY}} = \varnothing$, $\mathbf{X}_{s,t}^{\mathrm{NPY}} = \{X_1\}$ and $W = \{Z_1\}$. We do not compute the objective function for $\mathbf{X}_{s,1} = \{X_1, Z_1\}$ as this is equal to the function for $\mathbf{X}_{s,1} = \{Z_1\}$. Starting with $\mathbf{X}_{s,1} = \{Z_1\}$ we have:

$$
\begin{aligned}
\mathbb{E}[Y_1|\mathrm{do}\,(Z_1 = z), I_0] &= \int y_1 p(y_1|\mathrm{do}\,(Z_1 = z), I_0)\mathrm{d}y_1 \\
&= \int\int y_1 p(y_1|y_0, \mathrm{do}\,(Z_1 = z), I_0)\times \\
&\qquad \times p(y_0|\mathrm{do}\,(Z_1 = z), I_0)\mathrm{d}y_1\mathrm{d}y_0 \\
&= \int \mathbb{E}[Y_1|y_0, \mathrm{do}\,(Z_1 = z)]p(y_0|\mathrm{do}\,(Z_1 = z), I_0)\mathrm{d}y_0 \\
&= \int [f_Y^Y(y_0) + f_Y^{NY}(z)]p(y_0|I_0)\mathrm{d}y_0 \\
&= \int f_Y^Y(y_0)p(y_0|I_0)\mathrm{d}y_0 + f_Y^{NY}(z) \\
&= f_Y^Y(y_0^\star) + f_Y^{NY}(z)
\end{aligned}
$$

Notice that here $\mathbf{X}_{s,t}^{\mathrm{PY}} = \{Z_1\}$, $I_{0:t-1}^{\mathrm{PY}} = \varnothing$ and $W = \varnothing$. Therefore we have $\mathbb{E}_{p(\mathbf{w}|\mathrm{do}(\mathbf{X}_{s,t}=\mathbf{x}_{s,t}),I_{0:t-1})}\left[f_Y^{NY}(\mathbf{x}^{\mathrm{PY}}, \mathbf{i}^{\mathrm{PY}}, \mathbf{w})\right] = f_Y^{NY}(z)$. The objective function

for $\mathbf{X}_{s,1} = \{X_1\}$ can be written as:

$$\begin{aligned}
\mathbb{E}[Y_1|\mathrm{do}\left(X_1 = x\right), I_0] &= \int y_1 p(y_1|\mathrm{do}\left(X_1 = x\right), I_0)\mathrm{d}y_1 \\
&= \int\int\int y_1 p(y_1|y_0, z_1, \mathrm{do}\left(X_1 = x\right), I_0)\times \\
&\qquad \times p(y_0, z_1|\mathrm{do}\left(X_1 = x\right), I_0)\mathrm{d}y_1\mathrm{d}y_0\mathrm{d}z_1 \\
&= \int\int\int y_1 p(y_1|y_0, z_1) p(y_0, z_1|\mathrm{do}\left(X_1 = x\right), I_0)\mathrm{d}y_1\mathrm{d}y_0\mathrm{d}z_1 \\
&= \int\int \mathbb{E}[Y_1|y_0, z_1] p(y_0, z_1|\mathrm{do}\left(X_1 = x\right), I_0)\mathrm{d}y_0\mathrm{d}z_1 \\
&= \int\int [f_Y^Y(y_0) + f_Y^{NY}(z_1)] p(y_0, z_1|\mathrm{do}\left(X_1 = x\right), I_0)\mathrm{d}y_0\mathrm{d}z_1 \\
&= \int\int f_Y^Y(y_0) p(y_0, z_1|\mathrm{do}\left(X_1 = x\right), I_0)\mathrm{d}y_0\mathrm{d}z_1 \\
&\quad + \int\int f_Y^{NY}(z_1) p(y_0, z_1|\mathrm{do}\left(X_1 = x\right), I_0)\mathrm{d}y_0\mathrm{d}z_1 \\
&= \int f_Y^Y(y_0) p(y_0|I_0)\mathrm{d}y_0 + \int\int f_Y^{NY}(z_1) p(z_1|\mathrm{do}\left(X_1 = x\right), I_0)\mathrm{d}z_1 \\
&= f_Y^Y(y_0^\star) + \int f_Y^{NY}(z_1) p(z_1|\mathrm{do}\left(X_1 = x\right), I_0)\mathrm{d}z_1
\end{aligned} \tag{E.10}$$

In this case $\mathbf{X}_{s,t}^{\mathrm{PY}} = \varnothing$, $I_{0:t-1}^{\mathrm{PY}} = \varnothing$ and $\mathbf{W} = \{Z_1\}$ thus we have:

$$\mathbb{E}_{p(\mathbf{w}|\mathrm{do}(\mathbf{X}_{s,t}=\mathbf{x}_{s,t}), I_{0:t-1})}\left[f_Y^{NY}(\mathbf{x}^{\mathrm{PY}}, \mathbf{i}^{\mathrm{PY}}, \mathbf{w})\right] = \mathbb{E}_{p(z_1|\mathrm{do}(X_1=x), I_0)}\left[f_Y^{NY}(z_1)\right]. \tag{E.11}$$

We can further expand Eq. (E.10) noticing that in this case $\mathbf{W} = \{Z_1\} \not\subseteq \{X_1, Z_0\}$ but $\mathbf{X}_{s,t}^{PW} = \{X_1\}$, $I_{0:t-1}^{PW} = \{Z_0\}$ and $R = \varnothing$. Therefore we have $C(Z_1) = f_{Z_1}(\epsilon_{Z_1}, x_1, z_1)$ and Eq. (E.10) becomes:

$$\begin{aligned}
\mathbb{E}[Y|\mathrm{do}\left(X_1 = x\right), I_0] &= f_Y^Y(y_0^\star) + \int f_Y^{NY}(z_1) p(z_1|\mathrm{do}\left(X_1 = x\right), I_0)\mathrm{d}z_1 \\
&= f_Y^Y(y_0^\star) + \int\int f_Y^{NY}(z_1) p(z_1|\epsilon_{Z_1}, \mathrm{do}\left(X_1 = x\right), I_0)\times \\
&\quad \times p(\epsilon_{Z_1}|\mathrm{do}\left(X_1 = x\right), I_0)\mathrm{d}z_1\mathrm{d}\epsilon_{Z_1} \\
&= f_Y^Y(y_0^\star) + \int\int f_Y^{NY}(z_1)\delta(z_1 = f_{Z_1}(\epsilon_{Z_1}, x, z_0^\star)) p(\epsilon_{Z_1})\mathrm{d}z_1\mathrm{d}\epsilon_{Z_1} \\
&= f_Y^Y(y_0^\star) + \mathbb{E}_{p(\epsilon_{Z_1})}\left[f_Y^{NY}(f_{Z_1}(\epsilon_{Z_1}, x, z_0^\star))\right].
\end{aligned}$$

Derivations for DAG 2 in Fig. E.1(b) Next we consider the DAG in Fig. E.1b and assume that the optimal interventions implemented at time $t = 0$ and $t = 1$ are given by $I_0 = \mathrm{do}\left(X_0 = x_0^\star\right)$ and $I_1 = \mathrm{do}\left(Z_1 = z_1^\star\right)$. The optimal target values associated with these two interventions are given by $y_0^\star$ and $y_1^\star$ respectively. We are interested in computing two objective

functions: $\mathbb{E}[Y_2|\text{do}\left(X_2 = x_2\right), I_0, I_1]$ and $\mathbb{E}[Y_2|\text{do}\left(Z_2 = z_2\right), I_0, I_1]$. In this case $\mathbf{y}_t^{\text{PT}} = \{Y_1\}$, $Y_t^{\text{PNT}} = \{X_2, Z_2\}$, $I_{0:t-1}^{\text{PY}} = \varnothing$ and $I_{0:t-1}^{\text{NPY}} = \{X_0, Z_1\}$. Starting from $\mathbb{E}[Y_2|\text{do}\left(X_2 = x_2\right), I_0, I_1]$, when $\mathbf{X}_{s,2} = \{X_2\}$ we have $\mathbf{X}_{s,t}^{\text{PY}} = \{X_2\}$, $\mathbf{X}_{s,t}^{\text{NPY}} = \varnothing$ and $W = \{Z_2\}$. We can write:

$$
\begin{aligned}
\mathbb{E}[Y_2|\text{do}\left(X_2 = x_2\right), I_0, I_1] &= \int y_2 p(y_2|\text{do}\left(X_2 = x_2\right), I_0, I_1)\text{d}y_2 \\
&= \int\int\int y_2 p(y_2|y_1, z_2, \text{do}\left(X_2 = x_2\right), I_0, I_1)\times \\
&\quad \times p(y_1, z_2|\text{do}\left(X_2 = x_2\right), I_0, I_1)\text{d}y_2\text{d}y_1\text{d}z_2 \\
&= \int\int\int y_2 p(y_2|y_1, z_2, \text{do}\left(X_2 = x_2\right))\times \\
&\quad \times p(y_1, z_2|\text{do}\left(X_2 = x_2\right), I_0, I_1)\text{d}y_2\text{d}y_1\text{d}z_2 \\
&= \int\int \mathbb{E}[Y_2|y_1, z_2, \text{do}\left(X_2 = x_2\right)]\times \\
&\quad \times p(y_1, z_2|\text{do}\left(X_2 = x_2\right), I_0, I_1)\text{d}y_1\text{d}z_2 \\
&= \int\int [f_Y^Y(y_1) + f_Y^{NY}(x_2, z_2)]p(y_1, z_2|\text{do}\left(X_2 = x_2\right), I_0, I_1)\text{d}y_1\text{d}z_2 \\
&= \int\int f_Y^Y(y_1)p(y_1, z_2|\text{do}\left(X_2 = x_2\right), I_0, I_1)\text{d}y_1\text{d}z_2 \\
&+ \int\int f_Y^{NY}(x_2, z_2)p(y_1, z_2|\text{do}\left(X_2 = x_2\right), I_0, I_1)\text{d}y_1\text{d}z_2 \\
&= \int f_Y^Y(y_1)p(y_1|I_0, I_1)\text{d}y_1 + \int f_Y^{NY}(x_2, z_2)p(z_2|\text{do}\left(X_2 = x_2\right), I_0, I_1)\text{d}z_2 \\
&= f_Y^Y(y_1^\star) + \int f_Y^{NY}(x_2, z_2)p(z_2|I_1)\text{d}z_2 \\
&= f_Y^Y(y_1^\star) + \mathbb{E}_{p(\epsilon_{Z_2})}\left[f_Y^{NY}(x_2, f_{Z_2}(z_1^\star, \epsilon_{Z_2}))\right]
\end{aligned}
$$

Next we compute $\mathbb{E}[Y_2|\text{do}\left(Z_2 = z_2\right), I_0, I_1]$ by noticing that, when $\mathbf{X}_{s,2} = \{Z_2\}$, we have $\mathbf{X}_{s,t}^{\text{PY}} = \{Z_2\}$, $\mathbf{X}_{s,t}^{\text{NPY}} = \varnothing$ and $W = \{X_2\}$. In this case we have:

$$
\begin{aligned}
\mathbb{E}[Y_2|\text{do}\left(Z_2 = z_2\right), I_0, I_1] &= \int y_2 p(y_2|\text{do}\left(Z_2 = z_2\right), I_0, I_1)\text{d}y_2 \\
&= \int\int\int y_2 p(y_2|y_1, x_2, \text{do}\left(Z_2 = z_2\right), I_0, I_1)\times \\
&\quad \times p(y_1, x_2|\text{do}\left(Z_2 = z_2\right), I_0, I_1)\text{d}y_2\text{d}y_1\text{d}x_2 \\
&= \int\int\int y_2 p(y_2|y_1, x_2, \text{do}\left(Z_2 = z_2\right))\times \\
&\quad \times p(y_1, x_2|\text{do}\left(Z_2 = z_2\right), I_0, I_1)\text{d}y_2\text{d}y_1\text{d}x_2 \\
&= \int\int \mathbb{E}[Y_2|y_1, x_2, \text{do}\left(Z_2 = z_2\right)]\times \\
&\quad \times p(y_1, x_2|\text{do}\left(Z_2 = z_2\right), I_0, I_1)\text{d}y_1\text{d}x_2
\end{aligned}
$$

$$
\begin{aligned}
&= \int\int [f_Y^Y(y_1) + f_Y^{NY}(x_2, z_2)] \times \\
&\quad \times p(y_1, x_2 | \mathrm{do}\left(Z_2 = z_2\right), I_0, I_1) \mathrm{d}y_1 \mathrm{d}x_2 \\
&= \int f_Y^Y(y_1) p(y_1 | I_0, I_1) \mathrm{d}y_1 \\
&\quad + \int f_Y^{NY}(x_2, z_2) p(x_2 | \mathrm{do}\left(Z_2 = z_2\right), I_0, I_1) \mathrm{d}x_2 \\
&= f_Y^Y(y_1^\star) + \int f_Y^{NY}(x_2, z_2) p(x_2 | \mathrm{do}\left(Z_2 = z_2\right), I_0, I_1) \mathrm{d}x_2 \qquad \text{(E.12)}
\end{aligned}
$$

Let's now focus on Eq. (E.12). Here $\mathbf{W} = \{X_2\} \not\subseteq \{Z_2, X_0, Z_1\}$, $\mathbf{X}_{s,t}^{\mathrm{PW}} = \varnothing$, $I_{0:t-1}^{\mathrm{PW}} = \varnothing$ and $R = \{X_1\}$. Therefore we have $C(X_2) = f_{X_2}(\epsilon_{X_2}, C(R))$ as $R \not\subseteq \{Z_2, X_0, Z_1\}$. We thus need to compute $C(R) = C(X_1)$. When $W = X_1$, $\mathbf{X}_{s,t}^{PW} = \varnothing$ but $I_{0:t-1}^{PW} = \{X_0\}$ and $R = \varnothing$. We can thus write $C(X_2) = f_{X_2}(\epsilon_{X_2}, f_{X_1}(\epsilon_{X_1}, x_0))$ and replace it in Eq. (E.12) to get:

$$
\mathbb{E}[Y_2 | \mathrm{do}\left(Z_2 = z_2\right), I_0, I_1] = f_Y^Y(y_1^\star) + \mathbb{E}_{p(\epsilon_{X_2})p(\epsilon_{X_1})}\big[f_Y^{NY}(f_{X_2}(\epsilon_{X_2}, f_{X_1}(\epsilon_{X_1}, x_0)), z_2)\big].
$$

