Systems for Personalization of Hearing Instruments

A Machine Learning Approach

Nielsen, Jens Brehm; Larsen, Jan; Nielsen, Jakob

Publication date:
2015

Document Version
Publisher's PDF, also known as Version of record

Systems for Personalization of Hearing Instruments

A Machine Learning Approach

Jens Brehm Nielsen

Kongens Lyngby 2014
PHD-2014-325

Technical University of Denmark
Applied Mathematics and Computer Science
Building 303B, DK-2800 Kongens Lyngby, Denmark
Phone +45 45253031, Fax +45 45881399
reception@compute.dtu.dk
www.compute.dtu.dk

PHD: ISSN 0909-3192

Summary (English)

Today, modern digital devices can be customized significantly to the individual user by adjusting or optimizing multiple parameters affecting the output of the devices. Such personal optimization of devices is referred to as personalization. In the case of hearing aids, personalization is not only a possibility offered to the user, but a requirement that must be performed carefully and precisely in order for the user to utilize the full potential of modern multi-parameter hearing aids. Today though, personalization is still based on a manual time-consuming trial-and-error approach performed by the user himself or, in case of hearing aids, by a hearing-care professional based on typically ambiguous oral feedback from the user. This often results in sub-optimal or even inappropriate settings of multi-parameter devices. This dissertation presents research on a machine-learning based interactive personalization system to improve the personalization of devices and, in particular, of hearing-aid devices. The proposed personalization system iteratively learns a non-parametric probabilistic model of a user's assumed internal response function over all possible settings of a multi-parameter device based directly on sequential perceptual feedback from the user. A sequential design based on active learning is used to obtain the maximum of the user's unknown internal response function in as few iterations as possible. Experiments were conducted where the proposed personalization system obtained a significantly preferred setting for individual users within ten to twenty iterations in scenarios with up to four parameters.

Following a short introduction that includes a summary of results and contributions, the first main chapter focuses on the probabilistic modeling framework in which a Gaussian process is used to model the user's unobserved internal response function. The first main challenge addressed in this context is to account

for inconsistent and thus noisy user feedback. The second main challenge addressed is to support feedback which closely reflects the user's perception while providing maximal information about it without imposing a high cognitive load. In the second main chapter, active learning and sequential design are discussed in relation to the challenge of obtaining the setting that maximizes the user's unobserved internal response function in as few iterations as possible. For the Gaussian process framework, an active learning criterion is proposed specifically suitable for this type of optimization. The final chapter contains an overall discussion and conclusion of the present work and research based in part on the results from eight scientific paper contributions contained in the appendices.

Resumé (Danish)

Nutidens digitale apparater kan skræddersys betydeligt til den enkelte bruger ved justering eller optimering af en række parametre, der påvirker apparatets output. Personalisering referer til sådan en form for personlig optimering. For høreapparater er personalisering ikke kun et tilbud til brugeren, men en nødvendighed, hvis brugeren skal opnå det fulde udbytte af nutidens høreapparater indeholdende flere parametre. I dag bliver personalisering stadig baseret på manuelt at prøve sig frem, hvilket er tidskrævende. Det bliver gjort af brugeren selv, eller hvad angår høreapparater, af en professionel høreapparatsspecialist baseret på typisk uklar mundtlig feedback fra brugeren. Dette resulterer i ofte ikke optimale eller sågar uhensigtsmæssige apparatindstillinger. Denne afhandling præsenterer forskning omkring et *machine-learning*-baseret personaliseringssystem til at forbedre personaliseringen af apparater specielt med henblik på høreapparater. Det foreslåede personaliseringssystem lærer iterativt en ikke-parametrisk probabilistisk model af en brugers (antaget) interne responsfunktion over mulige parameterindstillinger baseret direkte på perceptuel feedback fra brugeren. Et sekventielt design baseret på *active learning* bruges for i så få iterationer som muligt at lære hvilken indstilling, der maksimerer brugerens interne responsfunktion. I udførte eksperimenter lærte det foreslåede personaliseringssystem en signifikant foretrukket indstilling for individuelle brugere indenfor ti til tyve iterationer i scenarier med op til fire parametre.

Efter en kort introduktion, der inkluderer en oversigt over resultater og forskningsbidrag, fokuserer det første hovedkapitel på den probabilistiske modelleringsmetode, hvor en Gaussisk proces bruges til modellering af brugerens ikke observerede interne responsfunktion. De adresserede hovedudfordringer er i denne kontekst at tage højde for inkonsistent og dermed støjfyldt brugerfeedback og

at supportere feedback, som nøje reflekterer brugerens perception uden dog at resultere i en høj kognitiv belastning. I det andet hovedkapitel bliver active learning og sekventielt design diskuteret i relation til udfordringen i at lære i så få iterationer som mulig den indstilling, der maksimerer brugerens ikke observerede interne responsfunktion. I relation til en Gaussisk proces foreslås et active learning kriterium, som er specifikt velegnet til den omtalte form for optimering. Det sidste kapitel indeholder en overordnet diskussion om det her omtale stykke arbejde og forskning baseret til dels på de otte videnskabelige artikler, som er at finde i appendikserne. Det sidste kapitel indeholder også konklusionen.

Preface

This dissertation was prepared at the Department of Applied Mathematics and Computer Science (former Department of Informatics and Mathematical Modelling) at the Technical University of Denmark in partial fulfillment of the requirements for acquiring the Ph.D. degree in engineering. The work has been done in collaboration with Widex A/S as an industrial Ph.D. project. Supervisor was Associate Professor Jan Larsen from the Technical University of Denmark and co-supervisor was R&D Engineer Jakob Nielsen from Widex A/S.

The dissertation consists of four chapters that summarize the work and a collection of seven published scientific papers and one paper currently under review. The work was carried out between January 2011 and January 2014.

Lyngby, January 17, 2014

Jens Brehm Nielsen

List of Publications

Papers included in the thesis

[A] Bjørn Sand Jensen, Jens Brehm Nielsen, Jan Larsen. Efficient Preference Learning with Pairwise Continuous Observations and Gaussian Processes. *2011 IEEE International Workshop on Machine Learning for Signal Processing*, 2011. Published.

[B] Jens Brehm Nielsen, Bjørn Sand Jensen, Jan Larsen. On Sparse Multi-Task Gaussian Process Priors for Music Preference Learning. *NIPS 2011 workshop on Choice Models and Preference Learning*, 2011. Published.

[C] Jens Brehm Nielsen, Bjørn Sand Jensen, Jan Larsen. Pseudo Inputs for Pairwise Learning with Gaussian Processes. *2012 IEEE International Workshop on Machine Learning for Signal Processing*, 2012. Published.

[D] Jens Brehm Nielsen, Jakob Nielsen. Efficient Individualization of Hearing Aid Processed Sound. *2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013. Published.

[E] Bjørn Sand Jensen, Jens Brehm Nielsen, Jan Larsen. Bounded Gaussian Process Regression. *2013 IEEE International Workshop on Machine Learning for Signal Processing*, 2013. Published.

[F] Jens Brehm Nielsen, Bjørn Sand Jensen, Toke Jansen Hansen, Jan Larsen. Personalized Audio Systems - A Bayesian Approach. *AES 135th Convention*, 2013. Published.

[G] Jens Brehm Nielsen, Bjørn Sand Jensen, Jakob Nielsen, Jan Larsen. Hearing Aid Personalization. *NIPS 2013 workshop on Personalization*, 2013. Published.

[H] Jens Brehm Nielsen, Jakob Nielsen, Jan Larsen. Perception based Personalization of Hearing Aids using Gaussian Processes and Active Learning. *IEEE Transaction on Audio, Speech, and Language Processing*. Submitted.

Papers not included in the thesis

- Jens Madsen, Jens Brehm Nielsen, Bjørn Sand Jensen, Jan Larsen. Modeling Expressed Emotions in Music using Pairwise Comparisons. *9th International Symposium on Computer Music Modelling and Retrieval*, 2012. Published.
- Jens Madsen, Bjørn Sand Jensen, Jan Larsen, Jens Brehm Nielsen. Towards Predicting Expressed Emotion in Music from Pairwise Comparisons. *Proceedings of the 9th Sound and Music Computing Conference*, 2012. Published.

Acknowledgements

Several people have contributed to make the work contained in this dissertation possible. First of all, I would like to thank my university supervisor associated professor Jan Larsen for advices, interesting discussions and feedback during the work, but not least his encouragement and support during the past years. Similarly, I would like to express my gratitude to my supervisor at Widex A/S R&D Engineer Jakob Nielsen who has always been willing to discuss matters related to the present work and to supply extensive help and introduction whenever needed in relation to various hearing-aid aspects. Jakob has also been a very good friend and associate at Widex A/S. A thank also goes to past and present members of the Cognitive Systems group and to my colleagues at Widex A/S for pleasant and friendly environments, but also for valuable feedback and discussions during the work. A very special thank goes to Post. Doc. Bjørn Sand Jensen, with whom I have had the pleasure to work with at several occasions. His inputs and expertise have been invaluable.

I am pleased with the financial support by the Ministry of Science, Innovation and Higher Education and by Widex A/S.

Nomenclature

Abbreviations

Abreviation	Explanation
HA(s)	**h**earing **a**id(s).
HCP	**h**earing-**c**are **p**rofessional.
IRF	**i**nternal **r**esponse **f**unction; refers to the assumed unobserved process related to a user's perception of stimuli.
MUSHRA	**mu**ltiple **s**timuli with **h**idden **r**eferences and **a**nchors; perceptual measurement paradigm.
i.i.d.	**i**ndependent and **i**dentically **d**istributed.
psd	**p**ositive **s**emi**d**efinite.
GP	**G**aussian **p**rocess.
ML	**m**arginal **l**ikelihood.
ML-II	**m**arginal-**l**ikelihood-**II**; also called the evidence, used to optimize hyperparameters, i.e. covariance and likelihood parameters, of a Gaussian process.
MAP-II	**M**aximum-**A**-**P**osterior-**II**; used to optimize hyperparameters, i.e. covariance and likelihood parameters, of a Gaussian process regularized by hyperpriors.
SE	**S**quared **E**xponential; a SE kernel is a common covariance or kernel function.
ISO	**iso**tropic.

ARD	**a**utomaic **r**elevance **d**etermination.
PP	**p**robability **p**roduct; a PP kernel is a special covariance or kernel function.
MT	**m**ulti **t**ask; MT learning.
CF	**c**ollaborative **f**iltering.
EP	**E**xpectation **P**ropagation; approximate inference method.
VB	**V**ariational **B**ayes; approximate inference method.
FI(T)C	**f**ully **i**ndependent (**t**raining) **c**onditional.
PI(T)C	**p**artially **i**ndependent (**t**raining) **c**onditional.
TG	**t**runcated **G**aussian; a TG distribution.
GP-WA	a warped GP model.
GP-TG	a GP model with the TG likelihood function.
GP-BE	a GP model with the Beta likelihood function.
2AFC	**two-a**lternative **f**orced-**c**hoice; perceptual measurement paradigm.
SPGP	sparse pseudo-input GP.
PJ	**p**airwise **j**udgment; a PJ kernel is a common covariance or kernel function for pairwise observations.
EI	**E**xpected **I**mprovement; an active-learning criterion.
uEI	**u**ni-variate **E**xpected **I**mprovement; original EI criterion.
bEI	**b**i-variate **E**xpected **I**mprovement; extended EI criterion.
UCB	**u**pper **c**onfidence **b**ound; an active-learning criterion.

Symbols and Notation

Symbol	Explanation
$\mathbb{R}$	real numbers.
a	a scalar.
$\mathbf{a}$	a vector.
a_i	i'th component of vector $\mathbf{a}$.
$\mathbf{a}_i$	a unique vector.
$\mathbf{A}$	a matrix.
$[\mathbf{A}]_{i,j}$ or $A_{i,i}$	the element of the matrix $\mathbf{A}$ that is in the i'th row and j'th column.

$[\mathbf{A}]_i$	the entire i'th row of the matrix $\mathbf{A}$.
$\mathbf{A}_i$	a unique matrix.
$\mathbf{a}^\top$, $\mathbf{A}^\top$	transpose of vector and matrix, respectively.
$\mathbf{A}^{-1}$	the inverse of a matrix.
$\mathbf{I}_{n\times n}$	n-by-n identity matrix.
diag($\mathbf{a}$)	diagonal matrix, where the diagonal elements are given by the vector, $\mathbf{a}$.
$\mathbf{x}$	a D-dimensional input, i.e., $\mathbf{x} \in \mathbb{R}^D$.
$\mathcal{X}$	set of n inputs, $\mathbf{x}_i$, i.e., $\mathcal{X} = \{\mathbf{x}_i \vert i = 1, ..., n\}$.
$\bar{\mathbf{x}}$	a D-dimensional pseudo input, i.e., $\bar{\mathbf{x}} \in \mathbb{R}^D$.
$\bar{\mathbf{X}}$	matrix containing the $\bar{n}$ pseudo inputs, i.e., $\bar{\mathbf{X}} = [\bar{\mathbf{x}}_1, ..., \bar{\mathbf{x}}_{\bar{n}}]^\top$.
$k(\mathbf{x}, \mathbf{x}')$	covariance or kernel function. Typically, covariance function is used.
$\mathbf{K}$	covariance matrix, where $[\mathbf{K}]_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j)$.
y	an observation.
$\mathbf{y}$	vector of m observations, i.e., $\mathbf{y} = [y_1, ..., y_m]^\top$.
$\mathcal{Y}$	set of pairwise observations between any two instances, u_k and v_k, i.e., $\mathcal{Y} = \{y_k \vert u_k, v_k, k = 1, ..., m\}$.
$\mathcal{GP}$	a Gaussian process.
$f(\mathbf{x})$	a function. In general, $f(\mathbf{x})$ is considered a latent representation. For the specific applications described in the present work, $f(\mathbf{x})$ models individual user's IRF
$\mathbf{f}$	vector of n function values of the function $f(\mathbf{x})$, i.e., $\mathbf{f} = [f(\mathbf{x}_1), ..., f(\mathbf{x}_n)]^\top$.
$\mathbf{f}_k$	function value(s) associated with the k'th likelihood function.
* or $_*$	the star is used rather loosely, either as a sub or super script, to indicate vectors and matrices associated with predictions.
$\boldsymbol{\theta}_\mathcal{L}$	set of likelihood parameters.
$p(\mathbf{z})$	distribution of the stochastic variable, $\mathbf{z}$.
$p(\mathbf{z}\vert\cdot)$	conditional distribution of the stochastic variable, $\mathbf{z}$.
$p(y_k\vert\mathbf{f}_k, \boldsymbol{\theta}_\mathcal{L})$, $p(y_k\vert\mathbf{f}_k)$	likelihood function with or without explicitly indicating the likelihood parameters, $\boldsymbol{\theta}_\mathcal{L}$.
$p(\mathbf{y}\vert\mathbf{f})$	likelihood.
$p(\mathbf{f})$	prior distribution.
$p(\mathbf{f}\vert\mathbf{y})$	posterior distribution.

$\boldsymbol{\mu}$	mean vector.
$\boldsymbol{\Sigma}$	covariance matrix.
$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, $\mathcal{N}(\mathbf{z}\|\boldsymbol{\mu}, \boldsymbol{\Sigma})$	(multi-variate) Gaussian distribution with mean, $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$. Hence, $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}\|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ or alternatively, $\mathbf{z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$
$\Phi(z)$	cumulative density function of the standard Gaussian distribution with zero mean and unity variance.
$\text{Beta}(z\|\alpha, \beta)$	beta distribution with shape parameters, α and β.
$B(\alpha, \beta)$	beta function.
$\mathcal{O}(\cdot)$	computational cost.

Contents

Chapter 1

Introduction

The opening chapter contains a general introduction to the present thesis. The chapter includes a background in Section 1.1 and a motivation in Section 1.2 with a non-technical conceptual description of the proposed personalization system. In Section 1.3, previous work related to the described personalization system is shortly summarized. Finally, a summary of research contributions is contained in Section 1.4.

1.1 Background

The capabilities of modern digital consumer devices, such as smartTVs, smartphones and professional audio equipment, are constantly becoming increasingly advanced and customizable in terms of the number of tunable parameters and thus possible settings. Consequently, the devices offer the user an increased level of *personalization*. For some devices, users may be able to obtain an optimal or possibly suboptimal setting based on the user's own perception by simple manual trial-and-error procedures. In general though, the increasing flexibility encourages methods that optimize or *personalize* the setting of device parameters more intelligently and systematically.

Digital hearing aids (HAs) are medical devices, which do not only *offer* a high level of personalization, but actually *require* that they are carefully personalized. HA personalization—more commonly referred to as HA fitting and fine-tuning—is today carried out using pre-defined rules named *prescriptions* [Dillon, 2012, Chapter 10] followed by up to several fine-tuning attempts [Dillon, 2012, Chapter 11-12]. A prescription defines rules for gain and compression settings of HAs based on the user's absolute pure-tone hearing thresholds measured at different frequencies. These thresholds are collected in what is referred to as an *audiogram*. The deterministic prescriptions are developed based on decades of research studying the human auditory system and on empirical and practical experience. Generally, the purpose of any prescription is to recover speech that is inaudible to the user. However, the prescriptive rules are based on the idea that given the audiogram *one size fits all*. In practice, HAs are fitted initially in the clinic by a hearing-care professional (HCP) based on a particular prescription with other HA features—such as noise reduction, microphone configurations etc.—set to reasonable default values. Often, fine-tuning is immediately needed to accommodate whatever the user may find inappropriate with the prescribed gain and compression settings and/or with the default feature settings. After a period of time in which the user during daily routines has become aware of potential problems with the initial fitting, the user is scheduled to come back to the clinic for further fine-tuning to address any problems. During fine tuning the HCP manually tries to adjust the setting of the HAs including gain and compressor settings as well as settings of available features based on the user's descriptions of the problem(s). Because the average HA user is seldom used to describe the perception of sound, the descriptions of problems are typically ambiguous and incomplete.

A persistent problem with HAs is that more than 10% of them end up "in the drawer" [Kochkin et al., 2010], meaning that after purchase the user never wears the HAs, because they do not meet expectations. Kochkin et al. [2010] argue this is because the protocol for fitting HAs is in many occasions not being fully

respected in the clinic. Particularly, HA fine-tuning and validation of achieved performance and benefit are being blamed for the high amount of "drawer" HAs. Hence, a key to successfully operating HAs is to have them carefully personalized to the individual *beyond* the prescription and default feature settings.