E.3 Reducing the search space

In this section we give the proof for Proposition 8.3.1 in the main text. Denote by $\mathbb{M}_t \subseteq \mathcal{P}(\mathbf{X}_t)$ the set of MISs at time t and let $\mathbb{S}_t = \mathcal{P}(\mathbf{X}_t) \backslash \mathbb{M}_t$ include the sets that are not MIS. For any set $\mathbf{X}_{s,t} \in \mathbb{S}_t$, we denote the *superfluous* variables by $\mathbf{S}_{s,t}$. These are the variables not needed in the computation of the objective functions. In other words, these are those variables for which $\mathbb{E}[Y_t | \mathrm{do}\left(\mathbf{X}_{s,t} = \mathbf{x}_{s,t}\right), I_{0:t-1}] = \mathbb{E}\big[Y_t | \mathrm{do}\left(\mathbf{X}'_{s,t} = \mathbf{x}'_{s,t}\right), I_{0:t-1}\big]$ where $\mathbf{X}'_{s,t} = \mathbf{X}_t \backslash \mathbf{S}_{s,t}$. Given the initial set of MISs at time $t = 0$ represented by $\mathbb{M}_0$ we have:

Proof of Proposition 8.3.1 Consider a generic set $\mathbf{X}_{s,t} \in \mathbb{S}_t$. The corresponding objective function $f_{s,t}(\mathbf{x}_{s,t}) = \mathbb{E}[Y_t | \mathrm{do}\left(\mathbf{X}_{s,t} = \mathbf{x}_{s,t}\right), I_{0:t-1}]$ can be

written as:

$$
\begin{aligned}
f_{s,t}(\mathbf{x}_{s,t}) &= \mathbb{E}\left[Y_t|\text{do}\left(\mathbf{X}'_{s,t}=\mathbf{x}'_{s,t}\right),\text{do}\left(\mathbf{S}_{s,t}=\mathbf{s}_{s,t}\right),I_{0:t-1}\right] \\
&= \int \mathbb{E}\left[Y_t|\text{do}\left(\mathbf{X}'_{s,t}=\mathbf{x}'_{s,t}\right),\text{do}\left(\mathbf{S}_{s,t}=\mathbf{s}_{s,t}\right),I_{0:t-1},\mathbf{V}_{0:t-1}\backslash I_{0:t-1}\right] \\
&\quad \times p(\mathbf{V}_{0:t-1}\backslash I_{0:t-1}|\text{do}\left(\mathbf{X}'_{s,t}=\mathbf{x}'_{s,t}\right),\text{do}\left(\mathbf{S}_{s,t}=\mathbf{s}_{s,t}\right),I_{0:t-1})\text{d}\mathbf{V}_{0:t-1} \\
&= \int \mathbb{E}\left[Y_t|\text{do}\left(\mathbf{X}'_{s,t}=\mathbf{x}'_{s,t}\right),I_{0:t-1},\mathbf{V}_{0:t-1}\backslash I_{0:t-1}\right] \qquad \text{(E.13)} \\
&\quad \times p(\mathbf{V}_{0:t-1}\backslash I_{0:t-1}|\text{do}\left(\mathbf{X}'_{s,t}=\mathbf{x}'_{s,t}\right),I_{0:t-1})\text{d}\mathbf{V}_{0:t-1} \\
&= \mathbb{E}\left[Y_t|\text{do}\left(\mathbf{X}'_{s,t}=\mathbf{x}'_{s,t}\right),I_{0:t-1}\right] \qquad \text{(E.14)}
\end{aligned}
$$

where Eq. (E.13) can be obtained by Rule 3 of the *do*-calculus noticing that $Y_t \perp\!\!\!\perp \mathbf{S}_{s,t}|\mathbf{X}'_{s,t}, I_{0:t-1}, \mathbf{V}_{0:t-1}\backslash I_{0:t-1}$ in $\mathcal{G}_{\overline{\mathbf{S}_{s,t},I_{0:t-1},\mathbf{X}'_{s,t}}}$. This is due to the fact that $\mathbf{S}_{s,t}$ does not have back door paths to Y_t in $\mathcal{G}_{\overline{\mathbf{S}_{s,t},I_{0:t-1},\mathbf{X}'_{s,t}}}$ and its front door paths to Y_t in $\mathcal{G}_{\overline{\mathbf{S}_{s,t},I_{0:t-1},\mathbf{X}'_{s,t}}}$ are blocked by $\mathbf{X}'_{s,t}$. Indeed, $\mathbf{S}_{s,t}$ cannot have outgoing edges to variables in $0:t-1$ and the front door paths to Y_t going through variables at time t are blocked by definition of a MIS set by $\mathbf{X}'_{s,t}$ in $\mathcal{G}_t = \mathcal{G}, \forall t$.

■

E.4 Additional experimental details and results

In this section, we give additional experimental details associated with the experiments discussed in Section 8.4 of the main text.

E.4.1 Stationary DAG and SCM (STAT.)

The SCM used for the experiment denoted by STAT. is given by:

$$
\begin{aligned}
X_t &= X_{t-1}\mathbb{1}_{t>0} + \epsilon_X \\
Z_t &= \exp(-X_t) + Z_{t-1}\mathbb{1}_{t>0} + \epsilon_Z \\
Y_t &= \cos(Z_t) - \exp(-Z_t/20) + Y_{t-1}\mathbb{1}_{t>0} + \epsilon_Y
\end{aligned}
$$

where $\epsilon_i \sim \mathcal{N}(0,1)$ for $i \in \{X, Z, Y\}$ and $\mathbb{1}_{t>0}$ represent an indicator function that is equal to one $t > 0$ and zero otherwise. We run this experiment 10 times by setting $T = 3$, $N = 10$, $D(X_t) = \{-5.0, 5.0\}$ and $D(Z_t) = \{-5.0, 20.0\}$. Notice that, given the DAG in Fig. E.2 (left panel), we have $\mathbb{M}_t = \{\{X_t\}, \{Z_t\}\}$. The right panel of Fig. E.2 shows the true objective functions together with the optimal intervention per time step (1^{st} row), the dynamic causal GP model for the intervention on Z (2^{nd} row) and the convergence of the DCBO algorithm to the optimum (3^{rd} row). Notice how the location of the optimum changes significantly both in terms of optimal set and intervention value when going from

$t = 0$ to $t = 1$. DCBO quickly identifies the optimum via the prior dependency on $y^{\star}_{0:t-1}$.

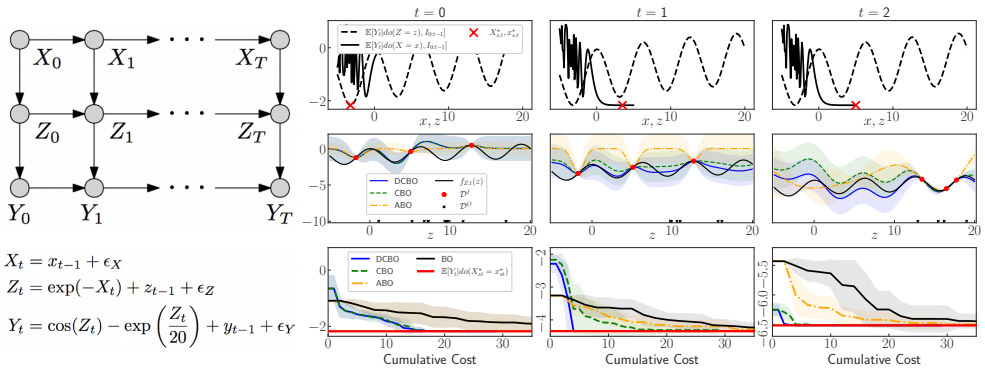


Figure E.2: Stationary synthetic experiment (STAT.). *Left panel*: $\mathcal{G}_{0:T}$ and SEM. *Right panel, 1st row*: Objective functions for the sets in $\mathbb{M} = \{\{Z\}, \{X\}\}$. *Right panel, 2nd row*: Posterior GP obtained when using the dynamic causal GP construction vs alternative models. *Right panel, 3rd row*: Convergence of DCBO and alternative models to the true optimum (red line) across 10 replicates. Shaded areas give ± one standard deviation.

E.4.2 Noisy manipulative variables (NOISY)

The SCM used for the experiment denoted by NOISY is given by:

$$
\begin{aligned}
X_t &= X_{t-1}\mathbb{1}_{t>0} + \epsilon_X \\
Z_t &= \exp(-X_t) + Z_{t-1}\mathbb{1}_{t>0} + \epsilon_Z \\
Y_t &= \cos(Z_t) - \exp(-Z_t/20) + Y_{t-1}\mathbb{1}_{t>0} + \epsilon_Y
\end{aligned}
$$

where, differently from before, we have $\epsilon_Y \sim \mathcal{N}(0, 1)$ and $\epsilon_i \sim \mathcal{N}(2, 4)$ for $i \in \{X, Z\}$. We keep the remaining parameters equal to the previous experiment. This means $T = 3$, $N = 10$, $D(X_t) = \{-5.0, 5.0\}$ and $D(Z_t) = \{-5.0, 20.0\}$.

E.4.3 Missing observational data (MISS.)

For this experiment we use the same SCM of the experiment denoted by STAT. However, we set $T = 6$, $N = 10$ for the first three time steps, and $N = 0$ afterwards. Fig. E.3 shows the convergence paths for this experiment. In this setting DCBO consistently outperforms CBO at every time step. However, notice how the ABO performance improves over time. This is due to the ability of ABO to learn the time dynamic of the objective function and exploit all interventional data collected over time to predict at the next time step.

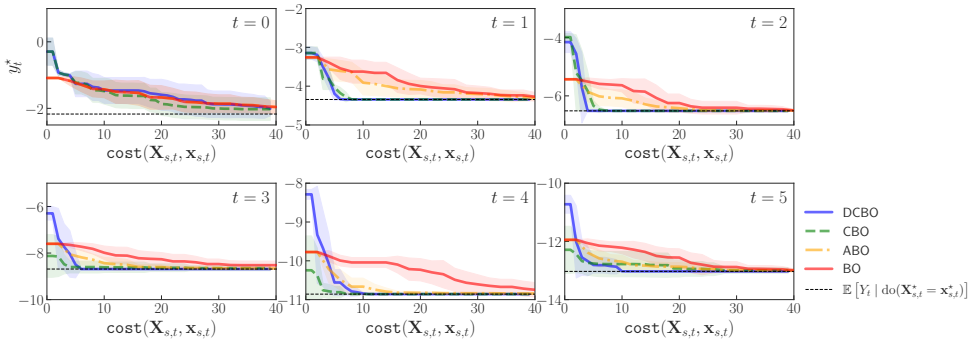


Figure E.3: Experiment MISS. Convergence of DCBO and competing methods across replicates. The red line gives the optimal $y_t^*, \forall t$. Shaded areas are $\pm$ standard deviation.

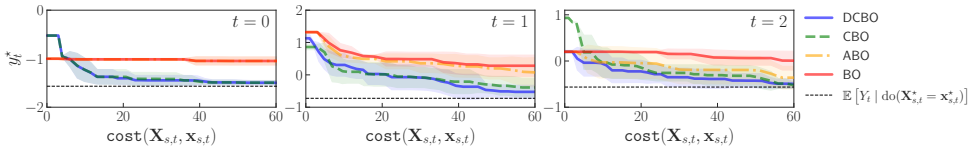


Figure E.4: Experiment MULTIV. Convergence of DCBO and competing methods across replicates. The red line gives the optimal $y_t^*, \forall t$. Shaded areas are $\pm$ standard deviation.

E.4.4 Multivariate intervention sets (MULTIV.)

The SCM used for the experiment denoted by MULTIV. is given by:

$$
\begin{aligned}
W_t &= \epsilon_W \\
X_t &= -X_{t-1}\mathbb{1}_{t>0} + \epsilon_X \\
Z_t &= \sin(W_t) - Z_{t-1}\mathbb{1}_{t>0} + \epsilon_Z \\
Y_t &= -2 * \exp(-(X_t - 1)^2) - \exp(-(X_t + 1)^2) - (Z_t - 1)^2 \\
&\quad - Z_T^2 + \cos(Z_t * Y_{t-1}) - Y_{t-1}\mathbb{1}_{t>0} + \epsilon_Y
\end{aligned}
$$

where $\epsilon_i \sim \mathcal{N}(0, 1)$ for $i \in \{X, Z, W, Y\}$. We set $T = 3$, $N = 500$, $D(X_t) = \{-5.0, 5.0\}$, $D(Z_t) = \{-5.0, 20.0\}$ and $D(W_t) = \{-3.0, 3.0\}$. Notice that here DCBO and CBO explore the set $\mathbb{M}_t = \{\{X_t\}, \{Z_t\}, \{X_t, Z_t\}\}$ while BO and ABO intervene on $\{X_t, Z_t, W_t\}$. Fig. E.4 shows the convergence paths for this experiment.

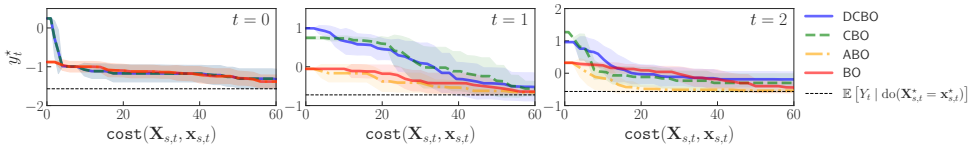


Figure E.5: Experiment IND. Convergence of DCBO and competing methods across replicates. The red line gives the optimal $y_t^*, \forall t$. Shaded areas are $\pm$ standard deviation.

E.4.5 Independent manipulative variables (IND.)

The SCM used for the experiment denoted by IND. is given by:

$$
\begin{aligned}
X_t &= -X_{t-1}\mathbb{1}_{t>0} + \epsilon_X \\
Z_t &= -Z_{t-1}\mathbb{1}_{t>0} + \epsilon_Z \\
Y_t &= -2 * \exp(-(X_t - 1)^2) - \exp(-(X_t + 1)^2) - (Z_t - 1)^2 \\
&\quad - Z_T^2 + \cos(Z_t * Y_{t-1}) - Y_{t-1}\mathbb{1}_{t>0} + \epsilon_Y
\end{aligned}
$$

where $\epsilon_i \sim \mathcal{N}(0, 1)$ for $i \in \{X, Z, Y\}$. We set $T = 3$, $N = 10$, $D(X_t) = \{-5.0, 5.0\}$ and $D(Z_t) = \{-5.0, 20.0\}$. Notice that here DCBO and CBO explore the set $\mathbb{M}_t = \{\{X_t\}, \{Z_t\}, \{X_t, Z_t\}\}$ while BO and ABO intervene on $\{X_t, Z_t\}$. In this case, exploring $\mathbb{M}_t$ and propagating uncertainty in the causal prior slows down DCBO convergence, see Fig. E.5.

E.4.6 Non-stationary DAG and SEM (NONSTAT.)

The SCM used for this experiment is more complex than the others due to the fact that both the DAG and the SCM are non-stationary. We have:

$$
\begin{cases}
f(t) & \text{if } t = 0 \\
g(t) & \text{if } t = 1 \\
h(t) & \text{if } t = 2
\end{cases}
$$

where

$$
f(t) = \begin{cases}
X_t &= \epsilon_X \\
Z_t &= X_t + \epsilon_Z \\
Y_t &= \sqrt{|36 - (Z_t - 1)^2|} + 1 + \epsilon_Y
\end{cases}
$$

$$g(t) = \begin{cases} X_t &= X_{t-1} + \epsilon_X \\ Z_t &= -\frac{X_t}{X_{t-1}} + Z_{t-1} + \epsilon_Z \\ Y_t &= Z_t \cos(Z_t \pi) - Y_{t-1} + \epsilon_Y \end{cases}$$

$$h(t) = \begin{cases} X_t &= X_{t-1} + \epsilon_X \\ Z_t &= X_t + Z_{t-1} + \epsilon_Z \\ Y_t &= Z_t - Y_{t-1} - Z_{t-1} + \epsilon_Y \end{cases}$$

with $\epsilon_i \sim \mathcal{N}(0,1)$ for $i \in \{X, Z, Y\}$. We set $T = 3$, $N = 10$, $D(X_t) = \{-5.0, 5.0\}$ and $D(Z_t) = \{-5.0, 20.0\}$. Notice that here DCBO and CBO explore the set $\mathbb{M}_t = \{\{X_t\}, \{Z_t\}, \{X_t, Z_t\}\}$ while BO and ABO intervene on $\{X_t, Z_t\}$.

E.4.7 Real-World Economic data (ECON.)

We obtain an observational dataset by extracting the following indicators from the OECD data portal (`https://data.oecd.org/`):

- GDP = GDP in milion of US dollars.
- CPI = annual growth of inflation measured by consumer price index CPI.
- TAXREV = tax revenues measured as a percentage of GDP.
- HUR = unemployment rate as measured by the numbers of unemployed people as a percentage of the labour force.

We manipulate these indicators to get the nodes in the DAG of Fig. 8.4(a). We define the following variables:

$$\begin{aligned} U_t &= \log(\text{HUR}_t) \\ T_t &= \frac{\text{TAXREV}_t * \text{GDP}_t - \text{TAXREV}_{t-1} * \text{GDP}_{t-1}}{\text{TAXREV}_{t-1} * \text{GDP}_{t-1}} \\ G_t &= \frac{\text{GDP}_t - \text{GDP}_{t-1}}{\text{GDP}_{t-1}} \\ I_t &= \text{CPI}_t \end{aligned}$$

For this analysis we consider the annual data for the period 2000 - 2019 and for 10 countries that are Australia, Canada, France, Germany, Italy, Japan, Korea, Mexico, Turkey, Great Britain and the United States of America. We fit the

following SCM:

$$
\begin{aligned}
T_t &= f_T(t) + \epsilon_T \\
I_t &= f_I(t) + \epsilon_I \\
G_t &= f_G(T_t, I_t) + \epsilon_G \\
U_t &= f_U(G_t, I_t) + \epsilon_U
\end{aligned}
$$

by placing GPs on all functions $f_i(\cdot), i \in \{T, I, G, U\}$. This SCM is then used to generate interventional data and compute the values of $y_t^\star, t = 2010, \ldots, 2012$.

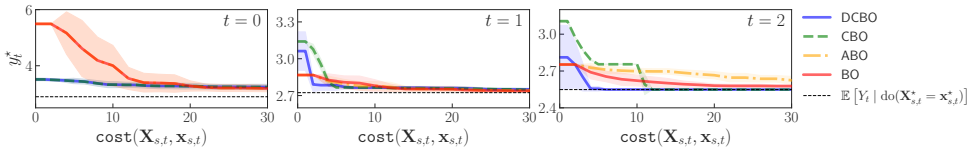


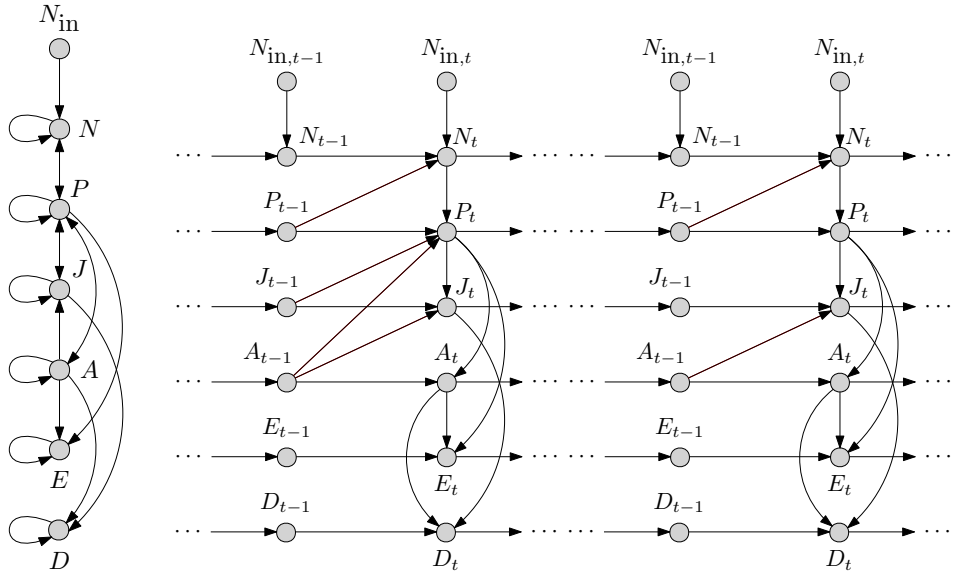
Figure E.6: Experiment ECON. Convergence of DCBO and competing methods across replicates. The black line gives the optimal $y_t^*, \forall t$. Shaded areas are $\pm$ one standard deviation.

We run the optimization algorithm for 10 times and plot the convergence path for DCBO and competing models (Fig. E.6). While all method perform similarly at $t = 2010$ and $t = 2011$, DCBO outperforms competing approaches at $t = 2012$. On average (Table 8.1) DCBO finds the optimal intervention faster.

E.4.8 Planktonic predator–prey community in a chemostat (EVOL.)

This experiment is based on the work by Blasius et al. [2020] in which they perform microcosm experiments in a chemostat to investigate a biological systems where two species interact, one as a predator and the other as prey. We use their system of ODE, which describes a stage-structured predator–prey community in a chemostat, and their experimental data collected in vitro as $\mathcal{D}^{O1}$ [1]. The DAG (Fig. E.7(c)) and SCM (Eq. (E.21)) are constructed from the system of ODE by rolling out the temporal variable dependencies, see [Bongers and Mooij, 2018; Hansen et al., 2014; Mooij et al., 2013b; Peters et al., 2020] for a review on how to interpret differential equations as causal models. The original rolled-out DAG (Fig. E.7(b)) is modified to remove graph cycles and simplify the causal dependencies on the phytoplankton (predator) concentration. The final DAG is given in Fig. E.7(c). We use DCBO to identify the optimal intervention to reduce the concentration of dead animals in the chemostat – D_t in Fig. E.7(c). The following variables are included in the DAG (we omit the

[1] We use data-files `C1.csv, C2.csv, C3.csv, C4.csv` from the original publication [Blasius et al., 2020] – available here: `https://figshare.com/articles/dataset/Time_series_of_long-term_experimental_predator-prey_cycles/10045976/1` [Accessed: 01/04/21].



(a) ODE variable dependencies. (b) First DAG approximation. (c) Second DAG approximation.

Figure E.7: DAGs representing the causal dependencies in the stage-structured predator–prey community in a chemostat. The nodes of the graph represent the concentrations of the different chemostat compounds at different discrete time points, where time is moving from left to right. (a) shows the variable dependencies as described in the original system of ODE – notice the presence of self-loops and cycles. (b) shows a first approximation to a corresponding causal graph, where the ODE has been 'rolled' out in time – note the absence of self-loops and cycles. (c) shows a second approximation to the original ODE dynamics but this time removing two parent dependencies from P_t.

time subscript):

- N_{in} = Nitrogen concentration in the external medium
- N = Nitrogen (prey) concentration
- P = Phytoplankton (predator) concentration
- J = Predator juvenile concentration
- A = Predator adult concentration
- E = Predator egg concentration
- D = Dead animal concentration

We fit the following SCM, based on the DAG in Fig. E.7(c):

$$N_{\text{in},t} = \epsilon_{N_{\text{in}}} \tag{E.15}$$
$$N_t = f_N(N_{\text{in},t}, N_{t-1}, P_{t-1}) + \epsilon_N \tag{E.16}$$
$$P_t = f_P(N_t, P_{t-1}) + \epsilon_P \tag{E.17}$$
$$J_t = f_J(P_t, J_{t-1}, A_{t-1}) + \epsilon_J \tag{E.18}$$
$$A_t = f_A(P_t, A_{t-1}) + \epsilon_A \tag{E.19}$$
$$E_t = f_E(P_t, A_t, E_{t-1}) + \epsilon_E \tag{E.20}$$
$$D_t = f_D(J_t, A_t, D_{t-1}) + \epsilon_D \tag{E.21}$$

by placing GPs on all functions $\{f_i(\cdot) \mid i \in \{N_{\text{in}}, N, P, E, J, A, D\}\}$. This SEM is then used to generate interventional data and compute the values of the optimal target variable $\{d_t^\star \mid t = 0, 1, 2\}$. Further, $\{\epsilon_j \sim \mathcal{N}(0, 1) \mid j \in \{N_{\text{in}}, N, P, E, J, A, D\}\}$. We set $T = 3$, $N = 4$ and let the manipulative variables be $N_{\text{in},t}, J_t$ and A_t. Intervention domains are given by:

$$D(N_{\text{in},t}) = [40.0, 160.0]$$
$$D(J_t) = [0.0, 20.0]$$
$$D(A_t) = [0.0, 100.0]$$

Notice that DCBO and CBO explore the set

$$\mathbb{M}_t = \{\{N_{\text{in},t}\}, \{J_t\}, \{A_t\}, \{N_{\text{in},t}, J_t\}, \{N_{\text{in},t}, A_t\}, \{J_t, A_t\}, \{N_{\text{in},t}, J_t, A_t\}\}$$

while BO and ABO will only intervene on $\{N_{\text{in},t}, J_t, A_t\}$. The optimal sequence of interventions is given by $\{\{J_0, A_0\}, \{M_1\}, \{M_2\}\}$. Results are shown in Fig. E.8.

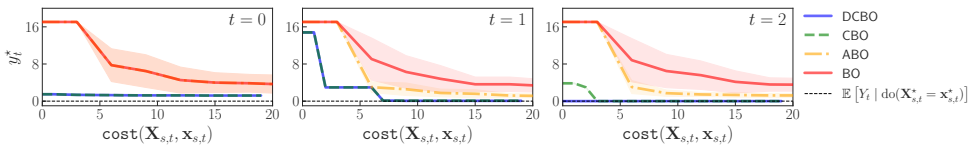


Figure E.8: Experiment EVOL. with maximum number of trials $H = 20$. Convergence of DCBO and competing methods across replicates. The black line gives the optimal $y_t^*, \forall t$. Shaded areas are $\pm$ one standard deviation.

E.4.9 Results without convergence

We repeat all experiments in the chapter allowing the algorithms to perform a lower number of trials at every time step. This means that, for $t > 0$, when moving to the next time step the convergence of the algorithm at the

previous step is not guaranteed. In turn, this affects the optimum value that the algorithm can reach at subsequent steps. Results are given in Table E.1 and Table E.2. The convergence paths for DCBO and competing methods are given in Fig. E.9 to Fig. E.13.

Table E.1: Average modified gap measure (10 replicates) across time steps and for different experiments. See Fig. 8.1 for a summary of the compared methods. Higher values are better. The best result for each experiment is bolded. Standard errors in brackets.

	Synthetic data						Real data	
	STAT.	MISS.	NOISY	MULTIV.	IND.	NONSTAT.	ECON.	EVOL.
DCBO	**0.88** (0.00)	**0.72** (0.07)	**0.73** (0.00)	**0.49** (0.00)	0.47 (0.05)	**0.47** (0.00)	0.40 (0.04)	**0.67** (0.00)
CBO	0.57 (0.02)	0.51 (0.09)	0.67 (0.01)	0.47 (0.04)	0.48 (0.04)	**0.47** (0.00)	**0.41** (0.04)	0.65 (0.00)
ABO	0.43 (0.06)	0.45 (0.04)	0.42 (0.06)	0.40 (0.05)	**0.50** (0.00)	0.41 (0.03)	0.38 (0.04)	0.47 (0.01)
BO	0.42 (0.06)	0.41 (0.05)	0.41 (0.07)	0.38 (0.07)	**0.50** (0.01)	0.40 (0.04)	0.40 (0.04)	0.46 (0.03)

Table E.2: Average percentage of replicates across time steps and for different experiments for which the optimal intervention set is identified. See Fig. 8.1 for a summary of the compared methods. Higher values are better. The best result for each experiment is bolded.