During the fine-tuning procedure, the user's perception is translated twice for every HA adjustment. The first translation is the often ambiguous explanation by the user about what sounds inappropriate. The second translation occurs when the HCP seek to understand what problems the user seek to describe. Thereby, a lot can get lost in translation even before the HCP may consider what the problem is with the current HA setting. Subsequently, the HCP can only resort to an *knowledge-based*[1] trial and error approach in order to iterate towards successfully personalized HAs. The above assumes that the user actually reports that there is a problem with the current setting. Because HAs are medical devices which are recommended and personalized by a professional, especially first time HA users are not aware that constant and valuable feedback from them is key in order to obtain good HA performance and benefit. Consequently, some users—in particular, first time users— initially accept an imperfect setting without mentioning that something might not live up to their expectations. The reason these users do not report any potential issues is simply because they believe that a professional knows how the HA should sound and thus knows what is best for the user. The truth is however that nobody knows exactly how the HA sound is perceived individually, except the individual user.

In summary, at least two problems persist with the existing personalization routine. First of all, the vast number of adjustable parameters embedded in both modern consumer devices and in digital HAs makes manual trial-and-error approaches extremely difficult to keep track of in practice. Secondly for the case of HA personalization and other related areas, the entire feedback link from the user through the HCP to the adjustment made in the HAs can be very imperfect. Although the exact proportion is unknown, a considerable amount of users wear HAs (or have them in the drawer) which are not carefully personalized, and consequently do not benefit fully from the technological leap over the last couple of decades. To close the gap between technology and benefit, a leap in the way devices are personalized in the future must be made to keep up with the advancing technology in modern digital devices.

[1] The use of *trial-and-error* as a reference to how hearing aids are being fine-tuned today, is by no means intended to discredit the HCP. Therefore, *knowledge-based* trial-and-error should be read throughout this thesis to highlight that each *trial* is a qualified-guess by a professional with years of audiological experience, training and expertize.

1.2 Motivation

Today, devices are highly personalizable, but the personalization approach used in practice arguably falls short, because a manual trial-and-error approach, and in case of HA fine-tuning also the oral feedback link from the user through the experienced professional to the HA adjustment, are insufficient. Hence, the fine-tuning and thus the personalization of devices could be improved, significantly, with more intelligent and structured optimization techniques based directly on the users perception, and not on an imperfect oral translation thereof.[2]

The hypothesis is that the personalization of devices—in particular HAs—can be improved significantly by the use of *interactive* optimization techniques based on probabilistic machine learning. The user must be provided with a simple user interface, where individual device settings are assessed in close resemblance with the user's perception, and possible inconsistent and thus noisy user feedback must be modeled carefully. Furthermore, the number of assessments required to obtain an optimal setting could be reduced if a model of the user's unobserved *internal response function* (IRF) over possible device settings is used to propose settings for user assessment actively. Hence in short, a robust and fast *personalization system* should sequentially update a model of the user's IRF over devices settings based on a current set of assessments and then actively query a new setting to be assessed based on the updated model. At the end, the suggested optimal setting is the one that maximizes the predicted IRF of the user. A conceptual illustration of the considered system is sketched in Fig. 1.1.

Such a system however imposes some key challenges from a machine learning perspective. First of all, user feedback is essentially very noisy and subject to various bias effects [Bech and Zacharov, 2007], but might also be exhausting for users to provide. For this reason, absolute ratings should be used with care and generally only when suitable anchors and/or reference examples can be defined. For audio evaluation, one of the most commonly used standards for absolute ratings—the ITU-R BS.1534-1 [2003], more commonly known as the MUSHRA test [Bech and Zacharov, 2007, Sec. 10.1.3]—is relying on suitable reference and anchor stimuli in order to obtain consistent absolute ratings. Pairwise, or more generally, relative assessments are arguably more appropriate when dealing with human perception as pointed out by Lockhead [2004]. Lockhead

[2] Since users will be guided through different device settings while being forced to decide which settings they prefer over others, a positive side effect is that users will to a greater extend explore and recognize their own individual preference. Thereby, users will obtain greater psychological ownership through active engagement, which is for instance known to results in a better outcome of the entire hearing-impairment therapy [Dillon et al., 2006, Convery et al., 2011].

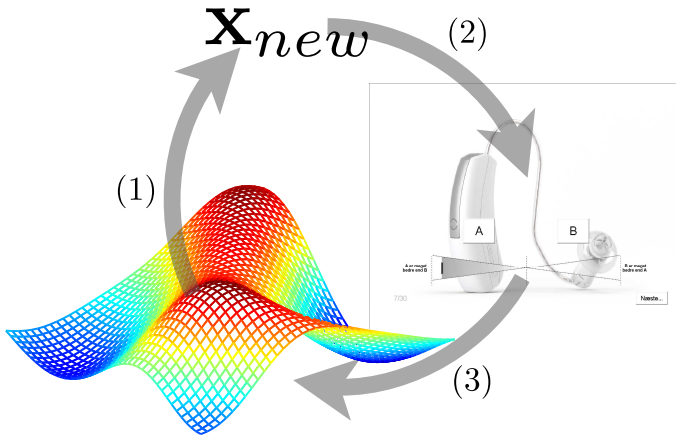


Figure 1.1: Conceptual sketch of the considered personalization framework. The framework utilizes loosely speaking an *interactive loop*, which consists of three parts. (1) Based on an estimate of a user's IRF, active learning is used to suggest settings—one or several—that the user should assess. (2) The user assesses the new setting(s) by evaluation. The assessment is given as feedback to the model of the user's IRF. (3) The estimate of the user's IRF is updated based on the past assessments including the new one.

[2004] argues that effectively all absolute ratings including Weber's law [Weber, 1965]—more commonly known as the *just noticeable difference*—are relative. A fundamental benefit of pairwise assessments is that they are typically easier and more intuitive for users to perform consistently. The reason for this is that a pairwise assessment is essentially simpler than an absolute assessment, because a pairwise assessment is inherently relative, and is thus following Lockhead [2004] in closer resemblance with human perception. In any case, it is in general not correct to assume the observational noise to be i.i.d Gaussian noise, when assessments are provided as either an absolute and bounded rating or a pairwise comparison. Hence one challenge from a machine-learning perspective is to support and possibly develop noise models for these kinds of noise types for the personalization system considered in the present work.

A second challenge is that the complexity of the user's IRF are unknown and will most certainly vary across applications, contexts and subjects. Therefore, a non-parametric approach towards modeling of the user's IRF is considered

in the present work, in which the functional complexity is not strictly specified beforehand. This however requires that the non-parametric model of the user's IRF can be trained robustly to avoid over fitting. A Gaussian process [Rasmussen and Williams, 2006] constitutes an appealing Bayesian non-parametric regression framework for this purpose, which is specifically considered in the present work to model the user's unobserved IRF. Due to the special nature of the observations, traditional Gaussian process regression is not directly applicable. Hence, the specially developed choice models must be adopted to the Gaussian process framework.

The final key challenge from a machine learning perspective is to apply active learning for (ideally global) optimization of the user's IRF modeled in the Gaussian process framework.

1.3 Related Work

The problem of personalizing systems and in particular system parameters have been studied considerably over the last two to three decades. For HA personalization, the earliest attempts use a *modified simplex procedure* [Neuman et al., 1987, Kuk and Pape, 1992] and later *Genetic algorithms* [Takagi and Ohsaki, 1999, Durant et al., 2004, Baskent et al., 2007] to optimize parameters based on responses from users. Both methods however require an unrealistic number of user interactions to convergence even for few system parameters.

The first to conceptualize an interactive machine-learning based approach for personalization of device parameters driven by forced-choice pairwise-comparison user feedback were Heskes and de Vries [2005]. They rely on a specific parametrized functional form of the user's IRF, which they refer to as the user's *utility function*. They assume this functional form to be known beforehand. In practice, this is an assumption that is difficult to satisfy. This makes the framework by Heskes and de Vries [2005] less applicable. Birlutiu et al. [2009, 2010] have proposed an approach for preference learning which is closer related to the present work also relying on non-parametric Gaussian processes, although their approach is not sequential. Instead, their framework use a multi-task formulation for preference learning, where the IRFs from like-minded users are able to transfer information between them to boost (or regularize) the learning of a single user's IRF. Groot et al. [2011] consider Gaussian processes for preference learning without the multi-task formulation on real-world data. Neither the approach by Birlutiu et al. [2009, 2010] nor the approach by Groot et al. [2011] are currently an optimization technique. Groot et al. [2010] consider Gaussian processes for optimization of some parameters (the amount of different ingredi-

ents in a cake mix) given that other parameters are uncontrolled (for instance the baking temperature). This framework however considers only traditional regression and thus not feedback from pairwise comparisons, which is crucial for perceptual evaluation. Another machine learning direction was proposed by Reed [2001], where a nearest neighbor approach was used to assist a user optimizing an equalization system. Also for optimal equalization, Sabin and Pardo [2008], Pardo et al. [2012] propose a system which linearly correlates a particular user's concept of for instance a "warm sound" with a specific equalization setting.

1.4 Outline and Contributions

In addition to this introductory chapter, the present thesis contains two main chapters, a chapter containing a discussion and a conclusion, and finally the eight paper contributions in the appendix. The first main chapter serves as a general introduction to the developed Gaussian process framework including the proposed noise models needed to support the special type of observations considered in the present work. The second main chapter describes the sequential design approach including the active learning criterion found to perform well for the problems considered in the present work.

Chapter 2 - Gaussian Processes: Provides an introduction to the Gaussian process framework applied in the paper contributions. The chapter leverages standard Gaussian process regression in an attempt to provide the reader with the foundation to use Gaussian process regression with non-traditional likelihood functions. In particular, the chapter reviews the non-standard likelihood functions developed in the present work for both absolute assessments and pairwise judgments. Additionally, the chapter throughout contains various details and references to interesting literature.

Chapter 3 - Active Learning: Describes active learning and sequential design heuristics in relation to the present work. Specifically, the focus is on *Expected Improvement* as a global optimization criterion and on a bivariate version thereof, which fully exploits the Gaussian process predictions.

Chapter 4 - Discussion and Conclusion: Contains discussions of further machine-learning research relevant for the present work, and real-world application perspectives of the present work. This chapter also contains the conclusion.

Paper A - Efficient Preference Learning with Pairwise Continuous Observations and Gaussian Processes: This paper proposes a novel pairwise likelihood function for the Gaussian process framework. The likelihood supports observations that encode both the pairwise decision of selecting one of two options and the *degree* to which the selected option is better, preferred or likewise than the other option. On an artificial example, it is demonstrated that a Gaussian process model can be learned using fewer observations of the type supported be the novel likelihood than with what is required with the state-of-the-art binary pairwise likelihood function by Chu and Ghahramani [2005a]—also under adverse noise conditions.

Paper B, C - On Sparse Multi-Task Gaussian Process Priors for Music Preference Learning, Pseudo Inputs for Pairwise Learning with Gaussian Processes: In these two papers, the pseudo-input formulation for sparse Gaussian processes is applied for the binary pairwise Gaussian process model by Chu and Ghahramani [2005a]. The papers, thus propose a sparse version of the binary pairwise likelihood function by Chu and Ghahramani [2005a]. The resulting sparse version scales $\mathcal{O}(\bar{n}^2 n)$ instead of $\mathcal{O}(n^3)$ for the standard version, where n is the number of observational inputs and $\bar{n} << n$ is the number of pseudo inputs.

Paper D - Efficient Individualization of Hearing Aid Processed Sound: For the case of personalizing hearing aids to the individual, the machine-learning approach described in the present work is examined in a real-world scenario. An experiment has been conducted where five hearing-aid users personalize two hearing-aid parameters controlling how the hearing aids process speech in the presence of background noise. Although the differences between settings of the two parameters are extremely subtle, the predictions of individual user's IRF between test and retest indicate somewhat robust reproducible and stable estimates of the obtained preferred setting for four out of the five subjects.

Paper E - Bounded Gaussian Process Regression: Two novel likelihood functions are proposed for the Gaussian process framework which model the observational noise of an absolute bounded response explicitly in the observational space. This is in contrast to the approach proposed by Snelson et al. [2004], where the bounded observations are *warped* to an unbounded space, such that the noise is modeled implicitly by a standard Gaussian process regression model. The noise-modeling abilities of the different approaches are compared on two real-would examples where several subjects have rated different signal processing strategies of a compressor. The comparison favors the novel likelihood functions slightly. Code for the two new likelihood functions is available for the matlab-gpml toolbox by Rasmussen and Nickisch [2010].

Paper F - Personalized Audio Systems - A Bayesian Approach: A personalization system using absolute assessments in a MUSHRA like paradigm is proposed for personalization of multi-parameter audio systems. A five-band equalizer is used as an example of a multi-parameter audio system. An experiment with test subjects has been conducted, which shows a significant benefit of the active learning/sequential design approach. Moreover, a special structure in the *kernel* is used specifically for audio applications, which exploits correlation between "adjacent" system parameters. The latter though is seen to decrease performance when used in the specific active-learning setup.

Paper G, H - Hearing Aid Personalization, Perception based Personalization of Hearing Aids using Gaussian Processes and Active Learning: The machine-learning personalization system proposed in the present work is described in detail and used to obtain preferred settings in two real-world HA experiments. In the two experiments, preferred settings of two and four parameters, respectively, are obtained in a set of hearing aids for a group of HA users in a music context. Only results from the last experiment are reported in Paper G. Generally, the results show a significant ($p_0 < 0.05$) preference for the setting obtained with the system when compared to the prescription. Based on test/retest results, it is demonstrated that the system is capable of reproducing the obtained settings for individual users within qualitatively reasonable limits, although both learning and fatigue effects come into play regarding user consistency.

Chapter 2

Gaussian Processes

Gaussian processes are priors over entire functions and are thus natural priors in Bayesian non-parametric regression frameworks. A (zero-mean) Gaussian process is defined only through a covariance (or kernel) function, which makes Gaussian processes very flexible. The Bayesian formulation makes it possible to learn kernel parameters from data in a principle manner. On the downside, the computational cost of a Gaussian process scales cubic in the number of training examples.

In the present work, only few training examples are available, hence the computational scaling is not a problem. Instead, the Bayesian non-parametric formulation makes it possible to consider a rich class of functions, with a principle way to restrict the solution based on the limited amount of training examples.

2.1 Introduction to Gaussian Processes

Formally, a Gaussian process (GP) *is a collection of random variables, any finite number of which have a joint Gaussian distribution* [Rasmussen and Williams, 2006, Def. 2.1]. A GP is completely defined by a mean function, $m(\mathbf{x})$, and a positive semidefinite (psd) covariance or kernel function, $k(\mathbf{x}, \mathbf{x}')$, written as

$$f(\mathbf{x}) \sim \mathcal{GP}\left(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')\right) \tag{2.1}$$

indicating that the function $f : \mathbb{R}^D \to \mathbb{R}, \mathbf{x} \mapsto f(\mathbf{x})$ is modeled by a GP. In the remainder of this thesis, only zero-mean GPs are considered, hence $m(\mathbf{x}) = 0$ will be implicit, but further details of non zero-mean GPs are given by Rasmussen and Williams [2006]. For a *finite* set of D-dimensional inputs, $\mathcal{X} = \{\mathbf{x}_i | i = 1, ..., n\}$, and corresponding function values, $\mathbf{f} = [f(\mathbf{x}_1), ..., f(\mathbf{x}_n)]^\top$, the prior over $\mathbf{f}$ is defined by the GP as a multivariate Gaussian distribution given by

$$\mathbf{f}|\mathcal{X} \sim \mathcal{N}\left(0, \mathbf{K}\right), \tag{2.2}$$

where the elements in the kernel matrix, $\mathbf{K}$, are given as $[\mathbf{K}]_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j)$. By far the simplest and most elegant way to derive a traditional GP for regression in the presence of i.i.d. Gaussian noise, ϵ, with variance σ^2, is to proceed as Rasmussen and Williams [2006]. Given the GP, the joint distribution between the noisy targets, $\mathbf{y} = [y_1, ..., y_n]$, where $y_i = f(\mathbf{x}_i) + \sigma$, and the function values, $\mathbf{f}^*$, for a new set of inputs, $\mathcal{X}^* = \{\mathbf{x}_l^* \in \mathbb{R}^D | l = 1, ..., n^*\}$, is given by

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}^* \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{K} + \sigma^2 \mathbf{I}_{n \times n} & \mathbf{K}_* \\ \mathbf{K}_*^\top & \mathbf{K}_{**} \end{bmatrix} \right), \tag{2.3}$$

where $[\mathbf{K}_{**}]_{l,r} = k(\mathbf{x}_l^*, \mathbf{x}_r^*)$ and $[\mathbf{K}_*]_{i,l} = k(\mathbf{x}_i, \mathbf{x}_l^*)$. By the use of Eq. 332 in Petersen and Pedersen [2008], the predictive distribution, $p(\mathbf{f}^*|\mathbf{y}, \mathcal{X}, \mathcal{X}^*)$, is directly available as

$$p(\mathbf{f}^*|\mathbf{y}, \mathcal{X}, \mathcal{X}^*) = \mathcal{N}\left(\mathbf{f}^*|\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*\right), \quad \text{with} \tag{2.4a}$$

$$\boldsymbol{\mu}^* = \mathbf{K}_*^\top \left(\mathbf{K} + \sigma^2 \mathbf{I}_{n \times n}\right)^{-1} \mathbf{y} \tag{2.4b}$$

$$\boldsymbol{\Sigma}^* = \mathbf{K}_{**} - \mathbf{K}_*^\top \left(\mathbf{K} + \sigma^2 \mathbf{I}_{n \times n}\right)^{-1} \mathbf{K}_* \tag{2.4c}$$

Despite the simplicity, this approach only shows little of what is needed to use GPs for more sophisticated (none-Gaussian) likelihoods. Another approach to derive the same fundamental equations from Eq. 2.4 is to define the Gaussian likelihood, $p(\mathbf{y}|\mathbf{f})$, explicitly as

$$p(\mathbf{y}|\mathbf{f}) = \prod_{k=1}^{n} \mathcal{N}\left(y_k | f_k, \sigma^2\right) = \mathcal{N}\left(\mathbf{y}|\mathbf{f}, \sigma^2 \mathbf{I}_{n \times n}\right) = \mathcal{N}\left(\mathbf{f}|\mathbf{y}, \sigma^2 \mathbf{I}_{n \times n}\right), \tag{2.5}$$

where the last equal sign comes from the fact that the random variable and mean of the Gaussian distribution can be interchanged. From Bayes formula, the posterior of $\mathbf{f}$ is given by

$$p(\mathbf{f}|\mathbf{y},\mathcal{X}) = \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathcal{X})}{p(\mathbf{y}|\mathcal{X})} \propto \mathcal{N}\left(\mathbf{f}|\mathbf{y},\sigma^2\mathbf{I}_{n\times n}\right)\mathcal{N}\left(\mathbf{f}|\mathbf{0},\mathbf{K}\right), \tag{2.6}$$

which up to the normalization constant, $p(\mathbf{y}|\mathcal{X})$, is seen just to be the product of two Gaussian distributions in $\mathbf{f}$. This product has a straight-forward solution listed in Eq. 348 in Petersen and Pedersen [2008], which also gives an expression for the important normalization constant, $p(\mathbf{y}|\mathcal{X})$, called the *marginal likelihood* (ML) or *evidence*. Thus, the posterior and marginal likelihood are directly available [Petersen and Pedersen, 2008, Eq. 348] as

$$p(\mathbf{f}|\mathbf{y},\mathcal{X}) = \mathcal{N}\left(\mathbf{f}|\boldsymbol{\mu},\boldsymbol{\Sigma}\right), \quad \text{with} \tag{2.7a}$$

$$\boldsymbol{\mu} = \mathbf{K}\left(\mathbf{K}+\sigma^2\mathbf{I}_{n\times n}\right)^{-1}\mathbf{y} \tag{2.7b}$$

$$\boldsymbol{\Sigma} = \left(\mathbf{K}^{-1}+\frac{1}{\sigma^2}\mathbf{I}_{n\times n}\right)^{-1} = \mathbf{K}\left(\mathbf{K}+\sigma^2\mathbf{I}_{n\times n}\right)^{-1}\sigma^2\mathbf{I}_{n\times n} \tag{2.7c}$$

$$p(\mathbf{y}|\mathcal{X}) = \mathcal{N}\left(\mathbf{y}|\mathbf{0},\mathbf{K}+\sigma^2\mathbf{I}_{n\times n}\right). \tag{2.7d}$$

Note, that the logarithm of Eq. 2.7d is identical to Eq. 2.30 in Rasmussen and Williams [2006] and is used for learning (or optimizing) covariance-function and likelihood parameters of the standard GP regression model. In the literature, this learning scheme is referred to as either evidence optimization or marginal-likelihood-II (ML-II) optimization. In the present work, weakly-informative hyper priors are typically used to regularize the marginal likelihood, $p(\mathbf{y}|\mathcal{X})$, to obtain a more robust Maximum-A-Posterior-II (MAP-II) scheme.