	Synthetic data						Real data	
	STAT.	MISS.	NOISY	MULTIV.	IND.	NONSTAT.	ECON.	EVOL.
DCBO	**90.0**	**70.00**	**93.00**	**93.33**	**96.67**	**66.67**	73.33	**33.33**
CBO	76.67	63.33	76.67	86.67	93.33	33.33	**80.00**	**33.33**
ABO	0.00	0.00	0.00	0.00	100.00	0.00	66.67	0.00
BO	0.00	0.00	0.00	0.00	100.00	0.00	66.67	0.00

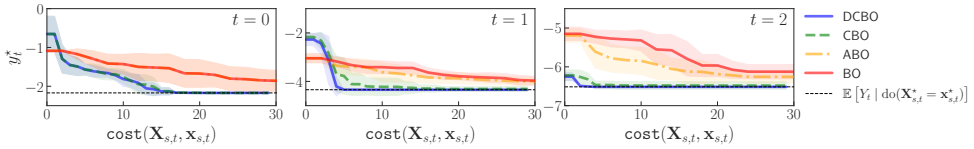


Figure E.9: Experiment STAT. with maximum number of trials $H = 30$. Convergence of DCBO and competing methods across replicates. The black line gives the optimal $y_t^*, \forall t$. Shaded areas are $\pm$ one standard deviation.

E.4.10 Results over multiple datasets and replicates

Finally, we repeat all experiments in the main paper by running DCBO and competing methods across 10 different observational dataset sampled from the SCM given above. Results are given in Table E.3.

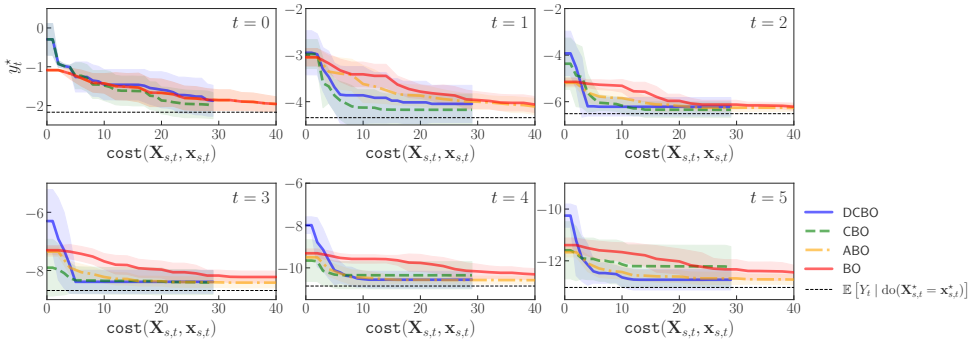


Figure E.10: Experiment Miss. with maximum number of trials $H = 30$. Convergence of DCBO and competing methods across replicates. The black line gives the optimal $y_t^*, \forall t$. Shaded areas are $\pm$ one standard deviation.

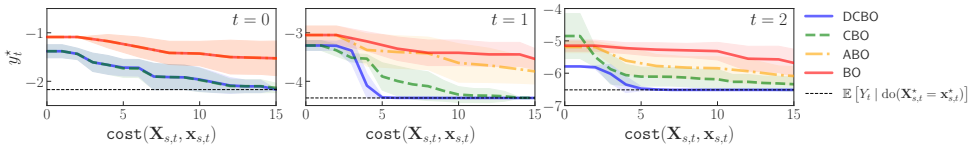


Figure E.11: Experiment Noisy. with maximum number of trials $H = 30$. Convergence of DCBO and competing methods across replicates. The black line gives the optimal $y_t^*, \forall t$. Shaded areas are $\pm$ one standard deviation.

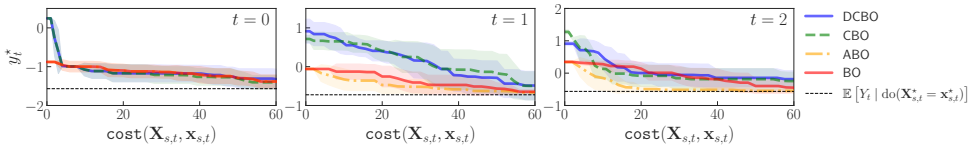


Figure E.12: Experiment Ind. with maximum number of trials $H = 30$. Convergence of DCBO and competing methods across replicates. The black line gives the optimal $y_t^*, \forall t$. Shaded areas are $\pm$ one standard deviation.

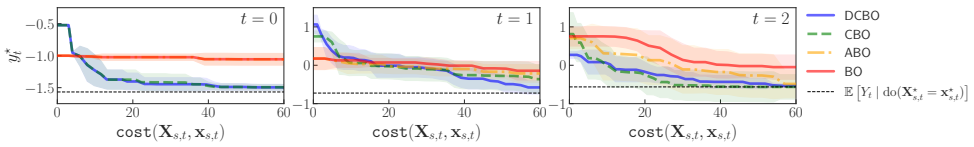


Figure E.13: Experiment Multiv. with maximum number of trials $H = 30$. Convergence of DCBO and competing methods across replicates. The black line gives the optimal $y_t^*, \forall t$. Shaded areas are $\pm$ one standard deviation.

Table E.3: Average modified gap measure across 10 observational datasets and 10 replicates. Results are average figures across time steps. See Fig. 8.1 for a summary of the compared methods. Higher values are better. The best result for each experiment is bolded. Standard errors in brackets.

	Synthetic data					
	Stat.	Miss.	Noisy	Multiv.	Ind.	NonStat.
DCBO	**0.83** (0.06)	**0.82** (0.05)	**0.82** (0.05)	**0.48** (0.02)	0.46 (0.03)	0.63 (0.06)
CBO	0.80 (0.05)	0.68 (0.04)	0.74 (0.09)	**0.48** (0.01)	0.47 (0.02)	**0.64** (0.04)
ABO	0.47 (0.01)	0.49 (0.00)	0.47 (0.01)	0.45 (0.08)	0.48 (0.00)	0.38 (0.01)
BO	0.47 (0.01)	0.47 (0.01)	0.47 (0.01)	0.40 (0.07)	**0.50** (0.00)	0.38 (0.01)

Bibliography

Odd O Aalen, Kjetil Røysland, Jon Michael Gran, and Bruno Ledergerber. Causality, mediation and time: a dynamic viewpoint. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 175(4):831–861, 2012.

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. TensorFlow: Large-scale Machine Learning on Heterogeneous Distributed Systems. *arXiv preprint arXiv:1603.04467*, 2016.

Alberto Abadie. Semiparametric difference-in-differences estimators. *The Review of Economic Studies*, 72(1):1–19, 2005.

Alberto Abadie. Using Synthetic Controls: Feasibility, Data Requirements, and Methodological Aspects. *Journal of Economic Literature*, 59(2):391–425, June 2021.

Alberto Abadie and Guido W Imbens. Large Sample Properties of Matching Estimators for Average Treatment Effects. *Econometrica*, 74(1):235–267, 2006.

Ryan Prescott Adams, Iain Murray, and David JC MacKay. Tractable Nonparametric Bayesian Inference in Poisson Processes with Gaussian Process Intensities. In *International Conference on Machine Learning*, pages 9–16. ACM, 2009.

Virginia Aglietti, Theodoros Damoulas, and Edwin V Bonilla. Efficient Inference in Multi-task Cox Process Models. In *Artificial Intelligence and Statistics*, pages 537–546. PMLR, 2019.

Mohamed Osama Ahmed, Bobak Shahriari, and Mark Schmidt. Do we need "harmless" bayesian optimization and "first-order" bayesian optimization. *NIPS BayesOpt*, 2016.

Ahmed M Alaa and Mihaela Van der Schaar. Bayesian Inference of Individualized Treatment Effects using Multi-task Gaussian processes. In *Neural Information Processing Systems*, pages 3424–3432, 2017.

Amit M Algotar, Patricia A Thompson, James Ranger-Moore, M Suzanne Stratton, Chiu-Hsieh Hsu, Frederick R Ahmann, Raymond B Nagle, and Steven P Stratton. Effect of aspirin, other NSAIDs, and statins on PSA and PSA velocity. *The Prostate*, 70(8):883–888, 2010.

Dionissi Aliprantis. A distinction between causal effects in structural and rubin causal models. Working papers (old series), Federal Reserve Bank of Cleveland, March 2015.

Mauricio Álvarez, David Luengo, and Neil D Lawrence. Latent Force Models. In *Artificial Intelligence and Statistics*, pages 9–16, 2009.

Mauricio A Álvarez and Neil D Lawrence. Computationally Efficient Convolved Multiple Output Gaussian Processes. *Journal of Machine Learning Research*, 12(May):1459–1500, 2011.

Mauricio A Alvarez, David Luengo, Michalis K Titsias, and Neil D Lawrence. Efficient Multioutput Gaussian Processes through Variational Inducing Kernels. *Artificial Intelligence and Statistics*, 9:25–32, 2010.

Ahsan S Alvi, Binxin Ru, Jan Calliess, Stephen J Roberts, and Michael A Osborne. Asynchronous Batch Bayesian Optimisation with Improved Local Penalisation. In *International Conference on Machine Learning*, 2019.

A. Andersson and N. Bates. In situ measurements used for coral and reef-scale calcification structural equation modeling including environmental and chemical measurements, and coral calcification rates in bermuda from 2010 to 2012 (beacon project). *Biological and Chemical Oceanography Data Management Office (BCO-DMO). Dataset version 2018-03-02. http://lod.bco-dmo.org/id/dataset/720788*, 2018.

Joshua D Angrist, Guido W Imbens, and Donald B Rubin. Identification of Causal Effects Using Instrumental Variables. *Journal of the American statistical Association*, 91(434):444–455, 1996.

Javad Azimi, Ali Jalali, and Xiaoli Fern. Dynamic Batch Bayesian Optimization. *arXiv preprint arXiv:1110.3347*, 2011.

Javad Azimi, Ali Jalali, and Xiaoli Fern. Hybrid Batch Bayesian Optimization. In *International Conference on Machine Learning*, 2012.

Elias Bareinboim and Judea Pearl. Causal Inference by Surrogate Experiments: z-Identifiability. In *Uncertainty in Artificial Intelligence*, 2012.

Elias Bareinboim and Judea Pearl. Meta-Transportability of Causal Effects: A Formal Approach. In *Artificial Intelligence and Statistics*, pages 135–143, 2013.

Elias Bareinboim and Judea Pearl. Transportability from Multiple Environments with Limited Experiments: Completeness Results. In *Neural Information Processing Systems*, pages 280–288, 2014.

Elias Bareinboim, Andrew Forney, and Judea Pearl. Bandits with Unobserved Confounders: A Causal Approach. In *Neural Information Processing Systems*, pages 1342–1350, 2015.

Matthias Bauer, Mark van der Wilk, and Carl Edward Rasmussen. Understanding Probabilistic Sparse Gaussian Process Approximations. In *Neural Information Processing Systems*, pages 1533–1541, 2016.

Matthew James Beal. *Variational Algorithms for Approximate Bayesian Inference*. University of London, University College London (United Kingdom), 2003.

Omar Besbes, Yonatan Gur, and Assaf Zeevi. Stochastic Multi-Armed-Bandit Problem with Non-stationary Rewards. In *Neural Information Processing Systems*, volume 27, pages 199–207. Citeseer, 2014.

Bernd Blasius, Lars Rudolf, Guntram Weithoff, Ursula Gaedke, and Gregor F Fussmann. Long-term cyclic persistence in an experimental predator–prey system. *Nature*, 577(7789):226–230, 2020.

David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational Inference: A Review for Statisticians. *Journal of the American statistical Association*, 112 (518):859–877, 2017.

Ilija Bogunovic, Jonathan Scarlett, and Volkan Cevher. Time-Varying Gaussian Process Bandit Optimization. In *Artificial Intelligence and Statistics*, pages 314–323. PMLR, 2016.

Stephan Bongers and Joris M Mooij. From Random Differential Equations to Structural Causal Models: the stochastic case. *arXiv preprint arXiv:1803.08784*, 2018.

Stephan Bongers, Jonas Peters, Bernhard Schölkopf, and Joris M Mooij. Theoretical Aspects of Cyclic Structural Causal Models. *arXiv preprint arXiv:1611.06221*, 2016.

Edwin V Bonilla, Kian M Chai, and Christopher Williams. Multi-task Gaussian Process Prediction. In *Neural Information Processing Systems*, pages 153–160, 2007.

Edwin V Bonilla, Karl Krauth, and Amir Dezfouli. Generic Inference in Latent Gaussian Process Models. *Journal of Machine Learning Research*, 20(117): 1–63, 2019.

Phelim P Boyle. Options: A Monte Carlo approach. *Journal of financial economics*, 4(3):323–338, 1977.

Nicola Branchini, Virginia Aglietti, and Theodoros Damoulas. Causal Entropy Optimization: Joint Optimization and Structure Learning. *In preparation for Artificial Intelligence and Statistics*, 2022.

Michael Braun and Jon McAuliffe. Variational inference for large-scale models of discrete choice. *Journal of the American Statistical Association*, 105(489): 324–335, 2010.

Anders Brix and Peter J Diggle. Spatiotemporal prediction for log-Gaussian Cox Processes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(4):823–841, 2001.

Steve Brooks, Andrew Gelman, Galin Jones, and Xiao-Li Meng. *Handbook of Markov Chain Monte Carlo*. CRC press, 2011.

Daphna Buchsbaum, Sophie Bridgers, Deena Skolnick Weisberg, and Alison Gopnik. The power of possibility: causal learning, counterfactual reasoning, and pretend play. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 367(1599):2202–2212, 2012.