The predictive distribution can be derived following a general procedure which also applies for none-Gaussian likelihoods given that the posterior distribution of $\mathbf{f}$ is Gaussian. Except for the standard Gaussian likelihood, the posterior is rarely analytically tractable, in which case the Gaussian posterior is an approximation to the true posterior. In any case, the predictive distribution, $p(\mathbf{f}^*|\mathbf{y},\mathcal{X},\mathcal{X}^*)$, is given by

$$p(\mathbf{f}^*|\mathbf{y},\mathcal{X},\mathcal{X}^*) = \int p(\mathbf{f}^*|\mathbf{f},\mathcal{X},\mathcal{X}^*)p(\mathbf{f}|\mathbf{y},\mathcal{X})d\mathbf{f}, \tag{2.8}$$

where the last term is the—typically approximate, but for standard GP regression, exact—posterior, whereas the first term follows from the joint distribution given the GP between $\mathbf{f}$ and $\mathbf{f}^*$ as

$$\begin{bmatrix}\mathbf{f}\\ \mathbf{f}^*\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\end{bmatrix}, \begin{bmatrix}\mathbf{K} & \mathbf{K}_*\\ \mathbf{K}_*^\top & \mathbf{K}_{**}\end{bmatrix}\right), \tag{2.9}$$

resulting in [Petersen and Pedersen, 2008, Eq. 348]

$$p(\mathbf{f}^*|\mathbf{f},\mathcal{X},\mathcal{X}^*) = \mathcal{N}\left(\mathbf{f}^*|\mathbf{K}_*^\top\mathbf{K}^{-1}\mathbf{f}, \mathbf{K}_{**} - \mathbf{K}_*^\top\mathbf{K}^{-1}\mathbf{K}_*\right) \tag{2.10}$$

A solution to the marginalization over $\mathbf{f}$ in Eq. 2.8 can be found by insterting Eq. 2.7a and Eq. 2.10 and use Eq. 2.115 in Bishop [2006] to give

$$\begin{aligned}
p(\mathbf{f}^*|\mathbf{y},\mathcal{X},\mathcal{X}^*) &= \int p(\mathbf{f}^*|\mathbf{f},\mathcal{X},\mathcal{X}^*)p(\mathbf{f}|\mathbf{y},\mathcal{X})d\mathbf{f}, && (2.11)\\
&= \int \mathcal{N}\left(\mathbf{f}^*|\mathbf{K}_*^\top\mathbf{K}^{-1}\mathbf{f}, \mathbf{K}_{**} - \mathbf{K}_*^\top\mathbf{K}^{-1}\mathbf{K}_*\right)\mathcal{N}\left(\mathbf{f}|\boldsymbol{\mu},\boldsymbol{\Sigma}\right)d\mathbf{f} && (2.12)\\
&= \mathcal{N}\left(\mathbf{f}^*|\boldsymbol{\mu}^*,\boldsymbol{\Sigma}^*\right), \quad \text{with} && (2.13)\\
\boldsymbol{\mu}^* &= \mathbf{K}_*^\top\mathbf{K}^{-1}\boldsymbol{\mu}, && (2.14)\\
\boldsymbol{\Sigma}^* &= \mathbf{K}_{**} - \mathbf{K}_*^\top\mathbf{K}^{-1}\mathbf{K}_* + \mathbf{K}_*^\top\mathbf{K}^{-1}\boldsymbol{\Sigma}\mathbf{K}^{-1}\mathbf{K}_* && (2.15)\\
&= \mathbf{K}_{**} - \mathbf{K}_*^\top\left(\mathbf{K}^{-1} + \mathbf{K}^{-1}\boldsymbol{\Sigma}\mathbf{K}^{-1}\right)\mathbf{K}_*. && (2.16)
\end{aligned}$$

Eq. 2.14 and Eq. 2.16 are the two general equations needed to make predictions for any GP model with a Gaussian posterior—also when the Gaussian posterior is an approximation to the true posterior. For the standard Gaussian likelihood, inserting the expressions for the mean and variance from Eq. 2.7 yields

$$\begin{aligned}
\boldsymbol{\mu}^* &= \mathbf{K}_*^\top\left(\mathbf{K} + \sigma^2\mathbf{I}_{n\times n}\right)^{-1}\mathbf{y} && (2.17)\\
\boldsymbol{\Sigma}^* &= \mathbf{K}_{**} - \mathbf{K}_*^\top\left(\mathbf{K}^{-1} + \mathbf{K}^{-1}\boldsymbol{\Sigma}\mathbf{K}^{-1}\right)\mathbf{K}_* && (2.18)\\
&= \mathbf{K}_{**} - \mathbf{K}_*^\top\left(\mathbf{K}^{-1} + \mathbf{K}^{-1}\left(\mathbf{K}^{-1} + \frac{1}{\sigma^2}\mathbf{I}_{n\times n}\right)^{-1}\mathbf{K}^{-1}\right)\mathbf{K}_* && (2.19)\\
&= \mathbf{K}_{**} - \mathbf{K}_*^\top\left(\mathbf{K} + \sigma^2\mathbf{I}_{n\times n}\right)^{-1}\mathbf{K}_*, && (2.20)
\end{aligned}$$

where the last expression is obtained by using the Woodbury identity [Petersen and Pedersen, 2008, Eq. 145]. Although the previous procedure is rather tedious for deriving the predictive distribution in the standard GP regression case compared to the procedure by Rasmussen and Williams [2006, Chapter 2.2], Eq. 2.14 and Eq. 2.16 apply for any GP model where a Gaussian (approximation to the) posterior can be obtained.

This concludes the introduction to GPs. The next sections describe a few general concepts useful for general GP models. Firstly, Section 2.1.1 contains a brief overview of covariance functions relevant for the work described in this thesis. Multi-task and collaborative filtering extensions for GPs are described in Section 2.1.2. In Section 2.1.3, a brief introduction to approximate inference is given focusing in particular on the Laplace approximation. In Section 2.1.4, the idea of *sparse* GPs is introduced which seeks to reduce the $\mathcal{O}(n^3)$ computational cost for general GP models.

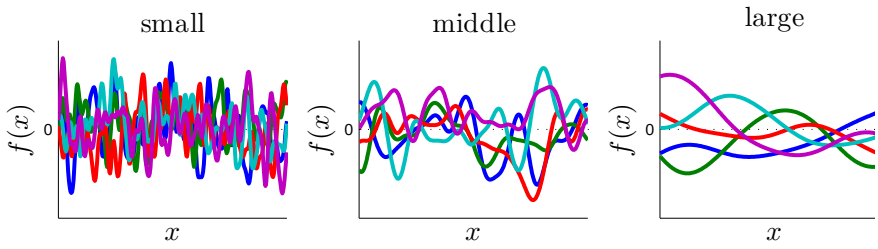


Figure 2.1: Samples from a (zero-mean) GP with a *squared exponential* covariance function for different sizes of the length-scale parameter, λ.

2.1.1 Covariance Functions

As mentioned, the covariance function[1], $k(\mathbf{x}, \mathbf{x}')$, fully defines a zero-mean GP prior, partly by one or several length-scale parameter(s), λ. In Fig. 2.1, samples from a GP prior with three different length scales of a standard covariance function are depicted. Obviously, the length scale is directly related to the *smoothness* of the sampled functions. In this section, only a few (stationary) covariance functions related to the present work are presented. For a more complete treatment of kernels—including for instance non-stationary kernels, construction of new kernels from known ones or kernels applicable when the inputs are distinct objects or instances—see for instance Rasmussen and Williams [2006], Bishop [2006].

By far the most widely used covariance function for GPs is the *squared exponential* (SE) kernel [Rasmussen and Williams, 2006, Chapter 4] given by[2]

$$k_{\mathrm{SE}}(\mathbf{x}, \mathbf{x}') = \sigma_f \exp\left(-\frac{1}{2}[\mathbf{x} - \mathbf{x}']^\top \mathbf{L}^{-1}[\mathbf{x} - \mathbf{x}']\right), \tag{2.21}$$

where the psd matrix, $\mathbf{L}$, is refereed to as the *input-covariance matrix*, not to be confused with the covariance matrix, $\mathbf{K}$. The two well-known versions of the SE kernel, namely the *isotropic* (ISO) version, k_{ISO}, and the *automatic relevance determination* (ARD) version k_{ARD}, are obtained when $\mathbf{L} = \lambda^2 \mathbf{I}_{D\times D}$ and $\mathbf{L} = \mathrm{diag}\left([\lambda_1^2, ..., \lambda_D^2]^\top\right)$, respectively[3]. Several other interesting anisotropic—hence, non-diagonal—versions of $\mathbf{L}$ exist, for instance to obtain dimensionality reduction Vivarelli and Williams [1999]. Rasmussen and Williams [2006,

[1] Specifically in relation to GPs, the term *covariance function* instead of kernel is typically used, to indicate that the kernel directly specifies the covariance of the GP prior.

[2] Note, that the definition by Rasmussen and Williams [2006, Chapter 4] does not contain the *function variance*, σ_f^2.

[3] Note, the ISO version of the SE kernel is similar to what is known as the *radial basic function* (RBF) kernel for other algorithms, such as the *support vector machine* (SVM) [Bishop, 2006, Chapter 7]

Eq. 4.22] suggest a general structure where $\mathbf{L}^{-1}$ is a low-rank matrix constructed from a few basis directions. A similar idea is adopted in Pap. F where the input-correlation matrix is directly specified to obtain (a fixed) correlation between adjacent input dimensions.

The SE kernel provides smooth non-linear interpolation between training examples. Hence, the SE covariance function is less suitable for extrapolation purposes because inputs "in distant areas" do not covary with any training data. The predictions are then only influenced by the prior in these distant areas. Naturally, periodic kernels overcome this in settings where a specific periodic structure can be assumed, but more generic approaches have been proposed recently. Duvenaud et al. [2011, 2013] consider finite-order *additive kernels* defined as a sum of higher-order one-dimensional *base* kernels, where interactions between base kernels have the ability to capture global structures. Duvenaud et al. [2011, 2013] show that typically only a few orders are required to capture global structures, which limits the computational burden. Wilson and Adams [2013] adopt a different approach, where a *spectral-mixture* kernel is proposed, in which a spectral density is learned during training capturing local and global patterns.

An interesting, yet less common kernel worth mentioning is the *probability product* (PP) kernel [Jebara et al., 2004]. The PP kernel measures covariance between two distributions instead of between scalar or vector inputs. In case of two Gaussian mixtures as inputs, the particular case of the PP kernel corresponding to the Hellinger divergence is analytically tractable. This could be particularly suitable for specific audio applications in which a single input is a sequence of features extracted at particular points in time.

2.1.2 Multi-Task and Collaborative-Filtering Extensions

In the literature, authors sometimes do not distinguish between *multi-task* (MT) learning and *collaborative filtering* (CF) [Su and Khoshgoftaar, 2009], whereas others might use completely different expressions. However, the idea in both MT and CF learning is that similar or related tasks should be able to learn from each other or *transfer information* between the tasks (users). This is thus useful in several applications where either the amount of data is sparse for a single user, but can be considered dense across multiple users, or where labeling is expensive for single users, but can be spread among several users. In the present thesis, the following distinction will be made between the CF approach and the MT approach. In a CF approach, users are not characterized by a set of observable features, hence the similarity between users is based only on the partly observed data for individual users. In traditional CF problems,

observations are organized in a sparse user-by-item matrix, where missing entries are predicted by some kind of matrix-factorization model. Similar or like-minded users then have similar *loadings* (hidden features). In MT learning, users are described by a set of observable features, such as age, gender etc. The approach in MT learning is to model the data as a function of both task and user features directly.

CF for GPs has been considered by several including Schwaighofer et al. [2005], Yu et al. [2005], where a hierarchical structure is used in which the mean vector and the covariance matrix of a GP prior are drawn from a conjugated prior—a Gaussian distribution for the mean and an inverse Wishart distribution for the covariance matrix. The conjugated prior is thus shared across users. This was later applied to the case of hearing-aid personalization by Birlutiu et al. [2009, 2010].

For MT learning, a hierarchical approach has been suggested by Bakker and Heskes [2004], where the mean vector is a linear combination of user features with a set of weights for individual clusters of users.

A fundamentally different MT formulation was proposed by Yu and Chu [2007] and later considered by Bonilla et al. [2007, 2008, 2010], in which the user features are included directly in the covariance function. This MT formulation basically correlates users directly in the kernel by defining a *MT kernel* as the product of two other kernels—one between user features, $\mathbf{x}^{(u)}$, and one between original input features, $\mathbf{x}$,

$$k_{\text{MT}}([\mathbf{x}, \mathbf{x}^{(u)}], [\mathbf{x}', \mathbf{x}^{(u)\prime}]) = k(\mathbf{x}, \mathbf{x}') \cdot k(\mathbf{x}^{(u)}, \mathbf{x}^{(u)\prime}). \tag{2.22}$$

If the input sets for all users are identical, the MT GP prior covariance matrix, $\mathbf{K}_{\text{MT}}$, is given as the Kronecker product of the covariance matrix between users, $\mathbf{K}^{(u)}$, and the covariance matrix between inputs, $\mathbf{K}$

$$\mathbf{K}_{\text{MT}} = \mathbf{K}^{(u)} \otimes \mathbf{K}. \tag{2.23}$$

Only the MT kernel has been used in the present work in contribution B, and generally, the CF/MT approach has not been a great part of the present work. The concept, however, would be a suitable and interesting research field following the present work once data for several users is collected. At this point, the hierarchical GP framework for CF [Schwaighofer et al., 2005, Yu et al., 2005] reads promising. The reason is that it overcomes a fundamental issue with the MT kernel approach [Bonilla et al., 2007, 2008, 2010], namely that the user features must alone posses the information needed to model the variation across functions for different users. Possibly, a combined approach similar to Houlsby et al. [2012] in which the MT kernel is incorporated at the top level in the hierarchy would be worth studying.

2.1.3 Approximate Posterior Inference

Only few likelihoods—essentially only Gaussian likelihoods—yield analytically tractable posterior distributions (Eq. 2.6). When the posterior is not analytically tractable, one needs to resort to either sampling based inference or analytical posterior approximations. Although sampling-based inference is exact when sampling is performed for an (effectively) infinitely amount of time, these methods are typically slow compared to analytical approximations, making them less attractive in an interactive system. Therefore, only analytical approximations—in particular the *Laplace approximation*—have been considered.

The basic idea of an analytical approximate-inference method is to fit a well-described distribution to the posterior. Typically, this distribution is chosen from the exponential family, and in particular, the Gaussian distribution is the choice for GPs. Thus, approximate inference consists of approximating the intractable posterior distribution, $p(\mathbf{f}|\mathbf{y}, \mathcal{X})$, with a Gaussian, $q(\mathbf{f}|\mathbf{y}, \mathcal{X})$ as

$$q(\mathbf{f}|\mathbf{y}, \mathcal{X}) = \mathcal{N}(\mathbf{f}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \approx p(\mathbf{f}|\mathbf{y}, \mathcal{X}) \tag{2.24}$$

Note already at this point, that once the posterior approximation is given, predictions are performed as described in Sec. 2.1 using Eq. 2.14 and 2.16. There exist essentially three methods for obtaining the Gaussian approximation, $q(\mathbf{f}|\mathbf{y}, \mathcal{X})$; the *Laplace approximation* [Mackay, 2003, Chapter 27], *Expectation Propagation* (EP) by Minka [2001], and *Variational Bayes* (VB) [Mackay, 2003, Chapter 33]. In general though, not all methods are equally suitable for a particular likelihood. For GPs, the two most common approximate inference methods are the Laplace approximation [Williams and Barber, 1998, Rasmussen and Williams, 2006, Section 3.4] and EP [Rasmussen and Williams, 2006, Section 3.6]. The Laplace approximation has been used to perform approximate inference for the pairwise models developed in the present work mainly due to its simplicity, although EP is generally found to better capture the relevant mass of the intractable posterior.