Lars Buesing, Theophane Weber, Yori Zwols, Sebastien Racaniere, Arthur Guez, Jean-Baptiste Lespiau, and Nicolas Heess. Woulda, Coulda, Shoulda: Counterfactually-guided Policy Search. In *International Conference on Learning Representations*, 2019.

Martin Buhmann. A new class of radial basis functions with compact support. *Mathematics of Computation*, 70(233):307–318, 2001.

Rich Caruana. Multitask Learning. In *Learning to learn*, pages 95–133. Springer, 1998.

Kian Ming A Chai. Variational Multinomial Logit Gaussian Process. *Journal of Machine Learning Research*, 13:1745–1808, 2012.

Bo Chen, Rui M. Castro, and Andreas Krause. Joint Optimization and Variable Selection of High-Dimensional Gaussian Processes. In *International Conference on Machine Learning*, page 1379–1386, 2012.

David Maxwell Chickering. Optimal Structure Identification With Greedy Search. *Journal of Machine Learning Research*, 3(Nov):507–554, 2002.

WG Cochran and GM Cox. Experimental Design. *Inc., New York, NY*, 1957.

Jean-François Coeurjolly, Jesper Møller, and Rasmus Waagepetersen. Palm distributions for log Gaussian Cox processes. *Scandinavian Journal of Statistics*, 44(1):192–203, 2017.

Gregory F Cooper and Changwon Yoo. Causal Discovery from a Mixture of Experimental and Observational data. In *Uncertainty in Artificial Intelligence*, 1999.

Travis A Courtney, Mario Lebrato, Nicholas R Bates, Andrew Collins, Samantha J De Putron, Rebecca Garley, Rod Johnson, Juan-Carlos Molinero, Timothy J Noyes, Christopher L Sabine, et al. Environmental controls on modern scleractinian coral and reef-scale calcification. *Science advances*, 3 (11):e1701356, 2017.

David R Cox. Some statistical methods connected with series of events. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 129–164, 1955.

Dennis D Cox and Susan John. A statistical method for global optimization. In *[Proceedings] 1992 IEEE International Conference on Systems, Man, and Cybernetics*, pages 1241–1246. IEEE, 1992.

Cecil C Craig. On the frequency function of xy. *The Annals of Mathematical Statistics*, 7(1):1–15, 1936.

Carlos Cruz, Juan R González, and David A Pelta. Optimization in Dynamic Environments: A Survey on Problems, Methods and Measures. *Soft Computing*, 15(7):1427–1448, 2011.

Lehel Csató and Manfred Opper. Sparse Online Gaussian Processes. *Neural computation*, 14(3):641–668, 2002.

Francisco Cuevas-Pacheco and Jesper Møller. Log Gaussian Cox processes on the sphere. *Spatial Statistics*, 26:69–82, 2018.

John P Cunningham, Krishna V Shenoy, and Maneesh Sahani. Fast Gaussian Process Methods for Point Process Intensity Estimation. In *International Conference on Machine Learning*, pages 192–199, 2008.

Kurt Cutajar, Michael Osborne, John Cunningham, and Maurizio Filippone. Preconditioning Kernel Matrices. In *International Conference on Machine Learning*, pages 2529–2538. PMLR, 2016.

Zhenwen Dai, Andreas Damianou, James Hensman, and Neil Lawrence. Gaussian process models with parallelization and gpu acceleration. *arXiv preprint arXiv:1410.4984*, 2014.

Daryl J Daley and David Vere-Jones. *An introduction to the theory of point processes: volume I: elementary theory and methods*. Springer, 2003.

Andreas Damianou and Neil D Lawrence. Deep Gaussian Processes. In *Artificial Intelligence and Statistics*, pages 207–215. PMLR, 2013.

Meyer Kris De, Nasuto J Slawomir, and Bishop Mark. Stochastic Diffusion Search: Partial Function Evaluation in Swarm Intelligence Dynamic Optimisation. In *Stigmergic optimization*, pages 185–207. Springer, 2006.

Amir Dezfouli and Edwin V Bonilla. Scalable Inference for Gaussian Process Models with Black-Box Likelihoods. In *Neural Information Processing Systems*, pages 1414–1422, 2015.

Peter J Diggle, Paula Moraga, Barry Rowlingson, and Benjamin M Taylor. Spatial and Spatio-Temporal Log-Gaussian Cox Processes: Extending the Geostatistical Paradigm. *Statistical Science*, pages 542–563, 2013.

Christian Donner and Manfred Opper. Efficient Bayesian Inference of Sigmoidal Gaussian Cox Processes. *Journal of Machine Learning Research*, 19(67):1–34, 2018.

Nicolas Durrande, David Ginsbourger, and Olivier Roustant. Additive Kernels for Gaussian Process Modeling. *arXiv preprint arXiv:1103.4023*, 2011.

David Duvenaud, James Lloyd, Roger Grosse, Joshua Tenenbaum, and Ghahramani Zoubin. Structure Discovery in Nonparametric Regression through Compositional Kernel Search. In *International Conference on Machine Learning*, pages 1166–1174. PMLR, 2013.

David Eriksson, Michael Pearce, Jacob Gardner, Ryan D Turner, and Matthias Poloczek. Scalable Global Optimization via Local Bayesian Optimization. *Neural Information Processing Systems*, 32:5496–5507, 2019.

Stefan Falkner, Aaron Klein, and Frank Hutter. Combining hyperband and bayesian optimization. In *NIPS 2017 Bayesian Optimization Workshop (Dec 2017)*, 2017.

Tamara Fernandez, Nicolas Rivera, and Yee Whye Teh. Gaussian Processes for Survival Analysis. In *Neural Information Processing Systems*, volume 29, 2016.

Ana Ferro, Francisco Pina, Milton Severo, Pedro Dias, Francisco Botelho, and Nuno Lunet. Use of statins and serum levels of Prostate Specific Antigen. *Acta Urológica Portuguesa*, 32(2):71–77, 2015.

Maurizio Filippone, Mingjun Zhong, and Mark Girolami. A Comparative Evaluation of Stochastic-based Inference Methods for Gaussian Process Models. *Machine Learning*, 93(1):93–114, 2013.

Ronald Aylmer Fisher. Design of experiments. *Br Med J*, 1(3923):554–554, 1936.

Seth Flaxman, Andrew Wilson, Daniel Neill, Hannes Nickisch, and Alex Smola. Fast Kronecker Inference in Gaussian Processes with non-Gaussian Likelihoods. In *International Conference on Machine Learning*, pages 607–616, 2015.

Seth Flaxman, Michael Chirico, Pau Pereira, and Charles Loeffler. Scalable high-resolution forecasting of sparse spatiotemporal events with kernel methods: A winning solution to the NIJ "Real-Time Crime Forecasting Challenge". *The Annals of Applied Statistics*, 13(4):2564–2585, 2019.

Jakob N Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual Multi-Agent Policy Gradients. In *AAAI Conference on Artificial Intelligence*, 2018.

Lawrence J Fogel, Alvin J Owens, and Michael J Walsh. *Artificial Intelligence Through Simulated Evolution*. Wiley, 1966.

Charles W Fox and Stephen J Roberts. A tutorial on variational Bayesian inference. *Artificial intelligence review*, 38(2):85–95, 2012.

Roger Frigola, Yutian Chen, and Carl Edward Rasmussen. Variational Gaussian Process State-Space Models. In *Neural Information Processing Systems*, pages 3680–3688, 2014.

Mathias Frisch. *Causal reasoning in physics*. Cambridge University Press, 2014.

Yarin Gal, Mark van der Wilk, and Carl Rasmussen. Distributed Variational Inference in Sparse Gaussian Process Regression and Latent Variable Models. *Neural Information Processing Systems*, 2014a.

Yarin Gal, Mark Van Der Wilk, and Carl E Rasmussen. Distributed variational inference in sparse gaussian process regression and latent variable models. *arXiv preprint arXiv:1402.1389*, 2014b.

David Galles and Judea Pearl. Testing Identifiability of Causal Effects. In *Uncertainty in Artificial Intelligence*, 1995.

Jacob R Gardner, Geoff Pleiss, David Bindel, Kilian Q Weinberger, and Andrew Gordon Wilson. GPyTorch: Blackbox Matrix-Matrix Gaussian Process

Inference with GPU Acceleration. In *Neural Information Processing Systems*, 2018.

Roman Garnett, Michael A Osborne, and Stephen J Roberts. Bayesian Optimization for Sensor Set Selection. In *Proceedings of the 9th ACM/IEEE international conference on information processing in sensor networks*, pages 209–219, 2010.

Michael A Gelbart, Jasper Snoek, and Ryan P Adams. Bayesian Optimization with Unknown Constraints. In *Uncertainty in Artificial Intelligence*, 2014.

Samuel J Gershman. Reinforcement Learning and Causal Models. *The Oxford handbook of causal reasoning*, page 295, 2017.

Zoubin Ghahramani and Matthew J Beal. Propagation Algorithms for Variational Bayesian Learning. In *Neural Information Processing Systems*, pages 507–513. Citeseer, 2000.

Subhashis Ghosal and Anindya Roy. Posterior consistency of Gaussian process prior for nonparametric binary regression. *The Annals of Statistics*, 34(5): 2413–2429, 2006.

Clark Glymour, Kun Zhang, and Peter Spirtes. Review of Causal Discovery Methods Based on Graphical Models. *Frontiers in Genetics*, 10, 2019.

Tilmann Gneiting. Compactly supported correlation functions. *Journal of Multivariate Analysis*, 83(2):493–508, 2002.

David E Goldberg and Robert E Smith. Nonstationary function optimization using genetic algorithms with dominance and diploidy. In *Genetic algorithms and their applications: proceedings of the second International Conference on Genetic Algorithms: July 28-31, 1987 at the Massachusetts Institute of Technology, Cambridge, MA*. Hillsdale, NJ: L. Erlhaum Associates, 1987., 1987.

Javier González, Zhenwen Dai, Philipp Hennig, and Neil Lawrence. Batch Bayesian Optimization via Local Penalization. In *Artificial Intelligence and Statistics*, pages 648–657. PMLR, 2016a.

Javier González, Michael Osborne, and Neil Lawrence. GLASSES: Relieving the Myopia of Bayesian Optimisation. In *Artificial Intelligence and Statistics*, pages 790–799. PMLR, 2016b.

Pierre Goovaerts et al. *Geostatistics for Natural Resources Evaluation*. Oxford University Press on Demand, 1997.

Olivier Goudet, Diviyan Kalainathan, Philippe Caillou, Isabelle Guyon, David Lopez-Paz, and Michele Sebag. Learning Functional Causal Models with Generative Neural Networks. In *Explainable and interpretable models in computer vision and machine learning*, pages 39–80. Springer, 2018.

Robert B Gramacy. laGP: large-scale spatial modeling via local approximate Gaussian processes in R. *Journal of Statistical Software*, 72(1):1–46, 2016.

Clive WJ Granger. Investigating Causal Relations by Econometric Models and Cross-spectral Methods. *Econometrica*, pages 424–438, 1969.

Kristjan Greenewald, Dmitriy Katz, Karthikeyan Shanmugam, Sara Magliacane, Murat Kocaoglu, Enric Boix Adsera, and Guy Bresler. Sample Efficient Active Learning of Causal Trees. In *Neural Information Processing Systems*, pages 14279–14289, 2019.

Tony H Grubesic and Elizabeth A Mack. Spatio-temporal interaction of urban crime. *Journal of Quantitative Criminology*, 24(3):285–306, 2008.

Tom Gunter, Chris Lloyd, Michael A Osborne, and Stephen J Roberts. Efficient Bayesian Nonparametric Modelling of Structured Point Processes. In *Uncertainty in Artificial Intelligence*, pages 310–319, 2014.

Ruocheng Guo, Lu Cheng, Jundong Li, P Richard Hahn, and Huan Liu. A Survey of Learning Causality with Data: Problems and Methods. *ACM Comput. Surv.*, 53(4), 2020.

Trygve Haavelmo. The statistical implications of a system of simultaneous equations. *Econometrica, Journal of the Econometric Society*, pages 1–12, 1943.

P Richard Hahn, Carlos M Carvalho, David Puelz, and Jingyu He. Regularization and confounding in linear regression for treatment effect estimation. *Bayesian Analysis*, 13(1):163–182, 2018.

Niels Hansen, Alexander Sokol, et al. Causal interpretation of stochastic differential equations. *Electronic Journal of Probability*, 19, 2014.

Jouni Hartikainen and Simo Särkkä. Kalman filtering and smoothing solutions to temporal Gaussian process regression models. In *2010 IEEE international workshop on machine learning for signal processing*, pages 379–384. IEEE, 2010.

Alain Hauser and Peter Bühlmann. Characterization and Greedy Learning of Interventional Markov Equivalence Classes of Directed Acyclic Graphs. *Journal of Machine Learning Research*, 13(1):2409–2464, 2012.

Alain Hauser and Peter Bühlmann. Two Optimal Strategies for Active Learning of Causal Models from Interventional Data. *International Journal of Approximate Reasoning*, 55(4):926–939, 2014.

Marton Havasi, José Miguel Hernández-Lobato, and Juan José Murillo-Fuentes. Inference in Deep Gaussian Processes using Stochastic Gradient Hamiltonian Monte Carlo. In *Neural Information Processing Systems*, 2018.

Yang-Bo He and Zhi Geng. Active Learning of Causal Networks with Intervention Experiments and Optimal Designs. *Journal of Machine Learning Research*, 9(Nov):2523–2547, 2008.

James J Heckman and Edward Vytlacil. Structural Equations, Treatment Effects, and Econometric Policy Evaluation. *Econometrica*, 73(3):669–738, 2005.

Peter Hedström and Petri Ylikoski. Causal mechanisms in the social sciences. *Annual review of sociology*, 36:49–67, 2010.

Philipp Hennig and Christian J Schuler. Entropy search for information-efficient global optimization. *Journal of Machine Learning Research*, 13(6), 2012.