In the Laplace approximation, the mode of the true posterior, $\hat{\mathbf{f}}$, is found. Then the Gaussian approximation to the posterior is obtained as

$$q_{\mathrm{LAP}}(\mathbf{f}|\mathbf{y}, \mathcal{X}) = \mathcal{N}\left(\mathbf{f}|\hat{\mathbf{f}}, (\mathbf{K}^{-1} + \mathbf{W})^{-1}\right), \tag{2.25}$$

where $\mathbf{W} = -\nabla\nabla_{\mathbf{f}} \log p(\mathbf{y}|\mathbf{f})$ is the Hessian of the negative log likelihood at the mode, which is typically diagonal. Exceptions are the pairwise models described in Sec. 2.3. The expression for the covariance of the approximation is the negative inverse Hessian of the log posterior at the mode. The mode, $\hat{\mathbf{f}}$, is found with a Newton method given by [Rasmussen and Williams, 2006, Eq. 3.18]

$$\mathbf{f}^{\mathrm{new}} = (\mathbf{K}^{-1} + \mathbf{W})^{-1}(\mathbf{W}\mathbf{f} + \nabla \log p(\mathbf{y}|\mathbf{f})). \tag{2.26}$$

The Laplace approximation also gives an (approximate) expression [Rasmussen and Williams, 2006, Eq. 3.32] for the log marginal likelihood, $\log p(\mathbf{y}|\mathcal{X})$,

$$\log q(\mathbf{y}|\mathcal{X}) = \log p(\mathbf{y}|\hat{\mathbf{f}}) - \frac{1}{2}\hat{\mathbf{f}}^\top \mathbf{K}^{-1}\hat{\mathbf{f}} - \frac{1}{2}\log\det(\mathbf{I}_{n\times n} + \mathbf{KW}), \tag{2.27}$$

which is optimized with respect to the hyper parameters, $\boldsymbol{\theta}$, to train the GP using the ML-II scheme or with additional hyper priors using the MAP-II scheme.

The EP approximation method approaches the approximating Gaussian, $q(\mathbf{f}|\mathbf{y},\mathcal{X})$, differently than the Laplace approximation. EP utilizes that it is normally assumed that the likelihood factorizes, such that the GP posterior factorizes as

$$\begin{aligned} p(\mathbf{f}|\mathbf{y},\mathcal{X}) &= \frac{1}{Z}\mathcal{N}(\mathbf{f}|\mathbf{0},\mathbf{K})\prod_{k=1}^{m} p(y_k|\mathbf{f}_k), \quad \text{where} \\ p(y_k|\mathbf{f}_k) &\approx \tilde{Z}_k \mathcal{N}\left(\mathbf{f}_k|\tilde{\boldsymbol{\mu}}_k, \tilde{\boldsymbol{\Sigma}}_k\right) \Rightarrow \\ q_{\text{EP}}(\mathbf{f}|\mathbf{y},\mathcal{X}) &= \frac{1}{Z_{EP}}\mathcal{N}(\mathbf{f}|\mathbf{0},\mathbf{K})\prod_{k=1}^{m}\mathcal{N}\left(\mathbf{f}_k|\tilde{\boldsymbol{\mu}}_k, \tilde{\boldsymbol{\Sigma}}_k\right) \\ &= \mathcal{N}(\mathbf{f}|\boldsymbol{\mu},\boldsymbol{\Sigma}), \end{aligned} \tag{2.28}$$

where Z has been substituted for the marginal likelihood, $\mathbf{f}_k$ is the function values of the k'th likelihood term and $\tilde{\boldsymbol{\Sigma}}_k$ is diagonal[4]. Note, that for standard GP regression and classification [Rasmussen and Williams, 2006, Chapter 2 & 3] $m = n$, $k = i$ and $\mathbf{f}_k = [f_i]$ has only a single variable such that the approximation is given [Rasmussen and Williams, 2006, Eq. 3.53]

$$\begin{aligned} \boldsymbol{\mu} &= \boldsymbol{\Sigma}\tilde{\boldsymbol{\Sigma}}^{-1}\tilde{\boldsymbol{\mu}} \\ \boldsymbol{\Sigma} &= \left(\tilde{\boldsymbol{\Sigma}}^{-1} + \mathbf{K}^{-1}\right), \end{aligned} \tag{2.29}$$

where $\tilde{\boldsymbol{\mu}}$ is a vector with elements $\tilde{\mu}_i$ and $\tilde{\boldsymbol{\Sigma}}$ is simply a diagonal matrix with elements $[\tilde{\boldsymbol{\Sigma}}]_{i,i} = \tilde{\Sigma}_{k=i}$ from Eq. 2.28. For the models described in Sec. 2.3, the latter is not the case as $\mathbf{f}_k = [f(\mathbf{x}_{u_k}), f(\mathbf{x}_{v_k})]^\top$, where $u_k, v_k \in \{1, ..., n\}$, such that $\mathbf{x}_{u_k}, \mathbf{x}_{v_k} \in \mathcal{X}$. In this case, Eq. 2.29 still applies, but the elements $[\tilde{\boldsymbol{\mu}}]_i$ and $[\tilde{\boldsymbol{\Sigma}}]_{i,i}$ are now given as the resulting mean and variance [Petersen and Pedersen, 2008, Eq. 348] from the product of multiple Gaussians—all the Gaussian factors containing f_i, which are essentially spread across several likelihood functions. EP has not been implemented for the pairwise models in Sec. 2.3, but doing so will require some book keeping with respect to the n function values, f_i, appearing in each of the m likelihood terms, $p(y_k|\mathbf{f}_k)$.

[4]For likelihoods depending on more than one functional value as for instance the pairwise likelihood in Sec. 2.3 with two function values, the diagonal version corresponds to a fully factorized posterior, where each multi-function-valued likelihood term is approximated with as many Gaussian factors as function values. This is the simplest factorization.

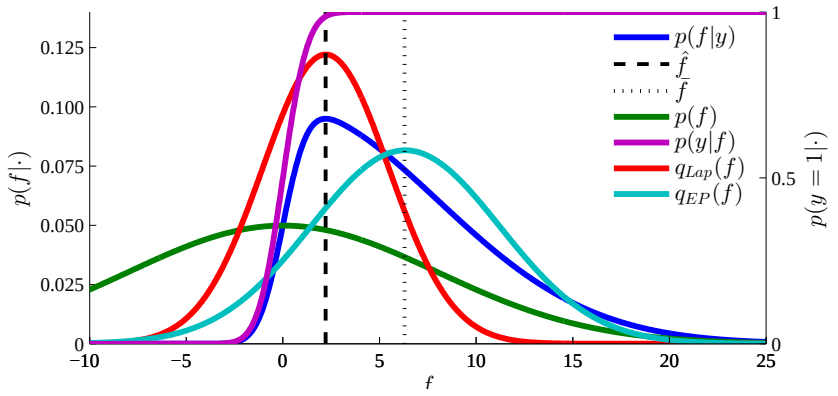


Figure 2.2: Illustration of the difference between the Laplace and the EP approximation on a simple example with the cumulative Gaussian likelihood [Rasmussen and Williams, 2006, Chapter 3]

With the above in mind, the parameters of each Gaussian factor in the approximation[5] are found by moment matching following Rasmussen and Williams [2006, Eq. 3.55-3.57 & 3.59]. The EP approximation to the marginal likelihood, $Z_{EP} \approx p(\mathbf{y}|\mathcal{X})$, is obtained by Rasmussen and Williams [2006, Eq. 3.65].

Fig. 2.2 illustrates a fundamental difference between the Laplace approximation and EP given a skewed one-dimensional posterior. While the Laplace approximation fits a Gaussian distribution around the mode of the true posterior, EP fits a Gaussian around the mean instead, that better captures the probability mass of the true posterior.

2.1.4 Sparse Approximation

Sparsity in the GP context refers to reduction of the $\mathcal{O}(n^3)$ scaling associated with GPs. Some of the earliest attempts, for instance Lawrence et al. [2002], simply reduce the number of data points included in $\mathcal{Y}$ by removing the *least informative* observations according to some criterion. Although such attempts are referred to as sparse GP solutions in the literature, such attempts are in the present thesis *not* considered to be so. Rather, such attempts are specific applications of active learning discussed in Chapter 3, where available observations are thrown away. Hence in this thesis, a sparse GP solution is one which

[5]Note, that as a multi-function-valued likelihood functions is approximated by several Gaussian factors, each one-dimensional Gaussian factor is considered separately.

includes *all* of the available data.

The general idea of the pseudo-input sparse GP formulation originally proposed by Snelson and Ghahramani [2006] is to model the n (original) function values, $\mathbf{f}$, by a smaller set of $\bar{n} << n$ *inducing variables*, $\bar{\mathbf{f}}$, corresponding to a set of *pseudo-inputs*, $\bar{\mathbf{X}} = [\bar{\mathbf{x}}_1, ..., \bar{\mathbf{x}}_{\bar{n}}]^\top$, where $\bar{\mathbf{x}}_j \in \mathbb{R}^D$.[6,7] The inducing variables, $\bar{\mathbf{f}}$, are modeled by the same GP as the original function values, $\mathbf{f}$, hence

$$\begin{bmatrix} \mathbf{f} \\ \bar{\mathbf{f}} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{K} & \mathbf{K}_{\bar{\mathbf{X}}} \\ \mathbf{K}_{\bar{\mathbf{X}}}^\top & \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}} \end{bmatrix} \right), \tag{2.30}$$

where $[\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}]_{i,j} = k(\bar{\mathbf{x}}_i, \bar{\mathbf{x}}_j)$ and $[\mathbf{K}_{\bar{\mathbf{X}}}]_{i,j} = k(\mathbf{x}_i, \bar{\mathbf{x}}_j)$. The trick in the pseudo-input formulation is to define a *sparse* likelihood function as

$$p(y_k|\bar{\mathbf{f}}) = \int p(y_k|\mathbf{f}_k) p(\mathbf{f}_k|\bar{\mathbf{f}}) d\mathbf{f}_k, \tag{2.31}$$

which for standard GP regression yields

$$p(y_i|\bar{\mathbf{f}}) = \int \mathcal{N}\left(y_i|f_i, \sigma^2\right) \mathcal{N}\left(f_i|\mathbf{k}_{\bar{\mathbf{X}}}\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1}\bar{\mathbf{f}}, [\mathbf{K}]_{i,i} - \mathbf{k}_{\bar{\mathbf{X}}}\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1}\mathbf{k}_{\bar{\mathbf{X}}}^\top\right) df_i \tag{2.32}$$

$$= \mathcal{N}\left(y_i|\mathbf{k}_{\bar{\mathbf{X}}}\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1}\bar{\mathbf{f}}, [\mathbf{K}]_{i,i} - \mathbf{k}_{\bar{\mathbf{X}}}\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1}\mathbf{k}_{\bar{\mathbf{X}}}^\top + \sigma^2\right), \tag{2.33}$$

where $\mathbf{k}_{\bar{\mathbf{X}}}$ is the i'th row of $\mathbf{K}_{\bar{\mathbf{X}}}$. The sparse approximation now enters by assuming that the sparse likelihood factorizes, hence

$$p(\mathbf{y}|\bar{\mathbf{f}}) = \prod_{i=1}^{n} p(\mathbf{y}_i|\bar{\mathbf{f}}) = \mathcal{N}\left(\mathbf{y}|\mathbf{K}_{\bar{\mathbf{X}}}\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1}\bar{\mathbf{f}}, \mathbf{\Lambda} + \sigma^2\mathbf{I}_{n\times n}\right) \tag{2.34}$$

where $\mathbf{\Lambda}$ is diagonal with elements $[\mathbf{\Lambda}]_{i,i} = [\mathbf{K}]_{i,i} - \mathbf{k}_{\bar{\mathbf{X}}}\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1}\mathbf{k}_{\bar{\mathbf{X}}}^\top$. With the sparse likelihood, inference is now performed for the inducing variables having a traditional GP prior. Effectively, this means that the original $n \times n$ covariance matrix, $\mathbf{K}$, has been substituted with a smaller $\bar{n} \times \bar{n}$ covariance matrix, $\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}$, which is inverted instead of the original one. Thereby, the computational cost has been reduced to $\mathcal{O}(\bar{n}^2 n)$ compared to the $\mathcal{O}(n^3)$ computational cost of a full GP.

For a unifying view on sparse GPs, see Quiñonero Candela and Rasmussen [2005]. Quiñonero Candela and Rasmussen [2005] define the above version [Snelson and Ghahramani, 2006] as the *fully independent training conditional* (FITC),

[6] Note, that the bar notation, $\bar{}$, is used to indicate pseudo inputs, variables etc, when referring to the sparse pseudo-input formulation and should here not be confused with the mean of a variable.

[7] Note, that the pseudo inputs are explicitly collected in a matrix to indicate that the pseudo-inputs are not true inputs, but parameters of a sparse GP model. Consequently, the locations of the pseudo-inputs can be optimized together with the likelihood and covariance parameters.

whereas a version in which $\mathbf{\Lambda}$ is block diagonal is referred to as the *partially independent training conditional* (PITC), which was also considered by Snelson and Ghahramani [2007]. Usually, predictions are conditionally independent of the original function values, $\mathbf{f}$, given the inducing variables, $\bar{\mathbf{f}}$, in which case FITC becomes FIC/FI(T)C, and PITC becomes PIC/PI(T)C [Quiñonero Candela and Rasmussen, 2005]. The FIC and PIC are common expression found in the literature.

Others have extended the original pseudo-input formulation. Vanhatalo and Vehtari [2008] model global variations with the FIC formulation in combination with compactly supported covariance function to capture local variation. Walder et al. [2008] propose an extension, where each covariance function for the inducing variables was allowed to have its own function variance, σ_f. This avoids the predictive distribution to be widened at new locations, that happen to be far from any pseudo inputs. Lazaro-Gredilla and Figueiras-Vidal [2009] define the pseudo inputs to be in another (lower dimensional) domain than the original function values.

A sparse approximation proposed in Pap. C for binary pairwise observations is described in Sec. 2.3.2.1.

2.2 Bounded Gaussian Process Regression

When observations are bounded and do not have infinite support, i.e., when $y \in]a, b[$ with $a \neq -\infty$ and $b \neq \infty$, a standard GP model from 2.1 is not appropriate. The bounded case occurs if the observations are for instance perceptual ratings, probabilities or proportions. To address this case, the simplest approach is to *warp* the observations onto a space with infinite support (unbounded) and model the warped observations $z = g(y)$ with a traditional GP regression model as Snelson et al. [2004]. Several warping functions, $g(y)$, apply depending on the problem. Snelson et al. [2004] propose a weighted sum of `tanh` functions. Recently, a *Bayesian Warped Gaussian Process* model has been proposed by Lázaro-Gredilla [2012], where the warping function is modeled by an additional GP and is thus non-parametric. Lázaro-Gredilla [2012] show that the model is highly generic in the sense that it supports a broad range of *both* regression *and* classification problems. On the downside, the model by Lázaro-Gredilla [2012] is not analytically tractable as is the case with the original warped GP model by Snelson et al. [2004].

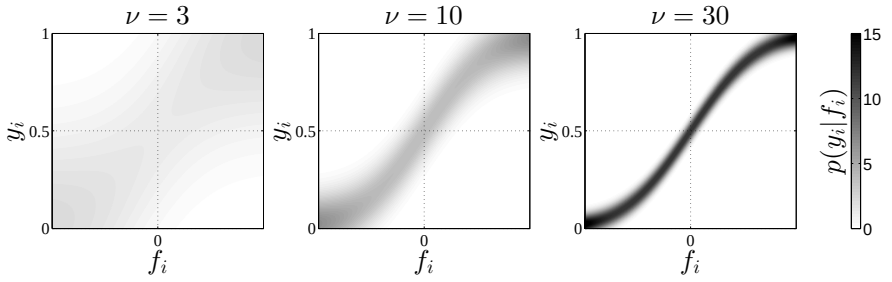
Effectively, the warped GP models model the observational noise indirectly as the noise is modeled in the latent function space with a traditional Gaussian likelihood. A fundamentally different approach is to model the noise in the bounded observational space directly, with an appropriate likelihood function with bounded support. Two alternatives have been proposed in Pap. E—a bounded likelihood based on a truncated distribution and one based on the beta distribution.

2.2.1 Truncated Gaussian Likelihood Function

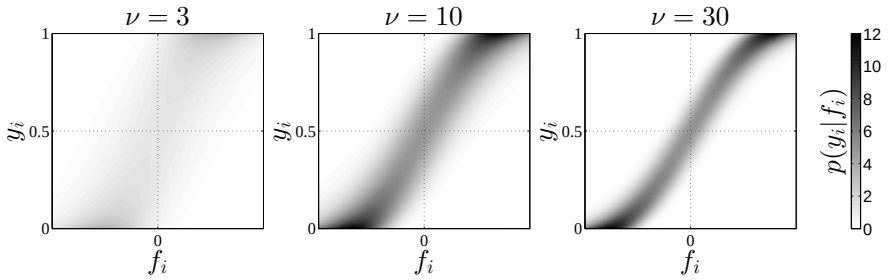
Bounded support can be imposed on the standard Gaussian distribution by truncation, or on any other relevant distribution such as the student's t-distribution. In the present work, the truncated Gaussian (TG) distribution Johnson and Kotz [1970a, Section 7.1]) is considered. A TG likelihood function has been defined as [Pap. E]

$$\mathcal{L}_{TG} \equiv p\left(y_i | f_i, \boldsymbol{\theta}_{\mathcal{L}}\right) = \frac{\nu \mathcal{N}\left(\nu\left(y_i - \hat{\mu}\left(f_i\right)\right)\right)}{\Phi\left(\nu\left(b - \hat{\mu}\left(f_i\right)\right)\right) - \Phi\left(\nu\left(a - \hat{\mu}\left(f_i\right)\right)\right)}, \tag{2.35}$$

where the distribution is parametrized by the mode $\hat{\mu}\left(f_i\right)$ and inverse dispersion parameter, ν. The domain limits a and b are assumed to be 0 and 1, respectively. The mode, $\hat{\mu}\left(f_i\right) = g(f_i)$, is given by a monotonic-increasing non-linear warping function, $g(f_i)$. In the present work, the standard cumulative Gaussian is used, hence $\hat{\mu}\left(f_i\right) = \Phi\left(f_i\right)$. The TG likelihood function is depicted in Fig. 2.3(a).



(a) TG Likelihood



(b) Beta Likelihood

Figure 2.3: Illustration of the (a) TG [Fig. 1, Pap. E] and (b) Beta [Fig. 2, Pap. E] likelihood functions proposed in the present work. $p(y_i|f_i)$ is shown as the gray-scale level for different values of ν.

Another interesting alternative for the warping function is a weighted sum of `tanh` functions similar to Snelson et al. [2004] or of cumulative Gaussian distributions. A more involved alternative is to use a non-parametric parametrization of $g(f_i)$ adopted from Lázaro-Gredilla [2012]. As noted in Pap. E, it is possible to parametrized the mean of the TG by $g(f_i)$ although the mean parametrization requires the use of numerical or approximate methods. This is avoided with the mode parametrization used in the present work [Pap. E].

2.2.2 Beta Likelihood Function

The beta distribution [Johnson and Kotz, 1970b, Chapter 24] has bounded support, thus it is a natural distribution to consider for bounded observations. The beta distribution has previously been applied in parametric settings by Ferrari and Cribari-Neto [2004] and Smithson and Verkuilen [2006].

In the present work, a beta likelihood function is derived by re-parametrizing the shape parameters, α, β, of the beta distribution

$$\text{Beta}\left(y|\alpha,\beta\right) = \frac{1}{B\left(\alpha,\beta\right)}\frac{(y-a)^{\alpha-1}(b-y)^{\beta-1}}{(b-a)^{\alpha+\beta-1}}, \tag{2.36}$$

in terms of the beta mean, $\mu(\mathbf{f}_i)$, such that the Beta likelihood function becomes [Pap. E]

$$\mathcal{L}_{\text{BE}} \equiv p(y_i|f_i,\boldsymbol{\theta}_{\mathcal{L}}) = \text{Beta}\left(y_i|\nu\mu(f_i),\nu\left(1-\mu(f_i)\right)\right), \tag{2.37}$$

where ν is an inverse dispersion parameter. Here, the domain limits, a, b, are assumed to be 0 to 1, respectively. The mean $\mu(f_i) = g(f_i)$ is parametrized by any monotonic-increasing non-linear warping function, $g(f_i)$. As for the TG likelihood function from Sec. 2.2.1, several—more or less involved—alternatives apply for $g(f_i)$. Here, the (standard) cumulative Gaussian is again chosen, hence $\mu(f_i) = \Phi(f_i)$. The resulting Beta likelihood function is depicted in Fig. 2.3(b).

2.2.3 Predicting Bounded Responses

None of the bounded likelihood functions from Sec. 2.2.1-2.2.2 result in analytically tractable posterior distributions. Approximate inference and ML-II parameter optimization for both models are performed with either EP or the Laplace approximation. Details about approximate inference are found in Pap. E. These details are also needed to perform the covariance function and likelihood parameter optimizations described in Sec. 2.1.3.

With a successful Gaussian approximation, $q(\mathbf{f}|\mathbf{y},\mathcal{X}) = \mathcal{N}\left(\mathbf{f}|\boldsymbol{\mu},\boldsymbol{\Sigma}\right)$, to the true posterior, the predictive distribution, $p(\mathbf{f}^*|\mathbf{y},\mathcal{X},\mathcal{X}^*)$, of the latent function values for unseen inputs, $\mathbf{x}_l^*$, follows directly from Eq. 2.14 and 2.16. As always in Bayesian modeling, the predictive distribution of the observed variable, y^*, is given by marginalizing over the joint distribution as

$$p(y^*|\mathcal{Y},\mathcal{X},\mathbf{x}^*) = \int p(y^*|f^*)\mathcal{N}\left(f^*|\mu*,\Sigma^*\right)df^* \tag{2.38}$$

which is only analytically tractable for the warped GP [Snelson et al., 2004, Eq. 8] as

$$p_{GP-WA}(y^*|\mathcal{Y},\mathcal{X},\mathbf{x}^*) = \frac{\mathcal{N}\left(\Phi^{-1}(y^*)|\mu^*,\Sigma^*\right)}{\Phi(\Phi^{-1}(y^*))}. \tag{2.39}$$

For the two bounded likelihood functions, the integration in Eq. 2.38 can only be performed numerically. Fortunately, it is only a one-dimensional integral

per new data point. For the Beta likelihood function, the predictive mean of the observed variable is analytically tractable, and is actually identical to the predictive mean of the warped GP[8] (derivations are shown in Pap. E)

$$\mathbb{E}_{\text{GP-WA}}\{y^*\} = \mathbb{E}_{\text{GP-BE}}\{y^*\} = \Phi\left(\frac{\mu^*}{\sqrt{1+(\sigma^*)^2}}\right).$$

In Fig. 2.2.3, the predictive distributions of the different bounded GP models

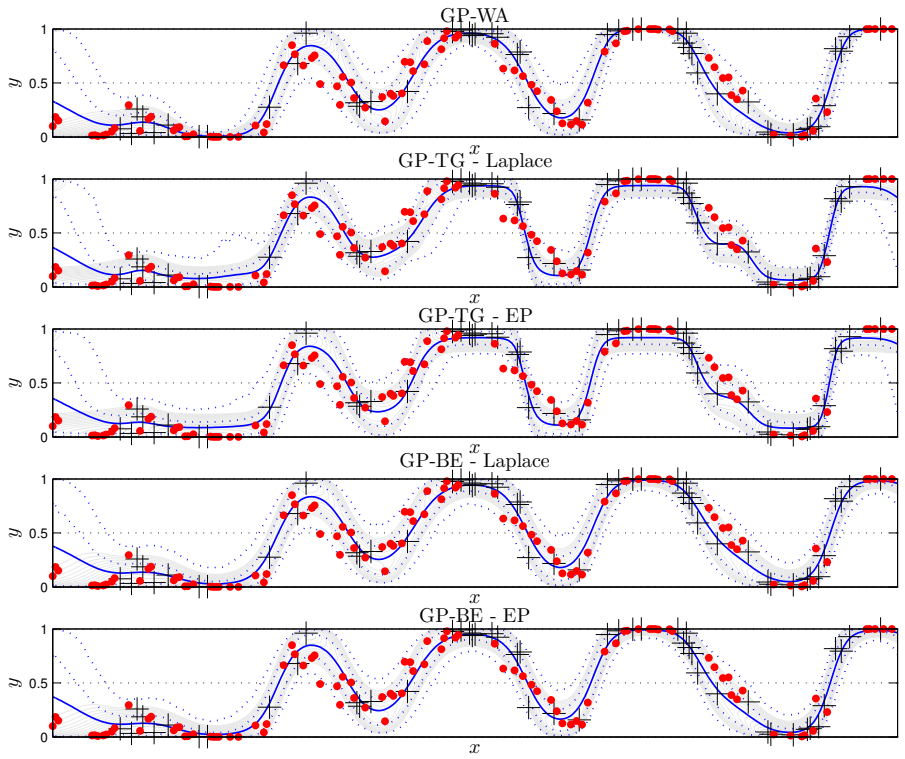


Figure 2.4: Predictive distributions for the three bounded models: warped GP (GP-WA), Truncated Gaussian (GP-TG) and Beta distribution (GP-BE). For GP-TG and GP-BE both Laplace and EP inference are shown. Training data: $+$, test examples: $\cdot$, predictive mean: $-$ and 68% and 95% percentiles: $\cdots$. Contours of the predictive distribution are shown in gray. [Fig. 3, Pap. E]

[8]Keep in mind that although there is an equal sign between the predictive mean of the cumulative-warped and the beta model, the means will in general be different due to the difference in the *latent* predictive distributions of the GP.

are shown using a simple one-dimensional toy example. Note, that the predictive distributions of the three models are different especially near the domain boundaries. In Pap. E, it is shown that the GP with the Beta likelihood models the noise slightly better than the two other models on two real-world data sets, in which the bounded observations are perceptual ratings on a bounded scale. But as noted, the better model really depends—as usual—on the specific application.

2.3 Gaussian Process Regression from Pairwise Data

In this section, $m \neq n$ pairwise observations, $\mathcal{Y} = \{y_k | u_k, v_k, k = 1, ..., m\}$, are given between any two distinct instances, $u_k, v_k \in \{1, ..., n\}$, implying $\mathbf{x}_{u_k}, \mathbf{x}_{v_k} \in \mathcal{X}$. The observations, y_k, can come in one of the following forms; unbounded $y \in \mathbb{R}$ (Sec 2.3.1), binary classification $y \in \{-1, 1\}$ (Sec. 2.3.2), or bounded $y \in]0, 1[$ (Sec. 2.3.3).

In the following sections, it is assumed that any type of relative responses, y_k, are generated by a (possibly non-linear) process depending explicitly on the difference between two function values, $\mathbf{f}_k = [f(\mathbf{x}_{u_k}), f(\mathbf{x}_{v_k})]^\top$, of a latent unobserved process, i.e., the user's IRF, modeled by a GP, thus $f(\mathbf{x}) \sim \mathcal{GP}(0, k(\mathbf{x}, \mathbf{x}'))$. In effect, this turns the pairwise modeling problem into a regression problem in which the observations define relative relations between function values.

Pairwise likelihood functions, $p(y_k|\mathbf{f}_k)$, for the three types of observations will be defined. In the unbounded case (Sec. 2.3.1), the observations are directly the difference between two function values and the observations are polluted with i.i.d Gaussian noise. This makes the corresponding GP model analytically tractable. In the binary case (Sec. 2.3.2), observations are two-alternative forced-choices (2AFC) given as a binary variable. This case is typically referred to as preference learning in the literature. In the bounded case (Sec. 2.3.3), observations are given as a bounded response capturing for instance the *degree of preference* between two input instances. The models for the last two cases above do not yield analytically tractable posterior distributions. Details about approximate inference using the Laplace approximation are found in Jensen and Nielsen [2011].

2.3.1 Pairwise Gaussian Likelihood

Formally, the observations, y_k, are noisy versions of the difference between the two function values, hence

$$y_k = f(\mathbf{x}_{v_k}) - f(\mathbf{x}_{u_k}) + \epsilon, \quad \epsilon \sim \mathcal{N}\left(0, \sigma^2\right) \tag{2.40}$$

It is convenient to formulate a sparse $m \times n$ *indicator matrix*, $\mathbf{M}$, with non-zero elements given as

$$[\mathbf{M}]_{k,u_k} = -1, \quad [\mathbf{M}]_{k,v_k} = 1 \tag{2.41}$$

Using the sparse indicator matrix, a pairwise Gaussian likelihood is defined as

$$\begin{aligned} p(\mathbf{y}|\mathbf{f}) &= \prod_{k=1}^{m} p(y_k|\mathbf{f}_k) = \prod_{k=1}^{m} \mathcal{N}\left(y_k|f(\mathbf{x}_{v_k}) - f(\mathbf{x}_{u_k}), \sigma^2\right) \\ &= \mathcal{N}\left(\mathbf{y}|\mathbf{M}\mathbf{f}, \sigma^2\mathbf{I}_{m\times m}\right). \end{aligned} \tag{2.42}$$

Using the GP prior from Eq. 2.2, Bayes rule yields

$$p(\mathbf{f}|\mathbf{y},\mathcal{X}) = \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathcal{X})}{p(\mathbf{y}|\mathcal{X})} \propto \mathcal{N}\left(\mathbf{y}|\mathbf{M}\mathbf{f}, \sigma^2\mathbf{I}_{m\times m}\right)\mathcal{N}\left(\mathbf{f}|\mathbf{0},\mathbf{K}\right) \tag{2.43}$$

The posterior, $p(\mathbf{f}|\mathbf{y},\mathcal{X})$, and marginal likelihood, $p(\mathbf{y}|\mathcal{X})$, are obtained from Eq. 2.116 and Eq. 2.115 in Bishop [2006], respectively, which result in

$$p(\mathbf{f}|\mathbf{y},\mathcal{X}) = \mathcal{N}\left(\mathbf{f}|\boldsymbol{\mu},\boldsymbol{\Sigma}\right), \quad \text{with} \tag{2.44a}$$

$$\boldsymbol{\mu} = \left(\mathbf{K}^{-1} + \frac{1}{\sigma^2}\mathbf{M}^\top\mathbf{M}\right)^{-1}\frac{1}{\sigma^2}\mathbf{M}^\top\mathbf{y} \tag{2.44b}$$

$$= \mathbf{K}\left(\mathbf{M}^\top\mathbf{M}\mathbf{K} + \sigma^2\mathbf{I}_{n\times n}\right)^{-1}\mathbf{M}^\top\mathbf{y} \tag{2.44c}$$

$$\boldsymbol{\Sigma} = \left(\mathbf{K}^{-1} + \frac{1}{\sigma^2}\mathbf{M}^\top\mathbf{M}\right)^{-1} \tag{2.44d}$$

$$p(\mathbf{y}|\mathcal{X}) = \mathcal{N}\left(\mathbf{y}|\mathbf{0}, \sigma^2\mathbf{I}_{m\times m} + \mathbf{M}\mathbf{K}\mathbf{M}^\top\right) \tag{2.44e}$$

The predictive distribution is as usual obtained by inserting the posterior mean and covariance from Eq. 2.44 into Eq. 2.14 and Eq. 2.16 to give

$$\boldsymbol{\mu}^* = \mathbf{K}_*^\top\mathbf{K}^{-1}\boldsymbol{\mu} = \mathbf{K}_*^\top\left(\mathbf{M}^\top\mathbf{M}\mathbf{K} + \sigma^2\mathbf{I}_{n\times n}\right)^{-1}\mathbf{M}^\top\mathbf{y} \tag{2.45}$$

$$\boldsymbol{\Sigma}^* = \mathbf{K}_{**} - \mathbf{K}_*^\top\left(\mathbf{K}^{-1} + \mathbf{K}^{-1}\boldsymbol{\Sigma}\mathbf{K}^{-1}\right)\mathbf{K}_* \tag{2.46}$$

$$= \mathbf{K}_{**} - \mathbf{K}_*^\top\left(\mathbf{K}^{-1} + \mathbf{K}^{-1}\left(\mathbf{K}^{-1} + \frac{1}{\sigma^2}\mathbf{M}^\top\mathbf{M}\right)^{-1}\mathbf{K}^{-1}\right)\mathbf{K}_* \tag{2.47}$$

$$= \mathbf{K}_{**} - \mathbf{K}_*^\top\left(\mathbf{K} + \sigma^2[\mathbf{M}^\top\mathbf{M}]^{-1}\right)^{-1}\mathbf{K}_* \tag{2.48}$$

$$= \mathbf{K}_{**} - \mathbf{K}_*^\top[\mathbf{M}^\top\mathbf{M}][\mathbf{M}^\top\mathbf{M}]^{-1}\left(\mathbf{K} + \sigma^2[\mathbf{M}^\top\mathbf{M}]^{-1}\right)^{-1}\mathbf{K}_* \tag{2.49}$$

$$= \mathbf{K}_{**} - \mathbf{K}_*^\top[\mathbf{M}^\top\mathbf{M}]\left(\mathbf{K}[\mathbf{M}^\top\mathbf{M}] + \sigma^2\mathbf{I}_{n\times n}\right)^{-1}\mathbf{K}_* \tag{2.50}$$

Note, that the warped GP framework [Snelson et al., 2004] can be applied in this pairwise case to transform bounded observations into unbounded versions. Thereby, an analytically tractable GP model is obtained, which however models the observational noise implicitly in latent space.

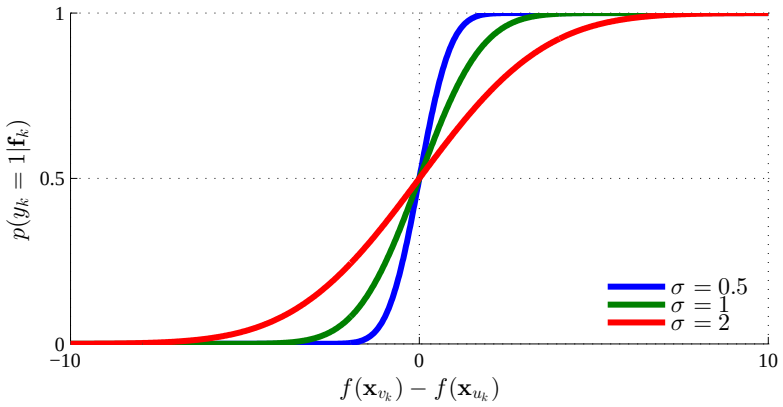


Figure 2.5:

2.3.2 Preference Learning

For preference learning [Fürnkranz, 2010, see *object ranking*], the response variable is a 2AFC paired-comparison, such that $y_k \in \{-1, 1\}$ indicates a preference for either u_k or v_k, respectively. Noisy observations of this type have historically been modeled either by the Logit or Probit choice model Bock and Jones [1968, chapter 6]. In the present work, only the Probit model is considered mainly for analytical reasons.

Given the function, f, the likelihood of observing the binary choice, y_k, is directly modeled as

$$p\left(y_k|\mathbf{f}_k, \boldsymbol{\theta}_{\mathcal{L}}\right) = \Phi\left(y_k \frac{f\left(\mathbf{x}_{v_k}\right) - f\left(\mathbf{x}_{u_k}\right)}{\sqrt{2}\sigma}\right), \tag{2.51}$$

where $\Phi(\cdot)$ is the standard cumulative Gaussian—with zero mean and unity variance—and $\boldsymbol{\theta}_{\mathcal{L}} = \{\sigma\}$. This classic Probit likelihood function is by no means a new invention and can be dated back to Thurstone and his fundamental definition of *The Law of Comparative Judgment* [Thurstone, 1927]. However, it has first been considered with GPs by Chu and Ghahramani [2005a] and later by for instance Chu and Ghahramani [2005b] and Bonilla et al. [2010]. The cumulative Gaussian pairwise likelihood for different values of the noise parameter, σ, is depicted in Fig 2.5.