James Hensman, Nicolo Fusi, and Neil D Lawrence. Gaussian Processes for Big Data. *Uncertainty in Artificial Intelligence*, 2013.

James Hensman, Alexander Matthews, and Zoubin Ghahramani. Scalable Variational Gaussian Process Classification. *Artificial Intelligence and Statistics*, 2015.

James Hensman, Nicolas Durrande, Arno Solin, et al. Variational Fourier Features for Gaussian Processes. *J. Mach. Learn. Res.*, 18(1):5537–5588, 2017.

Kristian Bjørn Hessellund, Ganggang Xu, Yongtao Guan, and Rasmus Waagepetersen. Second order semi-parametric inference for multivariate log Gaussian Cox processes. *arXiv preprint arXiv:2012.02155*, 2020.

Dave Higdon. Space and Space-Time Modeling using Process Convolutions. In *Quantitative methods for current environmental issues*, pages 37–56. Springer, 2002.

Matthew D Hoffman and David M Blei. Structured Stochastic Variational Inference. In *Artificial Intelligence and Statistics*, pages 361–369, 2015.

Lars Hörmander. *The Analysis of Linear Partial Differential Operators III: Pseudo-Differential Operators*. Springer Science & Business Media, 2007.

Patrik O Hoyer, Shohei Shimizu, Antti J Kerminen, and Markus Palviainen. Estimation of causal effects using linear non-Gaussian causal models with hidden variables. *International Journal of Approximate Reasoning*, 49(2): 362–378, 2008.

Biwei Huang, Kun Zhang, Mingming Gong, and Clark Glymour. Causal Discovery and Forecasting in Nonstationary Environments with State-Space Models. In *International Conference on Machine Learning*, pages 2901–2910. PMLR, 2019.

Deng Huang, Theodore T Allen, William I Notz, and Ning Zeng. Global Optimization of Stochastic Black-Box Systems via Sequential Kriging Meta-Models. *Journal of global optimization*, 34(3):441–466, 2006.

Yimin Huang and Marco Valtorta. Pearl's Calculus of Intervention is Complete. In *Uncertainty in Artificial Intelligence*, UAI'06, page 217–224, 2006.

David Hume. *A treatise of human nature*. Courier Corporation, 2003.

Frank Hutter, Holger H Hoos, and Kevin Leyton-Brown. Sequential Model-Based Optimization for General Algorithm Configuration. In *International conference on learning and intelligent optimization*, pages 507–523. Springer, 2011.

Antti Hyttinen, Frederick Eberhardt, and Patrik O Hoyer. Learning Linear Cyclic Causal Models with Latent Variables. *Journal of Machine Learning Research*, 13(1):3387–3439, 2012.

Antti Hyttinen, Patrik O Hoyer, Frederick Eberhardt, and Matti Jarvisalo. Discovering Cyclic Causal Models with Latent Variables: A General SAT-Based Procedure. In *Uncertainty in Artificial Intelligence*, 2013.

Janine B Illian, Sigrunn H Sørbye, and Håvard Rue. A toolbox for fitting complex spatial point process models using integrated nested laplace approximation (inla). *The annals of applied statistics*, pages 1499–1530, 2012a.

Janine B Illian, Sara Martino, Sigrunn H Sørbye, Juan B Gallego-Fernández, María Zunzunegui, M Paz Esquivias, and Justin MJ Travis. Fitting complex ecological point process models with integrated nested laplace approximation. *Methods in Ecology and Evolution*, 4(4):305–315, 2013.

Janine Barbel Illian, Sigrunn Holbek Sørbye, Håvard Rue, and Ditte Katrine Hendrichsen. Using inla to fit a complex point process model with temporally varying effects-a case study. *Journal of Environmental Statistics*, 3(7), 2012b.

Guido W Imbens. Potential Outcome and Directed Acyclic Graph Approaches to Causality: Relevance for Empirical Practice in Economics. *Journal of Economic Literature*, 58(4):1129–79, 2020.

Guido W Imbens and Thomas Lemieux. Regression discontinuity designs: A guide to practice. *Journal of econometrics*, 142(2):615–635, 2008.

Tommi Jaakkola and Michael Jordan. A Variational Approach to Bayesian Logistic Regression Models and their Extensions. In *Sixth International Workshop on Artificial Intelligence and Statistics*, 1997.

Mahaveer Jain, John McDonough, Gahgene Gweon, Bhiksha Raj, and Carolyn Rose. An Unsupervised Dynamic Bayesian Network Approach to Measuring Speech Style Accommodationn. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 787–797, 2012.

Shali Jiang, Henry Chai, Javier Gonzalez, and Roman Garnett. BINOCULARS for Efficient, Nonmyopic Sequential Experimental Design. In *International Conference on Machine Learning*, pages 4794–4803. PMLR, 2020.

ST John and James Hensman. Large-Scale Cox Process Inference using Variational Fourier Features. In *International Conference on Machine Learning*, 2018.

Olatunji Johnson, Peter Diggle, and Emanuele Giorgi. A spatially discrete approximation to log-Gaussian Cox processes for modelling aggregated disease count data. *Statistics in Medicine*, 38(24):4871–4887, 2019.

Donald R Jones, Matthias Schonlau, and William J Welch. Efficient Global Optimization of Expensive Black-Box Functions. *Journal of Global optimization*, 13(4):455–492, 1998.

Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An Introduction to Variational Methods for Graphical Models. *Machine learning*, 37(2):183–233, 1999.

Andre G Journel and Charles J Huijbregts. *Mining Geostatistics*. The Blackburn Press, 1976.

Jean Kaddour, Qi Liu, Yuchen Zhu, Matt J Kusner, and Ricardo Silva. Graph Intervention Networks for Causal Effect Estimation. *arXiv preprint arXiv:2106.01939*, 2021.

Leslie Pack Kaelbling, Michael L Littman, and Andrew W Moore. Reinforcement learning: A survey. *Journal of artificial intelligence research*, 4:237–285, 1996.

Immanuel Kant and Paul Guyer. *The Cambridge edition of the works of Immanuel Kant*. Cambridge Univ. Press, 1996.

Nan Rosemary Ke, Olexa Bilaniuk, Anirudh Goyal, Stefan Bauer, Hugo Larochelle, Bernhard Schölkopf, Michael C Mozer, Chris Pal, and Yoshua Bengio. Learning Neural Causal Models from Unknown Interventions. *arXiv preprint arXiv:1910.01075*, 2019.

Sathiya Keerthi and Wei Chu. A matching pursuit approach to sparse Gaussian process regression. *Advances in neural information processing systems*, 18: 643, 2006.

Niki Kilbertus, Matt J Kusner, and Ricardo Silva. A Class of Algorithms for General Instrumental Variable M]odels. In *Neural Information Processing Systems*, 2020.

David Knowles and Tom Minka. Non-conjugate Variational Message Passing for Multinomial and Binary Regression. In *Neural Information Processing Systems*, volume 24, pages 1701–1709, 2011.

Daphne Koller and Nir Friedman. *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning*. The MIT Press, 2009. ISBN 0262013193.

Jan TA Koster. Markov properties of nonrecursive causal models. *The Annals of Statistics*, 24(5):2148–2177, 1996.

Andreas Krause, Ajit Singh, and Carlos Guestrin. Near-Optimal Sensor Placements in Gaussian processes: Theory, Efficient Algorithms and Empirical Studies. *Journal of Machine Learning Research*, 9(Feb):235–284, 2008.

Harold J Kushner. A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. *Journal of Basic Engineering*, 1964.

Vidhi Lalchand and Carl Edward Rasmussen. Approximate Inference for Fully Bayesian Gaussian Process Regression. In *Symposium on Advances in Approximate Bayesian Inference*, pages 1–12. PMLR, 2020.

Thomas A Lasko. Efficient Inference of Gaussian-Process-Modulated Renewal Processes with Application to Medical Event Data. In *Uncertainty in Artificial Intelligence*, volume 2014, page 469. NIH Public Access, 2014.

Finnian Lattimore, Tor Lattimore, and Mark D Reid. Causal Bandits: Learning Good Interventions via Causal Inference. In *Neural Information Processing Systems*, pages 1181–1189, 2016.

Tor Lattimore and Csaba Szepesvári. *Bandit Algorithms*. Cambridge University Press, 2020.

Philip W Lavori and Ree Dawson. A design for testing clinical strategies: biased adaptive within-subject randomization. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 163(1):29–38, 2000.

Philip W Lavori and Ree Dawson. Adaptive treatment strategies in chronic disease. *Annu. Rev. Med.*, 59:443–453, 2008.

Neil Lawrence, Matthias Seeger, and Ralf Herbrich. Fast Sparse Gaussian Process Methods: The Informative Vector Machine. In *Neural Information Processing Systems*, pages 609–616, 2003.

Miguel Lázaro-Gredilla and Aníbal R Figueiras-Vidal. Inter-domain Gaussian Processes for Sparse Inference using Inducing Features. In *Neural Information Processing Systems*, volume 22, pages 1087–1095. Citeseer, 2009.

Sanghack Lee and Elias Bareinboim. Structural Causal Bandits: Where to Intervene? In *Neural Information Processing Systems*, pages 2568–2578, 2018.

Sanghack Lee and Elias Bareinboim. Structural Causal Bandits with Non-Manipulable Variables. In *AAAI Conference on Artificial Intelligence*, 2019.

Sanghack Lee, Juan D Correa, and Elias Bareinboim. General Identifiability with Arbitrary Surrogate Experiments. In *Uncertainty in Artificial Intelligence*, pages 389–398. PMLR, 2020.

Thomas J Leininger, Alan E Gelfand, et al. Bayesian Inference and Model Assessment for Spatial Point Patterns Using Posterior Predictive Samples. *Bayesian Analysis*, 12(1):1–30, 2017.

PA W Lewis and Gerald S Shedler. Simulation of nonhomogeneous poisson processes by thinning. *Naval research logistics quarterly*, 26(3):403–413, 1979.

Wenzhao Lian, Ricardo Henao, Vinayak Rao, Joseph Lucas, and Lawrence Carin. A Multitask Point Process Predictive Model. *International Conference on Machine Learning*, pages 2030–2038, 2015.

Haitao Liu, Jianfei Cai, Yi Wang, and Yew Soon Ong. Generalized robust Bayesian committee machine for large-scale Gaussian process regression. In *International Conference on Machine Learning*, pages 3131–3140. PMLR, 2018.

Haitao Liu, Yew-Soon Ong, Xiaobo Shen, and Jianfei Cai. When Gaussian Process Meets Big Data: A Review of Scalable GPs. *IEEE transactions on neural networks and learning systems*, 31(11):4405–4423, 2020.

C. Lloyd, T. Gunter, M. A. Osborne, and S. J. Roberts. Variational Inference for Gaussian Process Modulated Poisson Processes. *International Conference on Machine Learning*, 2015.

Chris Lloyd, Tom Gunter, Tom Nickson, M Osborne, and Stephen J Roberts. Latent Point Process Allocation. *Journal of Machine Learning Research*, 2016.

James Lloyd, David Duvenaud, Roger Grosse, Joshua Tenenbaum, and Zoubin Ghahramani. Automatic Construction and Natural-Language Description of Nonparametric Regression Models. In *AAAI Conference on Artificial Intelligence*, 2014.

Andrés F López-Lopera, ST John, and Nicolas Durrande. Gaussian Process Modulated Cox Processes Under Linear Inequality Constraints. In *Artificial Intelligence and Statistics*, pages 1997–2006. PMLR, 2019.

Christos Louizos, Uri Shalit, Joris Mooij, David Sontag, Richard Zemel, and Max Welling. Causal Effect Inference with Deep Latent-Variable Models. In *Neural Information Processing Systems*. PMLR, 2017.

Chaochao Lu, Bernhard Schölkopf, and José Miguel Hernández-Lobato. Deconfounding Reinforcement Learning in Observational Settings. *arXiv preprint arXiv:1812.10576*, 2018.

Christopher G Lucas and Thomas L Griffiths. Learning the Form of Causal Relationships Using Hierarchical Bayesian Models. *Cognitive Science*, 34(1): 113–147, 2010.

Jared K Lunceford, Marie Davidian, and Anastasios A Tsiatis. Estimation of survival distributions of treatment policies in two-stage randomization designs in clinical trials. *Biometrics*, 58(1):48–57, 2002.

David JC MacKay. Comparison of approximate methods for handling hyperparameters. *Neural computation*, 11(5):1035–1068, 1999.

Prashan Madumal, Tim Miller, Liz Sonenberg, and Frank Vetere. Explainable Reinforcement Learning Through a Causal Lens. In *AAAI Conference on Artificial Intelligence*, pages 2493–2500, 2020.

Sara Magliacane, Thijs van Ommen, Tom Claassen, Stephan Bongers, Philip Versteeg, and Joris M Mooij. Domain Adaptation by Using Causal Inference to

Predict Invariant Conditional Distributions. In *Neural Information Processing Systems*, pages 10846–10856, 2018.

Roman Marchant and Fabio Ramos. Bayesian optimisation for Intelligent Environmental Monitoring. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 2242–2249. IEEE, 2012.

David Marsan and Olivier Lengline. Extending earthquakes' reach through cascading. *Science*, 319(5866):1076–1079, 2008.

Ruben Martinez-Cantin, Nando de Freitas, Arnaud Doucet, and José A Castellanos. Active Policy Learning for Robot Planning and Exploration under Uncertainty. In *Robotics: Science and systems*, volume 3, pages 321–328, 2007.

Alexander G de G Matthews, James Hensman, Richard Turner, and Zoubin Ghahramani. On Sparse Variational Methods and the Kullback-Leibler Divergence between Stochastic Processes. In *Artificial Intelligence and Statistics*, pages 231–239. PMLR, 2016.

Alexander G. de G. Matthews, Mark van der Wilk, Tom Nickson, Keisuke. Fujii, Alexis Boukouvalas, Pablo León-Villagrá, Zoubin Ghahramani, and James Hensman. GPflow: A Gaussian Process Library using TensorFlow. In *Journal of Machine Learning Research*, volume 18, pages 1–6, apr 2017.