Given an analytical approximation to the intractable posterior, $q(\mathbf{f}|\mathcal{Y}, \mathcal{X}) = \mathcal{N}\left(\mathbf{f}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$—which in the present work is obtained by the Laplace approximation [Jensen and Nielsen, 2011]— the predictive distribution, $p(\mathbf{f}^*|\mathcal{Y}, \mathcal{X}, \mathbf{x}^*)$, of

latent function values follows directly from Eq. 2.14 and Eq. 2.16. Since the cumulative Gaussian is an odd function, the (hard) prediction of the preference relation $\mathbf{x}_r^* \succ \mathbf{x}_s^*$ or $\mathbf{x}_r^* \prec \mathbf{x}_s^*$ between two new inputs, $\mathbf{x}_r^*, \mathbf{x}_s^* \in \mathbb{R}$, is given by which of the inputs that has the largest predicted function mean value, μ^*. The predictive distribution, $p(y^*|\mathcal{Y}, \mathcal{X}, \mathbf{x}_r^*, \mathbf{x}_s^*)$, of the binary label involves computing the expectation of the likelihood function under the two-dimensional predictive distribution of the latent function values, f_r^*, f_s^*, which yields [Chu and Ghahramani, 2005a]

$$P(\mathbf{x}_r^* \succ \mathbf{x}_s^*|\mathcal{Y}) = \Phi\left(\frac{\mu_r^* - \mu_s^*}{\sqrt{2\sigma^2 + \Sigma_{r,r}^* + \Sigma_{s,s}^* - 2\Sigma_{r,s}^*}}\right) \tag{2.52}$$

2.3.2.1 Sparse Approximation for Cumulative Gaussian

In Pap. C, it is shown that for the cumulative Gaussian likelihood, it is possible to analytically solve the integration in Eq. 2.31 to obtain a *sparse* pseudo-input formulation of the cumulative Gaussian likelihood from Eq. 2.51 as

$$p(y_k|\bar{\mathbf{f}}) = \int p(y_k|\mathbf{f}_k)p(\mathbf{f}_k|\bar{\mathbf{f}})d\mathbf{f}_k \tag{2.53}$$

$$= \int \Phi\left(y_k \frac{f(\mathbf{x}_{v_k}) - f(\mathbf{x}_{u_k})}{\sqrt{2}\sigma}\right) \mathcal{N}\left(\mathbf{f}_k|\mathbf{K}_{k\bar{\mathbf{X}}}\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1}\bar{\mathbf{f}}, \bar{\mathbf{\Sigma}}\right) d\mathbf{f}_k, \tag{2.54}$$

$$= \Phi\left(y_k \frac{([\mathbf{K}_{\bar{\mathbf{X}}}]_{v_k} - [\mathbf{K}_{\bar{\mathbf{X}}}]_{u_k})\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1}\bar{\mathbf{f}}}{\sigma_{\bar{\mathbf{X}},k}}\right) \tag{2.55}$$

where $[\mathbf{K}_{\bar{\mathbf{X}}}]_i$ denotes the i'th row in $\mathbf{K}_{\bar{\mathbf{X}}}$ from Eq. 2.30, and

$$\begin{bmatrix}\mathbf{f}_k \\ \bar{\mathbf{f}}\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\mathbf{0} \\ \mathbf{0}\end{bmatrix}, \begin{bmatrix}\mathbf{K}_k & \mathbf{K}_{k\bar{\mathbf{X}}} \\ \mathbf{K}_{k\bar{\mathbf{X}}}^\top & \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}\end{bmatrix}\right), \tag{2.56}$$

$$\bar{\mathbf{\Sigma}} = \mathbf{K}_k - \mathbf{K}_{k\bar{\mathbf{X}}}\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1}\mathbf{K}_{k\bar{\mathbf{X}}}^\top, \tag{2.57}$$

$$\sigma_{\bar{\mathbf{X}},k} = \sqrt{2\sigma^2 + [\bar{\mathbf{\Sigma}}]_{1,1} + [\bar{\mathbf{\Sigma}}]_{2,2} + 2[\bar{\mathbf{\Sigma}}]_{1,2}}. \tag{2.58}$$

In Pap. C, inference is performed using the Laplace approximation and the locations of the pseudo inputs are optimized together with hyper parameters using ML-II optimization (details are shown in Pap. C). An illustration of the sparse cumulative Gaussian likelihood and a GP prior (SPGP) is shown in Fig. 2.6. Notice, that especially the predictive standard deviation of the SPGP model differs significantly from the full GP model. As mentioned in Sec. 2.1.4, Walder et al. [2008] address this issue by having individual function variances, σ_f^2, in each of the kernel functions for the pseudo inputs.

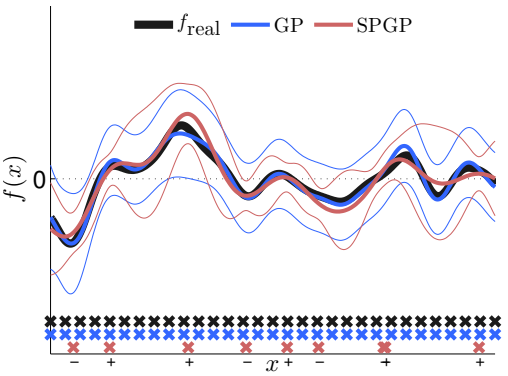


Figure 2.6: Predictive distribution of a full (GP) and sparse (SPGP) pairwise GP, respectively. f_{real} is a single draw from a GP prior used to generate a binary pairwise data set between the inputs marked with black crosses. The colored crosses indicate the (pseudo/real) inputs of the full and sparse GP models in corresponding color. In the example $n = 31$, $\bar{n} = 9$ and $m = 465$. [Fig. 1(a), Pap. C]

2.3.3 Continuous and Bounded Observations

The information carried in each binary preference relation does not express to what extent either of the two input instances, u_k or v_k, is preferred over the other. If the extend or degree of a single preference relation is available and the inherent noise can be modeled robustly, it is possible to learn preference relations—and thus indirectly a more informative latent representation—with a smaller set of observations. In the present work, the degree of preference is captured by formally defining the domain of the response variable as $y_k \in]0, 1[$. The first option, u_k, is preferred for $y_k < 0.5$. The second option, v_k, is preferred for $y_k > 0.5$ and none is preferred for $y_k = 0.5$. Hence, the response captures both the choice between u_k and v_k, and the degree of the preference.

The observational noise is modeled by the beta distribution similar to Sec. 2.2.2 in which the cumulative Gaussian is used as a link function to specify the mean, $\mu(\mathbf{f}_k, \sigma)$, of the beta distribution as

$$\mu\left(\mathbf{f}_k, \sigma\right) = \Phi\left(\frac{f\left(\mathbf{x}_{v_k}\right) - f\left(\mathbf{x}_{u_k}\right)}{\sqrt{2}\sigma}\right). \tag{2.59}$$

By parametrizing the beta distribution by the mean, the beta likelihood function becomes

$$p\left(y_k|\mathbf{f}_k, \boldsymbol{\theta}_{\mathcal{L}}\right) = \text{Beta}\left(y_k|\, \nu\mu(\mathbf{f}_k, \sigma), \nu(1 - \mu(\mathbf{f}_k, \sigma))\right), \tag{2.60}$$

$$= \text{Beta}\left(y_k|\, \alpha(\mathbf{f}_k), \beta(\mathbf{f}_k)\right), \quad \begin{matrix} \alpha(\mathbf{f}_k) = \nu\mu(\mathbf{f}_k, \sigma) \\ \beta(\mathbf{f}_k) = \nu(1 - \mu(\mathbf{f}_k, \sigma)) \end{matrix} \tag{2.61}$$

where ν is an inverse dispersion parameter and $\boldsymbol{\theta}_{\mathcal{L}} = \{\sigma, \nu\}$. The proposed beta likelihood function is depicted in Fig. 2.3.3 for different values of ν. Since

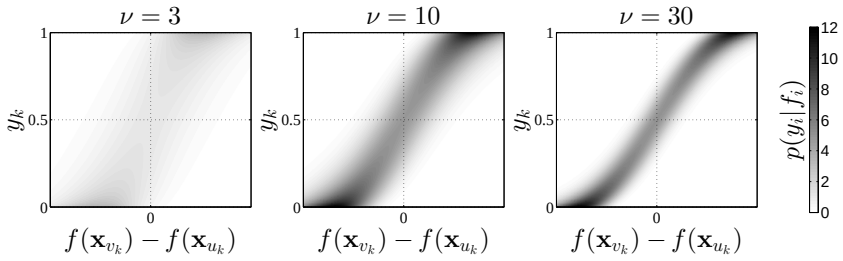


Figure 2.7: Illustration of beta likelihood with $p(\pi_k|\mathbf{f}_k, \boldsymbol{\theta}_\mathcal{L})$ shown as a color level. The likelihood parameters $\boldsymbol{\theta}_\mathcal{L}$ are $\sigma = 0.1$ and left: $\nu = 3$, middle: $\nu = 10$ and right: $\nu = 30$

posterior inference with the beta likelihood function is analytically intractable, approximate inference based on the Laplace approximation is performed, which is considered in Pap. A. Details are found in Jensen and Nielsen [2011]. With an approximation to the true posterior, the predictive distribution, $p(\mathbf{f}_t^*|\mathcal{Y}, \mathcal{X}, \mathbf{x}_r^*, \mathbf{x}_s^*)$, of latent function values, $\mathbf{f}_t^* = [f_r^*, f_s^*]^\top$, follows from Eq. 2.14 and Eq. 2.16. The predictive distribution of y^*,

$$p(y^*|\mathcal{Y}, \mathcal{X}, \mathbf{x}_r^*, \mathbf{x}_s^*) = \int p(y^*|\mathbf{f}_t^*, \boldsymbol{\theta}_\mathcal{L}) p(\mathbf{f}_t^*|\mathcal{Y}, \mathcal{X}, \mathbf{x}_r^*, \mathbf{x}_s^*) d\mathbf{f}_t^* \tag{2.62}$$

$$= \int \text{Beta}\left(y_k|\, \alpha(\mathbf{f}_t^*), \beta(\mathbf{f}_t^*)\right) \mathcal{N}\left(\mathbf{f}_t^*|\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*\right) d\mathbf{f}_t^* \tag{2.63}$$

involves an integration which is analytically intractable. Instead, numerical integration is used. The predictive distribution for binary relations, $\mathbf{x}_r^* \succ \mathbf{x}_s^*$ or $\mathbf{x}_r^* \prec \mathbf{x}_s^*$, can be obtained as

$$p(\mathbf{x}_r^* \succ \mathbf{x}_s^*|\mathcal{Y}, \mathcal{X}, \mathbf{x}_r^*, \mathbf{x}_s^*) = \int_0^{1/2} p(y^*|\mathcal{Y}, \mathcal{X}, \mathbf{x}_r^*, \mathbf{x}_s^*) dy^* \tag{2.64}$$

$$= \int_0^{1/2} \int \text{Beta}\left(y_k|\, \alpha(\mathbf{f}_t^*), \beta(\mathbf{f}_t^*)\right) \mathcal{N}\left(\mathbf{f}_t^*|\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*\right) d\mathbf{f}_t^* dy^* \tag{2.65}$$

$$= \int \int_0^{1/2} \text{Beta}\left(y_k|\, \alpha(\mathbf{f}_t^*), \beta(\mathbf{f}_t^*)\right) \mathcal{N}\left(\mathbf{f}_t^*|\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*\right) dy^* d\mathbf{f}_t^* \tag{2.66}$$

$$= \int \mathcal{N}\left(\mathbf{f}_t^*|\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*\right) \int_0^{1/2} \text{Beta}\left(y_k|\, \alpha(\mathbf{f}_t^*), \beta(\mathbf{f}_t^*)\right) dy^* d\mathbf{f}_t^* \tag{2.67}$$

$$= \int \mathcal{N}\left(\mathbf{f}_t^*|\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*\right) \text{Betacdf}\left(\frac{1}{2}\middle|\, \alpha(\mathbf{f}_t^*), \beta(\mathbf{f}_t^*)\right) d\mathbf{f}_t^*, \tag{2.68}$$

where by definition probabilities are finite measures in which case Fubini's theorom is used to interchange the integration in Eq. 2.66. Note, that in case

of noise-free predictions, $\nu \to \infty$, the beta distribution becomes a point mass around the mean and Eq. 2.68 has an analytical solution similar to Eq. 2.52.

In the present work, the proposed beta likelihood function is motivated from the perspective of learning an *individual* latent representation of the user's IRF over the adjustable parameters of a (HA) device with *a minimum* of observations while preserving in particular the robustness contained in a pairwise paradigm. Ultimately, the end goal is to optimize the IRF—not to predict outcomes of the bounded observed variable. Hence, the described issues mentioned above concerning the predictive distribution of relations—both binary, $\mathbf{x}_r^* \succ \mathbf{x}_s^*$, and bounded, y^*—with the beta likelihood function, are not of particular interest in the present work. However, a success criterion is that the beta likelihood function should require fewer observations to obtain a sufficient latent representation. In Pap. A, the cumulative Gaussian likelihood is compared to the beta likelihood on an artificial example with different amounts of observational noise. A contribution of the present work is the analysis showing that the beta likelihood function is always ahead of the cumulative Gaussian likelihood given the same number of observations. Moreover, the beta likelihood function is used to optimize settings of hearing aids in Pap. D, G and H.

As a final remark, it should be noted that the IRF, $f(\mathbf{x})$—together with a specific active learning criterion (see Chapter 3)—is used to optimize parameters, $\mathbf{x}$. However, *individual* IRFs can neither be compared across users, likelihood functions nor data sets. A valid approach to make comparisons across users or data is to do it in observational space, in which case the predictive distribution of the observations enters.

2.3.4 Pairwise Judgment Kernel

Recently, Huszár [2011], Houlsby et al. [2012] proposed a *pairwise-judgment* (PJ) kernel suitable for relational learning. Seen from a GP perspective, the PJ kernel circumvents the pairwise problem into a *single-instance* problem, by using the GP prior to model the difference between the function vales, $g(\mathbf{x}_v, \mathbf{x}_u) = f(\mathbf{x}_v) - f(\mathbf{x}_u) \sim \mathcal{GP}\left(0, k_{\mathrm{PJ}}((\mathbf{x}_v, \mathbf{x}_u), (\mathbf{x}'_v, \mathbf{x}'_u))\right)$. Hence from the GP perspective, an input now contains two instances, u and v, which changes the computational cost compared to the pairwise framework outlined in the present work. Two extreme pairwise cases can be identified if only unique comparisons exist, meaning that a comparison between any two instances is only made once. In the first extreme case, all possible relations among instances are observed whereby $m = \frac{n^2-n}{2}$. In the other extreme case, all instances are only compared to exactly one other instance, whereby $m = 2n$. Hence, n is generally neither larger nor smaller than m, but for all pairwise problems with only unique comparisons, n and m are

limited by each other by

$$n \leq 2m \leq n^2 - n. \tag{2.69}$$

The covariance matrix for the outlined framework is $n \times n$ and $m \times m$ for the PJ kernel formulation. When $m >> n$, the computational training cost of the PJ kernel formulation is dominated by the $\mathcal{O}(m^3)$ inversion of the covariance matrix. In this case, the framework outlined in the present work has a large computational advantage because it only has computational costs composed of $\mathcal{O}(nm^2)$ and $\mathcal{O}(n^3)$. As seen from the right-hand side inequality 2.69, $m >> n$ is a likely scenario. When $n >> m$, the computational training cost of the outlined framework is dominated by the $\mathcal{O}(n^3)$ inversion of the covariance matrix. This gives the computational advantage to the PJ kernel formulation, because it only has computational costs composed of $\mathcal{O}(mn^2)$ and $\mathcal{O}(m^3)$. This however is not a likely scenario, since from the left hand-side inequality 2.69, n can only be twice as large as m, which is not an order of magnitude.

Besides the above computational considerations, the PJ kernel formulation and the framework outlined in the present work are equally applicable to different pairwise problems, and active learning (Chapter 3) is applicable for both as well. Hence, besides the possible computational advantage with one formulation or the other, which to use is basically a matter of taste. Nevertheless, for the problem in this work where the end goal is to find the maximum of the latent function, f, the outlined framework is more appealing in practice as it constitutes a more direct route to the function, f, itself, and thus to the inputs, $\mathbf{x}$.

Chapter 3

Active Learning

In classical machine learning, observations are given beforehand and the task is to learn an appropriate model. In (discriminative) Bayesian modeling, this—loosely speaking—boils down to inferring the posterior distribution under the assumptions implied in the likelihood concerning the observational noise and in the prior concerning the model.

In active learning, all observations are not given beforehand and new observations are expensive to obtain. Consequently, only informative new observations should be queried. A variety of active learning settings exist. In the present work, active learning is used to find the more informative next input for which to query a label from the user (new observation). For this, the probabilistic modeling approach described in Chapter 2 is utilized.

3.1 Introduction to Active Learning

For some problems, it is expensive—for instance with respect to time, cost and/or computation—to obtain new observations. This is particular the case in the present work, where a new observation requires that a user listens carefully to one ore more input instance(s) and make a specific type of assessment based on the user's own perception. This is both tiring and time consuming for a user to do. Consequently, the number of total assessments that each user needs to perform in order to obtain a personalized device setting must be minimized. This problem is addressed by adopting a sequential approach in which the model is trained after each assessment, where after active learning is used to answer the question: "Where to measure next in order to improve the current model (ideally, as much as possible)?". Active learning is typically based on an approach where the horizon is only one step into the future, due to an otherwise extensive computational burden of *propagating* uncertainty multiple steps into the future. A one-step approach is also used in the sequential design applied in the present work.

Typically, model improvement refers to the generalization performance of a model for all future inputs. For this the expected reduction in posterior Shannon entropy Mackay [2003] is a suitable criterion taken from information theory, and for standard GP regression or classification it corresponds to selecting the point with the largest predictive variance [Rasmussen and Williams, 2006, Sec. 8.3.3]. For the preference model from Sec. 2.3.2, the posterior Shannon entropy criterion is considered by Houlsby et al. [2011, 2012], which obtain an approximate expression for the intractable expected posterior entropy change.

In the present work, the generalization performance of the GP model is not the key motivation for applying active learning. Conceptually, improving the predictive performance of the latent function in input regions that turn out to be less preferred than other regions is suboptimal. Thus, an active learning criterion that emphasizes *global optimization* [Jones, 2001, Rasmussen and Williams, 2006] instead of *generalization* is favorable. A criterion which utilizes the probabilistic GP framework is *Expected Improvement* (EI) [Jones, 2001], although EI in its definition utilizes only the predictive mean and variance of the latent function values—not the covariance. In Sec. 3.2, a *bi-variate* version of EI (bEI) is proposed which benefits from utilizing also the covariance of the GP predictions. Finally, the sequential design approach adopted in the present work is described in Sec. 3.3.

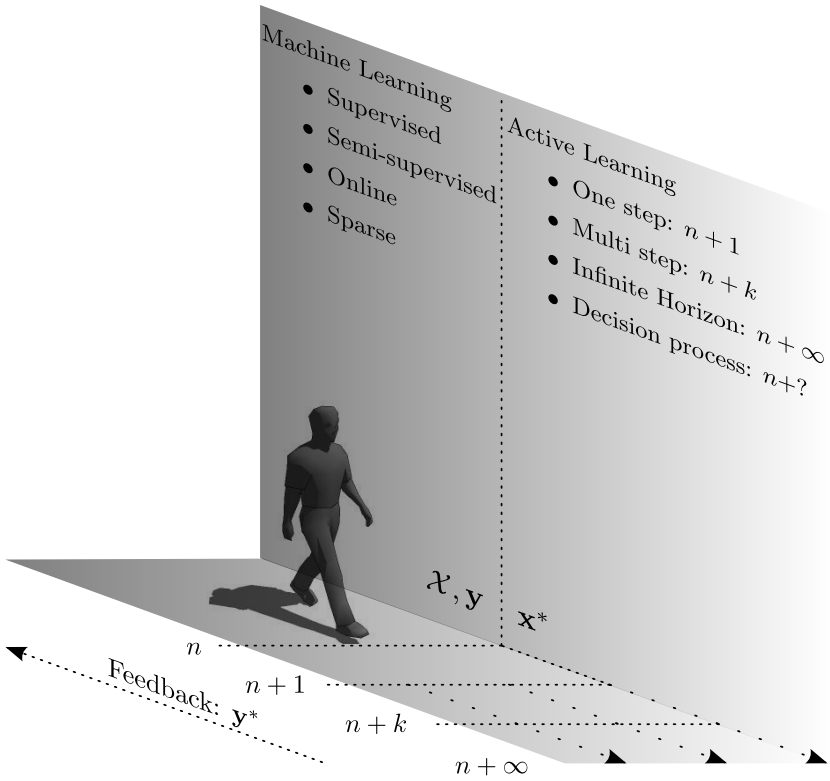


Figure 3.1: Conceptual overview of machine learning and active learning. Given a set of n observations, machine learning is concerned with the observations, $\mathbf{y}$, and the input set, $\mathcal{X}$, in order to obtain the best predictive performance at new inputs, $\mathbf{x}^*$. Active learning on the other hand is concerned with querying new observations, $\mathbf{y}^*$, *actively*, to refine the current model in order to improve predictions. Thus generally speaking, machine learning is used for all the *currently observed* data up to instance n, where active learning is used for unseen data *after* instance n.

3.2 Expected Improvement

In the remainder, the random variable $\hat{f}$ denotes the function value of the input $\mathbf{x}_i \in \mathcal{X}$, that given the current GP model has the largest predicted mean value denoted by $\hat{\mu}$. For each possible query, $\mathbf{x}_l^* \in \mathbb{R}^D$, with a corresponding function value, f_l^*, modeled by a GP, the *Improvement*, I_l, is simply given following Jones

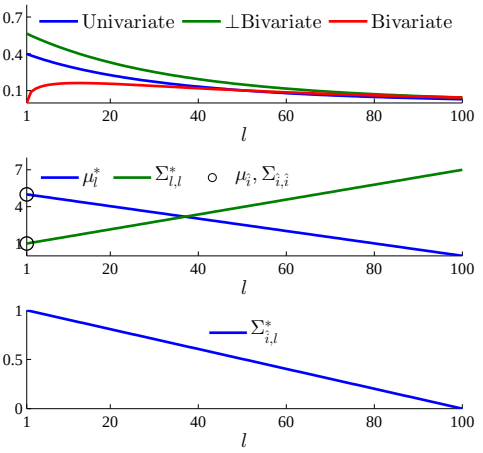


Figure 3.2: Illustration of the difference between the uni-variate and bi-variate EI. The current maximum indicated by a circle is at $l = 1$, which is also a possible query. **Top**: EI for the standard (uni-variate) version, a bi-variate version neglecting covariance (⊥Bivariate) and the bi-variate version incorporating covariance (Bivariate). **Middle**: mean and variance of queries. **Bottom**: covariance used for the bi-variate EI between query $\mathbf{x}_l^*$ and maximum at $l = 1$. [Fig. 4, Pap. H]

[2001] as

$$I_l \equiv f_l^* - \hat{f}. \tag{3.1}$$

Now, the *Expected* Improvement (EI) is given by [Jones, 2001]

$$EI_l \equiv \mathbb{E}_{p(I_l)}\{\max(I_l, 0)\} = \int_0^\infty I_l p(I_l) dI_l, \tag{3.2}$$

where $I_l \sim \mathcal{N}\left(\mu_l^* - \hat{\mu}, \Sigma_{l,l}^*\right)$ when $\hat{f}$ is *not* considered to be stochastic, which is the case in traditional (uni-variate) EI (uEI) [Jones, 2001]. The integration corresponds to the expected value of a single-sided truncated normal distribution and is given by

$$EI_l = \mu_{I_l}\Phi\left(\frac{\mu_{I_l}}{\sigma_{I_l}}\right) + \sigma_{I_l}\mathcal{N}\left(\frac{\mu_{I_l}}{\sigma_{I_l}}\middle| 0, 1\right), \tag{3.3}$$

where μ_{I_l} and σ_{I_l} are the mean and standard deviation of the normal distributed random variable I_l implying $I_l \sim \mathcal{N}\left(\mu_{I_l}, \sigma_{I_l}^2\right)$. One problem with the uEI is that it neglects that queries, $\mathbf{x}_l^*$, close to the location of the maximum, $\hat{\mathbf{x}}$, have correlated function values (non-zero covariance) under the predictive distribution of the GP. Consequently, the uEI is often maximized by points arbitrarily close to the maximum. It is by no means complicated to include the covariance into the EI framework to obtain a *bi-variate* EI (bEI) version proposed in Pap. D and later used in Pap. G and Pap. H. Given the predictive distribution of $[\hat{f}, f_l^*]^\top$ available from the GP, the improvement, I_l, is the difference between two dependent normal variables, thus

$$\mu_{I_l} = \mu_l^* - \hat{\mu} = \mu_l^* - \mu_{\hat{i}}^* \tag{3.4}$$

$$\sigma_{I_l} = \sqrt{\Sigma_{\hat{i},\hat{i}}^* + \Sigma_{l,l}^* - 2\Sigma_{\hat{i},l}^*}. \tag{3.5}$$

The difference between the uni-variate and bi-variate EI is illustrated in Fig. 3.2. An important observation is that the bEI goes towards zero as the function values become "identical". In the GP framework, this corresponds to the case in which the query, $\mathbf{x}_l^*$, and maximum, $\mathbf{x}_{\hat{i}}$, is co-located. When the next observation is obtained by maximizing the bEI, querying the current maximum point over and over again is avoided. This is not guaranteed with the uEI version.

Note, that although in the present work the current maximum point, $\mathbf{x}_{\hat{i}}$, is restricted to be in the set of the inputs for which an observation has been made ($\mathbf{x}_{\hat{i}} \in \mathcal{X}$), the previous derivation applies also if $\mathbf{x}_{\hat{i}} \in \mathbb{R}^D$.

3.3 Sequential Design Approaches

A fundamental concept in active learning targeting global optimization is the inherent trade-off between *exploration* and *exploitation*. Exploration refers to exploring regions in which the current model is uncertain, whereas exploitation refers to exploiting the current model to query regions with large predicted values. Hence, the former effectively reduces model variance whereas the latter exploits model mean. The *upper confidence bound* (UCB) [Auer, 2002] addresses this trade off directly. In a traditional GP regression setting, Srinivas et al. [2010] give exact regret bounds and show comparable regret between the uEI and UCB on both artificial and real-world examples.

In the present work, the sequential design builds on the bEI criterion, although the next observation is not directly obtained by maximizing the bEI from Sec. 3.2. Instead, a heuristic which effectively applies slightly more emphasis on exploration compared to pure maximization is used. The used heuristic is based on sampling the next observation from a multinomial distribution, where the probability of querying point $\mathbf{x}_l^*$ is equal to the corresponding bEI at the point. This approach reads somewhat similar to *Thompson sampling* [Thompson, 1933], which has recently been revisited by Chapelle and Li [2011][1]. In practice, though, the multinomial heuristic requires that the bEI is computed for every possible query, $\mathbf{x}_l^* \in \mathcal{X}^*$, assigned typically to a uniform grid over a restricted region of $\mathbb{R}^D$. For even a few number of input dimensions ($D \geq 3$), computing the bEI (or even the uEI) in a reasonable grid is infeasible making the multinomial heuristic inapplicable. In high dimensions ($D > 3$), the bEI is therefore maximized with a gradient ascent method, which in its nature does not generally converge to the global maximum of the bEI (hyper-)surface. By

[1] One possible implementation of Thompson sampling for global maximization with GPs would be to draw a function from the predictive distribution of the GP and query the input with the largest function value from the draw.

running a finite number of gradient ascent methods (say 5) with random initialization, the resulting heuristic inherently applies more emphasis on exploration compared to maximizing the bEI globally. The multinomial heuristic is applied to two-dimensional problems in Pap. D and H, whereas the multi-start gradient ascent method is applied in Pap. G and Pap. H. The derivatives required for the gradient ascent method are found in App. I.1.

Chapter 4

Discussion and Conclusion

In this chapter, perspectives and potential of the present work are discussed in Sec. 4.1 based on results from the papers included in the present thesis [Pap. A-H], but also from the experience gained during the present work. Sec. 4.2 contains the conclusion of the present work.

The reader is encouraged to read the included papers [Pap. A-H] before reading this chapter.

4.1 Discussion

The present work takes us somewhat down the road in obtaining an efficient interactive personalization system with a simple and robust pairwise interface proposed in Pap. A for especially HA fine-tuning. The system is intuitive for users to use and does not impose an inappropriate cognitive load on the users. In Pap. G and H, the system is used to efficiently optimize two and four parameters (input dimensions) of a pair of HAs to individual users in a real-world music context. The results in Pap. G and H thus confirm the apparent promising performance of the system in a challenging real-world speech-in-noise context reported in Pap. D. Hence, the proposed personalization system can potentially have a significant impact on how devices are personalized in the future. Nevertheless, several open questions exist, which have not or have only partly been addressed in the present work. This chapter serves to highlight some key fields of future research in this regard.

Currently, the personalization system shows to perform fine-tuning based on the user's perception for up to four HA parameters for which ten to twenty user assessments are required for the system to converge to an optimal setting [Pap. H]. Realistically, devices—and especially HAs—easily have more than four tunable parameters suitable for personalization. Therefore, it is important to investigate to a greater extend the *scaling* between the number of required assessments and the number of input dimensions. Similarly, if and when the modeling framework fails to model the user's IRF shall be studied. The results in Pap. H do not show a large scaling difference between two and four-dimensional problems within the same *fixed* context. Thus probably, scaling problems occur somewhat above four dimensions in a fixed context. Currently, no interaction between input dimensions is assumed a priori in the GP prior. For audio devices, multiple parameters typically arise due to a particular number of frequency bands or bins, in which similar sets of parameters across bands or bins control the device output. In these cases, it might for instance be fair to assume adjacent parameters to be correlated with respect to the user's IRF, f. Such a priori assumptions about interactions between input dimensions can be introduced via the input-covariance matrix, $\mathbf{L}$, in Eq. 2.21 and may reduce the effective number of assessments required to converge for multi-parameter devices. With a given parametrization of the input-covariance matrix, $\mathbf{L}$—for instance a low-rank factorization—the corresponding parameters may be learned by standard ML-II or MAP-II optimization.

Another important aspect of the present work in relation to a realistic application is the context, which refers to the external stimuli for which the devices are personalized. In the present work, it is assumed throughout, that the used stimulus is representative for the context in general, hence for a music context

one minute of a single music track is representative for music in general. Even if the used music track is chosen carefully such that it for instance is both dynamic and broadband, it remains an open question if the personalized setting generalizes to other stimuli within the same context. This is an aspect that requires further investigation in the future. The same considerations apply for other contexts such as for instance quiet speech, speech-in-noise etc. A more realistic approach is to randomly select a stimulus from the same context for each new assessment. This introduces noise on each assessment, but an optimal setting found in this manner will generalize better for the particular context. The Bayesian framework should be able to deal with the added noise, although the number of assessments required to converge can grow significantly because a possibly large amount of noise must be averaged out. Hence, it should be investigated if the Bayesian framework can obtain a proper optimal setting under these conditions and how these conditions influence the number of required assessments.

The present work may be applicable in a clinical setting for fine-tuning suitable parameters in HAs either physically in the clinic or remotely in a home fine-tuning scenario. For this, the above considerations apply indeed, but in addition the behavior of the user will have a prominent role during fine-tuning. In the experiments conducted during the present work, different behavioral effects have been observed. First of all, the behavior of the test subjects during the experiments indicates a learning effect, where users tent to spent more time assessing the first few assessments compared to the following assessments. Likewise, they seem to be less consistent in their judgment of the first few assessments. Generally, these observations are not quantitatively supported significantly in the experimental results. The first study (two-dimensional optimization) from Pap. H supports the learning-effect hypothesis somewhat, but not enough to express anything conclusive considering that the other experiments do not support the hypothesis. Nevertheless, the clear impression while observing the subjects conducting the experiments is that, overall, subjects tent to get more consistent during each experiment and from experiment to experiment. It remains unknown, whether or not it is simply the Bayesian framework that is more or less unaffected in the current setting by the extra observational noise, i.e. user inconsistency. Secondly, a few subjects were seen to get bored and thus distracted during the entire experimental procedure. However, this is not supported quantitatively by the results either. All the HA experiments were conducted without informing the subjects that they were actually personalizing the HAs to them selves. Subjects were only informed to assess each comparison based on their perception and personal opinion. The reason for this is to avoid any bias effects—either positive or negative—from subjects knowing that they were fine-tuning certain parameters of the current HAs to them selves. In practice, users will be aware that they are optimizing or fine-tuning HAs and will have worn the actual HAs for a longer period of time prior to the fine-tuning

procedure. Thereby, the learning effect can be less prominent in this scenario, although the effect will most certainly still be present. Furthermore, the typical HA user is most likely inspired and motivated by knowing that he/she is directly influencing the fine-tuning of the HAs [Dillon et al., 2006, Convery et al., 2011] as to get them exactly personalized to his/her individual needs. This is supported by comments from the test subjects when told after the experiments that they had actually been optimizing the HAs. Hence, in a real-world scenario users may be more motivated to listen carefully without getting bored and distracted in order to be as consistent as possible.

In the far future, assessments from a vast number of users can become available. Thereby an opportunity to exploit CF or MT learning in the modeling framework of individual user's IRF will be present. As mentioned in Sec. 2.1.2 different possible routes towards CF or MT learning within the GP framework already exist, which are worth studying if such amounts of data are available. Considering the size of a data set containing multiple user assessments, the number of inputs easily exceeds the number of inputs ($n > 1000$) that can be handle in a full GP model. Consequently, sparse representations [Sec. 2.1.4, Pap. B & C] are required in the GP model to overcome computational issues. Other algorithms more suitable for large amounts of data will be worth studying as well.

In the present work, the focus is mainly on the HA application, but as mentioned in Chapter 1 the framework applies to other areas as well, where other assessment types, such as direct bounded scaling [Pap. E & F], may be preferable.

4.2 Conclusion

In the present work, a machine-learning based interactive personalization system is proposed and developed to constitute efficient optimization of devices—in particular of HAs—driven directly on user feedback that closely reflects the user's perception. Special observational models [Pap. A & E] are proposed and implemented in a Gaussian process Bayesian optimization framework. The framework includes an active-learning criterion referred to as bi-variate Expected Improvement [Pap. D, G & H], that unlike standard EI exploits the full predictive distribution of the GP framework.

On a synthetic example, the developed bounded-and-continuous pairwise likelihood [Pap. A] embedded in the personalization system, is shown to reduce the number of assessments required to learn the user's IRF compared to state-of-the-art—also under adverse noise conditions.

Real-world experiments show that the personalization system obtains personalized settings of devices for individual users with only few user assessments and without imposing an inappropriately high cognitive load on the users for each assessment [Pap. D, G & H]. Generally, settings obtained with the personalization system are significantly preferred by the users over settings given by a current state-of-the-art prescriptive method [Pap. G & H].

Appendix A

Efficient Preference Learning with Pairwise Continuous Observations and Gaussian Processes

Bjørn Sand Jensen, Jens Brehm Nielsen, Jan Larsen. Efficient Preference Learning with Pairwise Continuous Observations and Gaussian Processes. Published in *2011 IEEE International Workshop on Machine Learning for Signal Processing*, 2011, ISSN 1551-2541. doi:10.1109/MLSP.2011.6064616.

Errata

- The second sentence of Section 3.1 should be "... and $\mathbf{A}$ is the Hessian of the negative log posterior at the mode."
- Eq. (6) should be

$$\mathbf{f}^{new} = (\mathbf{K}^{-1} + \mathbf{W} - \lambda\mathbf{I})^{-1}[(\mathbf{W} - \lambda\mathbf{I})\mathbf{f} + \nabla \log p(\mathcal{Y}|\mathbf{f}, \mathcal{X}, \boldsymbol{\theta}_{\mathcal{L}})]$$

- The equation before Eq. (13) should be

$$\mathbf{K}^* = \begin{bmatrix} \mathbf{K}^*_{rr} & \mathbf{K}^*_{rs} \\ \mathbf{K}^*_{sr} & \mathbf{K}^*_{ss} \end{bmatrix} = \mathbf{K}_t - \mathbf{k}_t^\top (\mathbf{I} + \mathbf{W}\mathbf{K})^{-1}\mathbf{W}\mathbf{k_t}$$

Efficient Preference Learning with Pairwise Continuous Observations and Gaussian Processes

Bjørn S. Jensen[1], Jens B. Nielsen[1,2] & Jan Larsen[1]

[1]Technical University of Denmark,
Department of Informatics and Mathematical Modeling,
Richard Petersens Plads B321, DK-2800 Lyngby,

[2]Widex A/S, Nymøllevej 6, DK-3540 Lynge

Preprint

Abstract

Human preferences can effectively be elicited using pairwise comparisons and in this paper current state-of-the-art based on binary decisions is extended by a new paradigm which allows subjects to convey their degree of preference as a continuous but bounded response. For this purpose, a novel Beta-type likelihood is proposed and applied in a Bayesian regression framework using Gaussian Process priors. Posterior estimation and inference is performed using a Laplace approximation.

The potential of the paradigm is demonstrated and discussed in terms of learning rates and robustness by evaluating the predictive performance under various noise conditions on a synthetic dataset. It is demonstrated that the learning rate of the novel paradigm is not only faster under ideal conditions, where continuous responses are naturally more informative than binary decisions, but also under adverse conditions where it seemingly preserves the robustness of the binary paradigm, suggesting that the new paradigm is robust to human inconsistency.

Pairwise Comparisons, Continuous Response, Gaussian Processes, Laplace Approximation

1 Introduction

Traditionally, various aspects of human perception and cognition are assumed to be related to absolute psychological magnitudes or intensities. This includes the classical findings by Weber, Fechner and Stevens who, for example, investigated the perception of light intensity. However, recently Lockhead [1] has argued that every aspect of perception is relative, even those apparently absolute aspects investigated by Weber, Fechner and Stevens. In accordance with the theory in [1], we investigate human perception from a relative viewpoint and examine one such highly relative aspect, namely preference.

Formal treatment of relative aspects goes back to the ideas of Thurnstone [2] and the principle of comparative judgments. In the present context it was revisited by Chu *et al.* [3] who formulated a Bayesian approach to preference learning using Gaussian Process (GP) priors. This formulation has initiated a number of related studies and applications, such as audiological preference [4], multi-subject food preference [5] and an extension for semi-supervised, active learning settings [6].