Alexander Graeme de Garis Matthews. *Scalable Gaussian process inference using variational methods*. PhD thesis, University of Cambridge, 2017.

Mark McLeod, Michael A Osborne, and Stephen J Roberts. Practical Bayesian Optimization for Variable Cost Objectives. *arXiv preprint arXiv:1703.04335*, 2017.

Arman Melkumyan and Fabio Tozeto Ramos. A sparse covariance function for exact gaussian process inference in large datasets. In *Twenty-first international joint conference on artificial intelligence*, 2009.

Albert Michotte. *The perception of causality*, volume 21. Routledge, 2017.

Jonas Mockus. *Bayesian Approach to Global Optimization: Theory and Applications*, volume 37. Springer Science & Business Media, 2012.

Jonas Mockus, Vytautas Tiesis, and Antanas Zilinskas. The Application of Bayesian Methods for Seeking the Extremum. *Towards Global Optimization*, 2(117-129):2, 1978.

Jesper Moller and Rasmus Plenge Waagepetersen. *Statistical inference and simulation for spatial point processes*. CRC Press, 2003.

Jesper Møller, Anne Randi Syversveen, and Rasmus Plenge Waagepetersen. Log Gaussian Cox Processes. *Scandinavian journal of statistics*, 25(3):451–482, 1998.

Joris M Mooij, Dominik Janzing, Tom Heskes, and Bernhard Schölkopf. On Causal Discovery with Cyclic Additive Noise Model. In *Neural Information Processing Systems*, 2011.

Joris M Mooij, Dominik Janzing, and Bernhard Schölkopf. From Ordinary Differential Equations to Structural Causal Models: the deterministic case. In *Uncertainty in Artificial Intelligence*, 2013a.

Joris M. Mooij, Dominik Janzing, and Bernhard Schölkopf. From Ordinary Differential Equations to Structural Causal Models: The Deterministic Case. In *Uncertainty in Artificial Intelligence*, UAI'13, page 440–448, Arlington, Virginia, USA, 2013b. AUAI Press.

Pablo Moreno-Muñoz, Antonio Artés-Rodríguez, and Mauricio A Álvarez. Heterogeneous Multi-output Gaussian Process Prediction. In *Neural Information Processing Systems*, 2018.

Riccardo Moriconi, Marc P Deisenroth, and KS Kumar. High-dimensional bayesian optimization using low-dimensional feature spaces. *arXiv preprint arXiv:1902.10675*, 2019.

Henry Moss, David Leslie, Daniel Beck, Javier González, and Paul Rayson. BOSS: Bayesian Optimization over String Spaces. In *Neural Information Processing Systems*, volume 33, 2020.

Kevin Patrick Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. University of California, Berkeley, 2002.

Susan A Murphy. Optimal dynamic treatment regimes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 65(2):331–355, 2003.

Susan A Murphy. An experimental design for the development of adaptive treatment strategies. *Statistics in medicine*, 24(10):1455–1481, 2005.

Iain Murray, Zoubin Ghahramani, and David J. C. MacKay. MCMC for doubly-intractable distributions. In *Uncertainty in Artificial Intelligence*, pages 359–366, 2006.

Fariba Nasirzadeh, Zohreh Shishebor, and Jorge Mateu. On new families of anisotropic spatial log-Gaussian Cox processes. *Stochastic Environmental Research and Risk Assessment*, 35(2):183–213, 2021.

Radford M Neal. Monte Carlo Implementation of Gaussian Process Models for Bayesian Regression and Classification. *arXiv preprint physics/9701026*, 1997.

Radford M Neal. On Deducing Conditional Independence from d-Separation in Causal Graphs with Feedback (Research Note). *Journal of Machine Learning Research*, 12:87–91, 2000.

Trung V Nguyen and Edwin V Bonilla. Automated Variational Inference for Gaussian Process Models. In *Neural Information Processing Systems*, pages 1404–1412, 2014.

Favour M. Nyikosa, Michael A. Osborne, and Stephen J. Roberts. Bayesian Optimization for Dynamic Problems, 2018.

Pedro A Ortega and Daniel A Braun. Generalized Thompson Sampling for Sequential Decision-Making and Causal Inference. *Complex Adaptive Systems Modeling*, 2(1):2, 2014.

John Paisley, David Blei, and Michael Jordan. Variational Bayesian Inference with Stochastic Search. In *International Conference on Machine Learning*, 2012.

Judea Pearl. [Bayesian Analysis in Expert Systems]: Comment: Gaphical models, Causality and Intervention. *Statistical Science*, 8(3):266–269, 1993.

Judea Pearl. Causal Diagrams for Empirical Research. *Biometrika*, 82(4): 669–688, 1995.

Judea Pearl. Causal inference in statistics: An overview. *Statistics surveys*, 3: 96–146, 2009a.

Judea Pearl. *Causality*. Cambridge university press, 2009b.

Judea Pearl and Elias Bareinboim. Transportability of Causal and Statistical Relations: A Formal Approach. In *AAAI Conference on Artificial Intelligence*, 2011.

Judea Pearl and Rina Dechter. Identifying Independencies in Causal Graphs with Feedback. In *Uncertainty in Artificial Intelligence*, 1996.

Judea Pearl and Dana Mackenzie. *The Book of Why: The New Science of Cause and Effect*. Basic books, 2018.

David Pelta, Carlos Cruz, and José L Verdegay. Simple control rules in a cooperative system for dynamic optimisation problems. *International Journal of General Systems*, 38(7):701–717, 2009.

Derek C Penn and Daniel J Povinelli. Causal cognition in human and nonhuman animals: a comparative, critical review. *Annu. Rev. Psychol.*, 58:97–118, 2007.

Bruno-Edouard Perrin, Liva Ralaivola, Aurelien Mazurie, Samuele Bottani, Jacques Mallet, and Florence d'Alche Buc. Gene networks inference using dynamic Bayesian networks. *Bioinformatics*, 19(suppl_2):ii138–ii148, 2003.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. Causal Inference on Time Series using Restricted Structural Equation Models. In *Neural Information Processing Systems*, volume 26, 2013.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of Causal Inference: Foundations and Learning Algorithms*. The MIT Press, 2017.

Jonas Peters, Stefan Bauer, and Niklas Pfister. Causal models for dynamical systems. *arXiv preprint arXiv:2001.06208*, 2020.

Niklas Pfister, Peter Bühlmann, and Jonas Peters. Invariant Causal Prediction for Sequential Data. *Journal of the American Statistical Association*, 114 (527):1264–1276, 2019.

Geoff Pleiss, Jacob Gardner, Kilian Weinberger, and Andrew Gordon Wilson. Constant-Time Predictive Distributions for Gaussian Processes. In *International Conference on Machine Learning*, pages 4114–4123. PMLR, 2018.

Joaquin Quinonero-Candela and Carl Edward Rasmussen. A Unifying View of Sparse Approximate Gaussian Process Regression. *Journal of Machine Learning Research*, 6:1939–1959, 2005.

Barbara Rakitsch, Christoph Lippert, Karsten Borgwardt, and Oliver Stegle. It is all in the noise: Efficient multi-task Gaussian process inference with structured residuals. In *Neural Information Processing Systems*, volume 26, pages 1466–1474, 2013.

Rajesh Ranganath, Sean Gerrish, and David Blei. Black Box Variational Inference. In *Artificial Intelligence and Statistics*, pages 814–822. PMLR, 2014.

Vinayak A Rao and Yee Whye Teh. Gaussian Process Modulated Renewal Processes. In *Neural Information Processing Systems*, pages 2474–2482, 2011.

Carl Edward Rasmussen and Zoubin Ghahramani. Infinite mixtures of Gaussian process experts. *Advances in neural information processing systems*, 2:881–888, 2002.

Sami Remes, Markus Heinonen, and Samuel Kaski. Non-Stationary Spectral Kernels. In *Neural Information Processing Systems*, 2017.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic Backpropagation and Approximate Inference in Deep Generative Models. In *International Conference on Machine Learning*, pages 1278–1286. PMLR, 2014.

Thomas S Richardson. A Discovery Algorithm for Directed Cyclic Graphs. In *Uncertainty in Artificial Intelligence*, 2013.

Mateo Rojas-Carulla, Bernhard Schölkopf, Richard Turner, and Jonas Peters. Invariant Models for Causal Transfer Learning. *Journal of Machine Learning Research*, 19(1):1309–1342, 2018.

Simone Rossi, Markus Heinonen, Edwin Bonilla, Zheyang Shen, and Maurizio Filippone. Sparse Gaussian Processes Revisited: Bayesian Approaches to Inducing-Variable Approximations. In *International Conference on Machine Learning*, pages 1837–1845. PMLR, 2021.

Dominik Rothenhäusler, Christina Heinze, Jonas Peters, and Nicolai Meinshausen. BACKSHIFT: Learning causal cyclic graphs from unknown shift interventions. In *Neural Information Processing Systems*, 2015.

Binxin Ru, Ahsan Alvi, Vu Nguyen, Michael A Osborne, and Stephen Roberts. Bayesian Optimisation over Multiple Continuous and Categorical Inputs. In *International Conference on Machine Learning*, pages 8276–8285. PMLR, 2020.

Paul K Rubenstein, Ilya Tolstikhin, Philipp Hennig, and Bernhard Schölkopf. Probabilistic active learning of functions in structural causal models. *Causality Workshop at Uncertainty in Artificial Intelligence. arXiv preprint arXiv:1706.10234*, 2017a.

Paul K Rubenstein, Ilya Tolstikhin, Philipp Hennig, and Bernhard Schölkopf. Probabilistic Active Learning of Functions in Structural Causal Models. *Causality Workshop. Uncertainty in Artificial Intelligence*, 2017b.

Donald B Rubin. Matching to remove bias in observational studies. *Biometrics*, pages 159–183, 1973.

Donald B Rubin. Causal Inference Using Potential Outcomes: Design, Modeling, Decisions. *Journal of the American Statistical Association*, 100(469):322–331, 2005.

Håvard Rue, Sara Martino, and Nicolas Chopin. Approximate bayesian inference for latent gaussian models by using integrated nested laplace approximations. *Journal of the royal statistical society: Series b (statistical methodology)*, 71 (2):319–392, 2009.

Yunus Saatçi. *Scalable inference for structured Gaussian process models*. PhD thesis, Citeseer, 2012.

Yves-Laurent Kom Samo and Stephen Roberts. Scalable nonparametric bayesian inference on point processes with gaussian processes. In *International Conference on Machine Learning*, pages 2227–2236. PMLR, 2015.

Yves-Laurent Kom Samo and Stephen J Roberts. String and Membrane Gaussian Processes. *Journal of Machine Learning Research*, 17(1):4485–4571, 2016.

Francesca Sargolini, Marianne Fyhn, Torkel Hafting, Bruce L McNaughton, Menno P Witter, May-Britt Moser, and Edvard I Moser. Conjunctive representation of position, direction, and velocity in entorhinal cortex. *Science*, 312(5774):758–762, 2006.

Simo Sarkka and Jouni Hartikainen. Infinite-Dimensional Kalman Filtering Approach to Spatio-Temporal Gaussian Process Regression. In *Artificial Intelligence and Statistics*, pages 993–1001. PMLR, 2012.

Simo Särkkä, Arno Solin, and Jouni Hartikainen. Spatiotemporal Learning via Infinite-Dimensional Bayesian Filtering and Smoothing: A Look at Gaussian Process Regression Through kalman Filtering. *IEEE Signal Processing Magazine*, 30(4):51–61, 2013.

Alexandra M Schmidt and Alan E Gelfand. A bayesian coregionalization approach for multivariate pollutant data. *Journal of Geophysical Research: Atmospheres*, 108(D24), 2003.

Bernhard Schölkopf, Alexander J Smola, Francis Bach, et al. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT press, 2002.

Matthias Seeger. Bayesian Gaussian process models: PAC-Bayesian generalisation error bounds and sparse approximations. Technical report, University of Edinburgh, 2003.

Matthias W Seeger, Christopher KI Williams, and Neil D Lawrence. Fast Forward Selection to Speed Up Sparse Gaussian Process Regression. In *International Workshop on Artificial Intelligence and Statistics*, pages 254–261. PMLR, 2003.

Antonio Seijas-Macías and Amílcar Oliveira. An approach to distribution of the product of two normal variables. *Discussiones Mathematicae Probability and Statistics*, 32(1-2):87–99, 2012.

Burr Settles. Active Learning. *Synthesis lectures on artificial intelligence and machine learning*, 6(1):1–114, 2012.

Amar Shah, Andrew Wilson, and Zoubin Ghahramani. Student-t Processes as Alternatives to Gaussian Processes. In *Artificial Intelligence and Statistics*, pages 877–885. PMLR, 2014.

Amar Shah, David Knowles, and Zoubin Ghahramani. An Empirical Study of Stochastic Variational Inference Algorithms for the Beta Bernoulli Process. In *International Conference on Machine Learning*, pages 1594–1603. PMLR, 2015.

Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando De Freitas. Taking the Human Out of the Loop: A Review of Bayesian Optimization. *Proceedings of the IEEE*, 104(1):148–175, 2015.

Shinichiro Shirota and Sudipto Banerjee. Scalable inference for space-time Gaussian Cox processes. *Journal of Time Series Analysis*, 40(3):269–287, 2019.

Shinichiro Shirota and Alan E Gelfand. Inference for log Gaussian Cox processes using an approximate marginal posterior. *arXiv preprint arXiv:1611.10359*, 2016.

Shinichiro Shirota and Alan E. Gelfand. Space and Circular Time Log-Gaussian Cox Process with Application to Crime Event Data. *The Annals of Applied Statistics*, 11(2):481–503, 2017.

Ilya Shpitser and Judea Pearl. Identification of Joint Interventional Distributions in Recursive Semi-Markovian Causal Models. In *AAAI Conference on Artificial Intelligence*, page 1219. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2006.

Ricardo Silva. Observational-Interventional Priors for Dose-Response Learning. In *Neural Information Processing Systems*, 2016.

Ricardo Silva and Robert B Gramacy. Gaussian Process Structural Equation Models with Latent Variables. In *Uncertainty in Artificial Intelligence*, 2010.

Ricardo Silva, Clark Glymour, and Peter Spirtes. Learning the Structure of Linear Latent Variable Models. *Journal of Machine Learning Research*, 7: 2006, 2005.

D. Simpson, J. B. Illian, F. Lindgren, S. H. Sørbye, and H. Rue. Going off grid: computationally efficient inference for log-Gaussian Cox processes. *Biometrika*, 103(1):49–70, 2016a.

Daniel Simpson, Janine Baerbel Illian, Finn Lindgren, Sigrunn H Sørbye, and Havard Rue. Going off grid: Computationally efficient inference for log-gaussian cox processes. *Biometrika*, 103(1):49–70, 2016b.

Grigorios Skolidis and Guido Sanguinetti. Bayesian Multitask Classification with Gaussian Process Priors. *IEEE Transactions on Neural Networks*, 22 (12), 2011.

Aleksandrs Slivkin. Introduction to Multi-Armed Bandits. *Foundations and Trends® in Machine Learning*, 12(1-2):1–286, 2019.

Steven A Sloman and David Lagnado. Causality in thought. *Annual review of psychology*, 66:223–247, 2015.

Alex J Smola and Peter L Bartlett. Sparse Greedy Gaussian Process Regression. In *Neural Information Processing Systems*, pages 619–625, 2001.

Edward Snelson and Zoubin Ghahramani. Sparse Gaussian Processes using Pseudo-inputs. *Neural Information Processing Systems*, 18:1257–1264, 2005.

Jasper Snoek, Kevin Swersky, Rich Zemel, and Ryan Adams. Input Warping for Bayesian Optimization of Non-Stationary Functions. In *International Conference on Machine Learning*, pages 1674–1682. PMLR, 2014.

Jasper Snoek, Oren Rippel, Kevin Swersky, Ryan Kiros, Nadathur Satish, Narayanan Sundaram, Mostofa Patwary, Mr Prabhat, and Ryan Adams. Scalable Bayesian Optimization using Deep Neural Networks. In *International Conference on Machine Learning*, pages 2171–2180. PMLR, 2015.

Jialin Song, Yuxin Chen, and Yisong Yue. A General Framework for Multi-fidelity Bayesian Optimization with Gaussian Processes. In *Artificial Intelligence and Statistics*, pages 3158–3167. PMLR, 2019.

Peter Spirtes, Clark N Glymour, Richard Scheines, and David Heckerman. *Causation, Prediction, and Search*. MIT press, 2000.

Peter L Spirtes. Directed Cyclic Graphical Representations of Feedback Models. In *Uncertainty in Artificial Intelligence*, 1995.

Jerzy Splawa-Neyman, Dorota M Dabrowska, and TP Speed. On the application of probability theory to agricultural experiments. essay on principles. section 9. *Statistical Science*, pages 465–472, 1990.

Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias W Seeger. Information-Theoretic Regret Bounds for Gaussian Process Optimization in the Bandit Setting. *IEEE Transactions on Information Theory*, 58(5): 3250–3265, 2012.

Ingo Steinwart. On the Influence of the Kernel on the Consistency of Support Vector Machines. *Journal of Machine Learning Research*, 2(Nov):67–93, 2001.

Roy L Streit. *Poisson Point Processes: Imaging, Tracking, and Sensing*. Springer Science & Business Media, 2010.

Xiaohai Sun, Dominik Janzing, Bernhard Schölkopf, and Kenji Fukumizu. A Kernel-based Causal Learning Algorithm. In *International Conference on Machine Learning*, pages 855–862, 2007.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

Kevin Swersky, Jasper Snoek, and Ryan P. Adams. Multi-Task Bayesian Optimization. In *Neural Information Processing Systems*, page 2004–2012, 2013.

Benjamin Taylor, Tilman Davies, Barry Rowlingson, and Peter Diggle. Bayesian Inference and Data Augmentation Schemes for Spatial, Spatiotemporal and Multivariate Log-Gaussian Cox Processes in r. *Journal of Statistical Software*, 63:1–48, 2015.

Yee Whye Teh, Matthias Seeger, and Michael I. Jordan. Semiparametric Latent Factor Models. In *Artificial Intelligence and Statistics*, 2005a.

Yee Whye Teh, Matthias Seeger, and Michael I. Jordan. Semiparametric Latent Factor Models. In *Artificial Intelligence and Statistics*, 2005b.

Clay Thompson. Causal Graph Analysis with the CAUSALGRAPH Procedure. `https://www.sas.com/content/dam/SAS/support/en/sas-global-forum-proceedings/2019/2998-2019.pdf`, 2019.

Jin Tian and Judea Pearl. A General Identification Condition for Eausal effects. In *AAAI Conference on Artificial Intelligence*, pages 567–573, 2002.

Michalis Titsias. Variational Learning of Inducing Variables in Sparse Gaussian Processes. In *Artificial Intelligence and Statistics*, pages 567–574. PMLR, 2009a.

Michalis Titsias and Neil D Lawrence. Bayesian Gaussian Process Latent Variable Model. In *Artificial Intelligence and Statistics*, pages 844–851, 2010.

Michalis K Titsias. Variational Learning of Inducing Variables in Sparse Gaussian Processes. In *Artificial Intelligence and Statistics*, volume 5, pages 567–574, 2009b.

Michalis K Titsias and Miguel Lázaro-Gredilla. Spike and Slab Variational Inference for Multi-Task and Multiple Kernel Learning. In *Neural Information Processing Systems*, pages 2339–2347, 2011.

Michel Tokic. Adaptive ε-Greedy Exploration in Reinforcement Learning Based on Value Differences. In *Annual Conference on Artificial Intelligence*, pages 203–210. Springer, 2010.

Jean-Francois Ton, Seth Flaxman, Dino Sejdinovic, and Samir Bhatt. Spatial Mapping with Gaussian Processes and Nonstationary Fourier Features. *Spatial statistics*, 28:59–78, 2018.

Marc Toussaint. The Bayesian Search Game. In *Theory and Principled Methods for the Design of Metaheuristics*, pages 129–144. Springer, 2014.

Dustin Tran, Rajesh Ranganath, and David M Blei. The Variational Gaussian Process. In *International Conference on Learning Representations*, 2016.

Krzysztof Trojanowski and Sławomir T Wierzchoń. Immune-based algorithms for dynamic optimization. *Information Sciences*, 179(10):1495–1515, 2009.

Mark van der Wilk, Carl Edward Rasmussen, and James Hensman. Convolutional Gaussian Processes. In *Neural Information Processing Systems*, volume 30, 2017.

Jay M Ver Hoef and Ronald Paul Barry. Constructing and fitting models for cokriging and multivariable spatial prediction. *Journal of Statistical Planning and Inference*, 69(2):275–294, 1998.

Sofía S Villar, Jack Bowden, and James Wason. Multi-armed Bandit Models for the Optimal Design of Clinical Trials: Benefits and Challenges. *Statistical science: a review journal of the Institute of Mathematical Statistics*, 30(2): 199, 2015.

Takashi Wada and Hideitsu Hino. Bayesian Optimization for Multi-objective Optimization and Multi-point Search. *arXiv preprint arXiv:1905.02370*, 2019.

Abdus S Wahed and Anastasios A Tsiatis. Semiparametric efficient estimation of survival distributions in two-stage randomisation designs in clinical trials with censored data. *Biometrika*, 93(1):163–177, 2006.

Christian J. Walder and Adrian N. Bishop. Fast Bayesian Intensity Estimation for the Permanental Process. *International Conference on Machine Learning*, 2017.

Xingchen Wan, Vu Nguyen, Huong Ha, Binxin Ru, Cong Lu, and Michael A Osborne. Think Global and Act Local: Bayesian Optimisation over High-Dimensional Categorical and Mixed Search Spaces. In *International Conference on Machine Learning*, 2021.

Kangrui Wang, Oliver Hamelijnck, Theodoros Damoulas, and Mark Steel. Non-Separable Non-Stationary Random Fields. In *International Conference on Machine Learning*, pages 9887–9897. PMLR, 2020.

Ke Wang, Geoff Pleiss, Jacob Gardner, Stephen Tyree, Kilian Q Weinberger, and Andrew Gordon Wilson. Exact Gaussian Processes on a Million Data Points. In *Neural Information Processing Systems*, volume 32, pages 14648–14659, 2019.

Tao Wang, Qian Diao, Yimin Zhang, Gang Song, Chunrong Lai, and Gary Bradski. A dynamic Bayesian network approach to multi-cue based visual tracking. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 2, pages 167–170. IEEE, 2004.

Zi Wang and Stefanie Jegelka. Max-value Entropy Search for Efficient Bayesian Optimization. In *International Conference on Machine Learning*, pages 3627–3635. PMLR, 2017.

Ziyu Wang and Nando de Freitas. Theoretical Analysis of Bayesian Optimisation with Unknown Gaussian Process Hyper-Parameters. *arXiv preprint arXiv:1406.7758*, 2014.

Ziyu Wang, Masrour Zoghi, Frank Hutter, David Matheson, Nando De Freitas, et al. Bayesian Optimization in High Dimensions via Random Embeddings. In *International Joint Conference on Artificial Intelligence*, pages 1778–1784, 2013.

Ziyu Wang, Frank Hutter, Masrour Zoghi, David Matheson, and Nando de Feitas. Bayesian Optimization in a Billion Dimensions via Random Embeddings. *Journal of Artificial Intelligence Research*, 55:361–387, 2016.

Larry Wasserman. *All of Nonparametric Statistics*. Springer Science & Business Media, 2006.

Christopher Williams and Matthias Seeger. Using the Nyström Method to Speed Up Kernel Machines. In *Neural Information Processing Systems*, pages 682–688, 2001.

Christopher KI Williams and Carl Edward Rasmussen. Gaussian Processes for Machine Learning. *the MIT Press*, 2(3):4, 2006.

Andrew Wilson and Hannes Nickisch. Kernel Interpolation for Scalable Structured Gaussian Processes (kiss-gp). In *International Conference on Machine Learning*, pages 1775–1784. PMLR, 2015.

Andrew Gordon Wilson, David A Knowles, and Zoubin Ghahramani. Gaussian Process Regression Networks. In *International Conference on Machine Learning*, 2011a.

Andrew Gordon Wilson, David A Knowles, and Zoubin Ghahramani. Gaussian Process Regression Networks. In *International Conference on Machine Learning*, 2011b.

James Wilson, Viacheslav Borovitskiy, Alexander Terenin, Peter Mostowsky, and Marc Deisenroth. Efficiently Sampling Functions from Gaussian Process Posteriors. In *International Conference on Machine Learning*, pages 10292–10302. PMLR, 2020.

Sewall Wright. The method of path coefficients. *The annals of mathematical statistics*, 5(3):161–215, 1934.

Jian Wu, Matthias Poloczek, Andrew Gordon Wilson, and Peter I Frazier. Bayesian optimization with gradients. *arXiv preprint arXiv:1703.04389*, 2017.

Qingyun Wu, Naveen Iyer, and Hongning Wang. Learning Contextual Bandits in a Non-stationary Environment. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 495–504, 2018.

Guillaume Wunsch, Federica Russo, Michel Mouchart, and Renzo Orsi. Time and Causality in the Social Sciences. Working papers, Dipartimento Scienze Economiche, Universita' di Bologna, September 2020.

Xin-Qiu Yao, Huaiqiu Zhu, and Zhen-Su She. A dynamic Bayesian network approach to protein secondary structure prediction. *BMC bioinformatics*, 9 (1):1–13, 2008.

Cong Ye, Liam Butler, C Bartek, Marat Iangurazov, Qiuchen Lu, Alastair Gregory, Mark Girolami, and Campbell Middleton. A Digital Twin of Bridges for Structural Health Monitoring. In *12th International Workshop on Structural Health Monitoring 2019*. Stanford University, 2019.

Xubo Yue and Raed AL Kontar. Why Non-myopic Bayesian Optimization is Promising and How Far Should We Look-ahead? A Study via Rollout. In *Artificial Intelligence and Statistics*, pages 2808–2818. PMLR, 2020.

Seniha Esen Yuksel, Joseph N Wilson, and Paul D Gader. Twenty years of mixture of experts. *IEEE transactions on neural networks and learning systems*, 23(8):1177–1193, 2012.

Junzhe Zhang. Designing Optimal Dynamic Treatment Regimes: A Causal Reinforcement Learning Approach. In *International Conference on Machine Learning*, pages 11012–11022. PMLR, 2020.

Junzhe Zhang and Elias Bareinboim. Markov Decision Processes with Unobserved Confounders: A Causal Approach. Technical report, Technical Report R-23, Purdue AI Lab, 2016.

Junzhe Zhang and Elias Bareinboim. Near-Optimal Reinforcement Learning in Dynamic Treatment Regimes. In *Neural Information Processing Systems*, 2019a.

Junzhe Zhang and Elias Bareinboim. Near-optimal reinforcement learning in dynamic treatment regimes. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019b.

Kun Zhang, Bernhard Schölkopf, Krikamol Muandet, and Zhikun Wang. Domain Adaptation under Target and Conditional Shift. In *International Conference on Machine Learning*, pages 819–827, 2013.

Yu Zhang and Qiang Yang. A Survey on Multi-Task Learning. *IEEE Transactions on Knowledge and Data Engineering*, 2021.

Xun Zheng, Bryon Aragam, Pradeep Ravikumar, and Eric P Xing. DAGs with NO TEARS: Continuous Optimization for Structure Learning. In *Neural Information Processing Systems*. PMLR, 2018.

Min Zou and Suzanne D Conzen. A new dynamic bayesian network (DBN) approach for identifying gene regulatory networks from time course microarray data. *Bioinformatics*, 21(1):71–79, 2005.

Mauricio A. Álvarez, Lorenzo Rosasco, and Neil D. Lawrence. Kernels for Vector-Valued Functions: A Review. *Foundations and Trends® in Machine Learning*, 4(3):195–266, 2012. ISSN 1935-8237. doi: 10.1561/2200000036.