In this work we extend the likelihood model in [3] to support observations which in effect measure *the perceived degree to which one option is preferred over another*. This degree of preference can be obtained from a traditional paired comparison test, which implies that a subject is asked to give a subjective assessment of the degree to whether A or B is preferred over the other. Specifically, we model the observed degrees of preferences through a likelihood conditioned on a functional value difference and support inconsistent observations by applying a re-parameterized Beta distribution.

In a traditional setting, users would not be trusted to be able to quantify such an abstract and difficult aspect as degree of preference. Instead, we would rely on massive repetitions of a standard binary experiment to estimate the proportion of $A \succ B$ using this as an expression of the degree of any preferences. However, we want to exploit the extra information from continuous responses to get a faster method for preference elicitation without jeopardizing the robustness from standard binary responses. The hypothesis is that we are able to learn faster by (indirectly) observing the perceived probability of $A \succ B$ as opposed to a binary decision. Applying appropriate priors and noise modeling should ensure this to be true also under adverse conditions.

In order to examine this hypothesis, we apply the novel likelihood in a flexible Bayesian setup similar to [3] in which the prior on the underlying preference function is defined by a GP with a potentially complex covariance structure. The Laplace approximation is used for inference and model selection by *maximum-a-posteriori* (MAP) estimates. This provides a consistent probabilistic framework for making predictions and evaluating the predictive uncertainty. We use simulations with different synthetic noise scenarios in order to compare a standard binary decision with the novel model. The performance of both methods is evaluated using the predictive performance.

2 Models for Pairwise Observations

In the previous section, we motivated pairwise comparisons from a cognitive perspective, yet pairwise comparisons can be considered more broadly. It is usually possible to describe any aspect of a pairwise comparison, such as preference, real difference, or perceived similarity in terms of a latent function [2].

In the following we will model the preference of two distinct inputs, $u \in \mathcal{X}$ and $v \in \mathcal{X}$, in terms of the difference between two functional values, $f(u)$ and $f(v)$. This implies a function, $f : \mathcal{X} \to \mathbb{R}$, which defines an internal, but latent absolute preference.

The general setup is as follows: We consider n distinct inputs $x_i \in \mathcal{X}$ denoted $\mathcal{X} = \{x_i | i = 1, ..., n\}$, and a set of m responses on pairwise comparisons between any two inputs in $\mathcal{X}$, denoted by

$$\mathcal{Y} \quad = \quad \{(y_k; u_k, v_k) | k = 1, ..., m\},$$

where $y_k \in \mathbb{Y}$. $u_k \in \mathcal{X}$ and $v_k \in \mathcal{X}$ are option one and two in the k'th pairwise comparison, respectively. The main topic of this paper is how the domain of the response variable influences the learning rate of the latent function f in relation to the number of paired comparisons. As previously indicated, we will consider two cases:

- **binary** where $y_k = d_k, d_k \in \{-1, 1\}$
- **continuous and bounded** where $y_k = \pi_k, \pi_k \in]0, 1[$.

In both cases we consider y a stochastic variable, informally implying the definition of the conditional density given by $p\left(y_k | f_k\left(u_k\right), f\left(v_k\right)\right)$, denoted by $p(y_k|\mathbf{f}_k)$ with $\mathbf{f}_k = \left[f\left(u_k\right), f\left(v_k\right)\right]^\top$.

2.1 Binary Response

When restricting the response variable to be a discrete, two-alternatives, forced choice, paired-comparison between the two presented options, we define the response variable as $d_k \in \{-1, 1\}$. A preference for either u_k or v_k is indicated by -1 or $+1$, respectively.

When considering noise on the forced decisions the resulting random variable can be modeled by a classic choice model such as the Logit or Probit [7, chapter 6]. In the current setting we restrict ourself to the Probit model mainly for analytical reasons.

Given a function, f, we can define the likelihood of observing a discrete choice d_k directly as the conditional density.

$$p\left(d_k | \mathbf{f}_k, \boldsymbol{\theta}_{\mathcal{L}}\right) = \Phi\left(d_k \frac{f\left(v_k\right) - f\left(u_k\right)}{\sqrt{2}\sigma}\right), \tag{1}$$

where $\Phi(x)$ is the cumulative Gaussian (with zero mean and unity variance) and $\boldsymbol{\theta}_{\mathcal{L}} = \{\sigma\}$. This classic Probit likelihood is by no means a new invention and can be dated back to Thurstone and his fundamental definition of *The Law of Comparative Judgment*[2]. However, it was first considered with GPs in [3] and later in e.g. [5] and [6].

2.2 Continuous Response

The primary contribution of this paper is a novel response model allowing for more subtle judgments, where the response variable describes the degree to which the prevailing option is preferred.

For this purpose we formally define a continuous but bounded response $\pi \in]0; 1[$ observed when comparing u and v. The first option, u, is preferred for $\pi < 0.5$. The second option, v, is preferred for $\pi > 0.5$ and none is preferred for $\pi = 0.5$. Hence, the response captures both the choice between u and v, and the degree of the preference.

Instead of using the Probit function directly as the choice model, it is used as a link function mapping from functional differences to continues bounded responses. More precisely, the Probit is used as a mean function for a Beta type distribution with parameterized shape parameters α and β, thus

$$p\left(\pi_k | \mathbf{f}_k\right) = \text{Beta}\left(\pi_k | \alpha(\mathbf{f}_k), \beta(\mathbf{f}_k)\right).$$

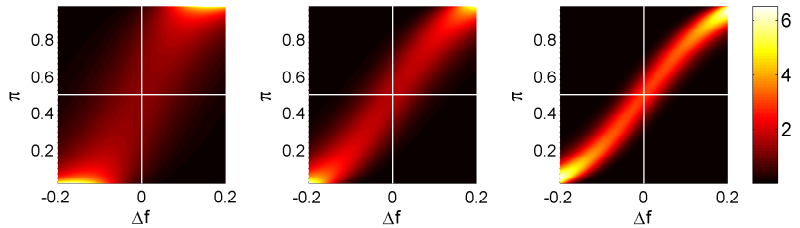


Figure 1: Illustration of the proposed likelihood with $p(\pi_k|\mathbf{f}_k, \boldsymbol{\theta}_\mathcal{L})$ shown as a color level. The likelihood parameters $\boldsymbol{\theta}_\mathcal{L}$ are $\sigma = 0.1$ and left: $\nu = 3$, middle: $\nu = 10$ and right: $\nu = 30$

To express the shape parameters of the Beta distribution as a function of the Probit mean function $\mu(\mathbf{f}_k)$, we apply a well-known re-parametrization of the Beta distribution [8].

$$\alpha(\mathbf{f}_k) = \nu\mu(\mathbf{f}_k), \qquad \beta(\mathbf{f}_k) = \nu(1 - \mu(\mathbf{f}_k)), \tag{2}$$

where ν relates to the precision of the Beta distribution and is not parameterized by f. Finally, our novel likelihood depicted in Fig. 1 is described by

$$p\left(\pi_k|\mathbf{f}_k, \boldsymbol{\theta}_\mathcal{L}\right) = \text{Beta}\left(\pi_k|\, \nu\mu(\mathbf{f}_k, \sigma), \nu(1 - \mu(\mathbf{f}_k, \sigma))\right), \tag{3}$$

where $\boldsymbol{\theta}_\mathcal{L} = \{\sigma, \nu\}$ and $\mu(\mathbf{f}_k, \sigma)$ is given by

$$\mu\left(\mathbf{f}_k, \sigma\right) = \Phi\left(\frac{f\left(v_k\right) - f\left(u_k\right)}{\sqrt{2}\sigma}\right).$$

The precision term ν in Eq. (2) and Eq. (3) is inversely related to the observation noise on the continuous bounded responses. In general, ν can be viewed as a measure of how consistent the scale is used in a given comparison.

2.3 Gaussian Process Priors

At this point we have not specified any form, order or shape of f, but referred to f as an abstract function. We maintain the abstraction by considering a non-parametric approach and use a Gaussian process (GP) to formulate our beliefs about f.

A GP is typically defined as "a collection of random variables, any finite number of which have a joint Gaussian distribution" [9]. Following [9] we denote a function drawn from a GP as $f\left(x\right) \sim \mathcal{GP}\left(\mathbf{0}, k(\cdot,\cdot)_{\boldsymbol{\theta}_c}\right)$ with a zero mean function, and $k(\cdot,\cdot)_{\boldsymbol{\theta}_c}$ referring to the covariance function with hyper-parameters $\boldsymbol{\theta}_c$, which defines the covariance between the random variables as a function of the inputs $\mathcal{X}$. The fundamental consequence of this formulation is that the GP can be considered a distribution over functions, i.e., $p\left(\mathbf{f}|\mathcal{X}, \boldsymbol{\theta}_c\right)$, with hyper-parameters $\boldsymbol{\theta}_c$ and $\mathbf{f} = [f(x_1), f(x_2), ..., f(x_n)]^T$, i.e., dependent on $\mathcal{X}$.

In a Bayesian setting we can directly place the GP as a prior on the function defining the likelihood. This leads us directly to a formulation given Bayes relation with $\boldsymbol{\theta} = \{\boldsymbol{\theta}_\mathcal{L}, \boldsymbol{\theta}_c\}$

$$p\left(\mathbf{f}|\mathcal{Y}, \mathcal{X}, \boldsymbol{\theta}\right) = \frac{p\left(\mathcal{Y}|\mathbf{f}, \boldsymbol{\theta}_\mathcal{L}\right) p(\mathbf{f}|\mathcal{X}, \boldsymbol{\theta}_c)}{p\left(\mathcal{Y}|\boldsymbol{\theta}, \mathcal{X}\right)}. \tag{4}$$

The prior $p(\mathbf{f}|\mathcal{X}, \boldsymbol{\theta}_c)$ is given by the GP and the likelihood $p(\mathcal{Y}|\mathbf{f}, \boldsymbol{\theta}_\mathcal{L})$ is either of the two likelihoods defined previously, with the assumption that the likelihood factorizes as usual, i.e., $p(\mathcal{Y}|\mathbf{f}, \boldsymbol{\theta}_\mathcal{L}) = \prod_{k=1:m} p(y_k|f(u_k), f(v_k), \boldsymbol{\theta}_\mathcal{L})$

The posterior of interest, $p(\mathbf{f}|\mathcal{Y}, \mathcal{X}, \boldsymbol{\theta})$, is directly defined when equipped with the likelihood and the prior, but it is unfortunately not of any known analytical form in either the binary nor the continuous case.

3 Inference & Predictions

Since the likelihoods considered in this paper do not result in closed form solutions to the posterior in Eq. (4), we must resort to approximations, such as the Laplace approximation, Expectation Propagation or sampling. Since the main focus of this work is to examine the general properties of the likelihood proposed in Sec. 2.2, we use the well-know and relatively simple Laplace approximation. The required steps have previously been derived for the binary likelihood [3] (see [10] for a detailed derivation), and in the following it will be derived for the proposed likelihood from Sec. 2.2.

3.1 Laplace Approximation

The main idea is to approximate the posterior by a single Gaussian distribution, such that $p(\mathbf{f}|\mathcal{Y}) \approx \mathcal{N}(\mathbf{f}|\hat{\mathbf{f}}, \mathbf{A}^{-1})$. Where $\hat{\mathbf{f}}$ is the mode of the posterior and $\mathbf{A}$ is the Hessian of the negative log posterior at the mode. The mode is found as $\hat{\mathbf{f}} = \arg\max_\mathbf{f} p(\mathbf{f}|\mathcal{Y}) = \arg\max_\mathbf{f} p(\mathcal{Y}|\mathbf{f})\, p(\mathbf{f})$.

The general solution to the problem can be found by considering the unnormalized log-posterior and the resulting cost function which is to be maximized, is given by

$$\psi(\mathbf{f}|\mathcal{Y}, \mathcal{X}, \boldsymbol{\theta}) = \log p(\mathcal{Y}|\mathbf{f}, \mathcal{X}, \boldsymbol{\theta}_\mathcal{L}) - \frac{1}{2}\mathbf{f}^T\mathbf{K}^{-1}\mathbf{f} - \frac{1}{2}\log|\mathbf{K}| - \frac{N}{2}\log 2\pi. \tag{5}$$

where $\mathbf{K}_{i,j} = k(x_i, x_j)_{\boldsymbol{\theta}_c}$. We use a damped Newton method with soft linesearch to maximize Eq. (5). In our case the basic damped Newton step (with adaptive damping factor λ) can be calculated without inversion of the Hessian (see [10])

$$\mathbf{f}^{new} = \left(\mathbf{K}^{-1} + \mathbf{W} - \lambda\mathbf{I}\right)^{-1}\left[(\mathbf{W} - \lambda\mathbf{I})\,\mathbf{f} + \nabla \log p(\mathcal{Y}|\mathbf{f}, \mathcal{X}, \boldsymbol{\theta}_\mathcal{L})\right], \tag{6}$$

Using the notation $\nabla\nabla_{i,j} = \frac{\partial^2}{\partial f(x_i)\partial f(x_j)}$ we apply the definition $\mathbf{W}_{i,j} = -\sum_k \nabla\nabla_{i,j} \log p(y_k|\mathbf{f}_k, \boldsymbol{\theta}_\mathcal{L})$. We note that the term $\nabla\nabla_{i,j} \log p(y_k|\mathbf{f}_k, \boldsymbol{\theta}_\mathcal{L})$ is only nonzero when both x_i and x_j occur as either v_k or u_k in $\mathbf{f}_k$. In contrast to standard binary GP classification the Hessian $\mathbf{W}$ is not diagonal, which makes the approximation slightly more involved.

When converged, the resulting approximation is

$$p(\mathbf{f}|\mathcal{Y}, \mathcal{X}, \boldsymbol{\theta}) \approx \mathcal{N}\left(\mathbf{f}|\hat{\mathbf{f}}, \left(\mathbf{W} + \mathbf{K}^{-1}\right)^{-1}\right). \tag{7}$$

In the Beta case the required two first derivatives of the likelihood are given by:

$$\begin{aligned}\nabla_i \log p(\pi_k \mid \mathbf{f}_k, \boldsymbol{\theta}_{\mathcal{L}}) &= \mathbb{I}(x_i) \cdot \nu \cdot \mathcal{N}(\mathbf{f}_k) \\ &\cdot [\log(\pi_k) - \log(1-\pi_k) - \psi(\alpha) + \psi(\beta)] \text{ and} \qquad (8)\\ \nabla\nabla_{i,j} \log p(\pi_k \mid \mathbf{f}_k, \boldsymbol{\theta}_{\mathcal{L}}) &= -\mathbb{I}(x_i)\mathbb{I}(x_j) \cdot \nu^2 \cdot \mathcal{N}(\mathbf{f}_k), \\ &\cdot \left[\mathcal{N}(\mathbf{f}_k) \cdot \left(\psi^{(1)}(\alpha) + \psi^{(1)}(\beta)\right) + \frac{f(v_k) - f(u_k)}{2\nu\sigma^2}\right. \\ &\cdot \left.(\log(\pi_k) - \log(1-\pi_k) - \psi(\alpha) + \psi(\beta))\right], \qquad (9)\end{aligned}$$

where we for convenience write α and β without the dependency on $\mathbf{f}_k$ Eq. (2). $\psi(z)$ and $\psi^{(1)}(z)$ are the digamma function of zero'th and first order, respectively, $\mathcal{N}(\mathbf{f}_k) = \mathcal{N}\left(\frac{f(v_k)-f(u_k)}{\sqrt{2}\sigma}\middle| 0, 1\right)$ and $\mathbb{I}(z)$ is an indicator function defined by

$$\mathbb{I}(z) = \begin{cases} 1 & \text{if } z = u_k \\ -1 & \text{if } z = v_k \\ 0 & \text{otherwise.} \end{cases} \qquad (10)$$

We refer to [10] for a full derivation and for the required derivatives for the binary case as first described in [3].

3.2 Hyper-parameter Estimation

So far we have simply considered the hyper-parameters $\boldsymbol{\theta} = \{\boldsymbol{\theta}_{\mathcal{L}}, \boldsymbol{\theta}_c\}$ variables on which we can condition the primary posterior, and not worried about their values or distributions. In the following, we consider the hyper-parameters random variables on which we place a prior and the full posterior would be $p(\mathbf{f}, \boldsymbol{\theta}|\mathcal{Y})$. However, since the focus in this work is $p(\mathbf{f}|\mathcal{Y}, \mathcal{X}, \boldsymbol{\theta})$ we only use the prior on $\boldsymbol{\theta}$ to make point estimates of the hyper-parameters in terms of *maximum-a-posteriori* (MAP) estimates.

We obtain the MAP estimates by iterating between the Laplace approximation with fixed hyper-parameters, i.e. finding $p(\mathbf{f}|\mathcal{Y}, \mathcal{X}, \boldsymbol{\theta}^{\text{MAP}})$, followed by a maximization step in which $\boldsymbol{\theta}^{\text{MAP}} = \arg\max_{\boldsymbol{\theta}} p(\boldsymbol{\theta}|\mathcal{Y}, \mathcal{X})$.

We first consider the standard evidence approach which seeks to optimize the marginal likelihood given by

$$p(\mathcal{Y}|\boldsymbol{\theta}, \mathcal{X}) = \int p(\mathcal{Y}|\mathbf{f}, \boldsymbol{\theta}_{\mathcal{L}}) p(\mathbf{f}|\mathcal{X}, \boldsymbol{\theta}_c) d\mathbf{f} = p(\boldsymbol{\theta}|\mathcal{Y}, \mathcal{X}) p(\mathcal{Y}|\mathcal{X}) / p(\boldsymbol{\theta}|\mathcal{X}). \qquad (11)$$

Our interest is in the posterior term, $p(\boldsymbol{\theta}|\mathcal{Y}, \mathcal{X})$, so considering Eq. (11) in terms of the log-posterior of $\boldsymbol{\theta}$ we obtain $\log p(\boldsymbol{\theta}|\mathcal{Y}, \mathcal{X}) = \log p(\boldsymbol{\theta}|\mathcal{X}) + \log p(\mathcal{Y}|\boldsymbol{\theta}, \mathcal{X}) - \log p(\mathcal{Y}|\mathcal{X})$, where $p(\boldsymbol{\theta}|\mathcal{X})$ is the prior and typical considered independent of $\mathcal{X}$. The evidence term, $\log p(\mathcal{Y}|\boldsymbol{\theta}, \mathcal{X})$, is analytical intractable in both likelihood cases, but we can approximate it using the existing Laplace approximation to obtain [10] $\log p(\mathcal{Y}|\boldsymbol{\theta}) \approx \log p(\mathcal{Y}|\hat{\mathbf{f}}, \boldsymbol{\theta}_{\mathcal{L}}) - \frac{1}{2}\hat{\mathbf{f}}^T \mathbf{K}^{-1} \hat{\mathbf{f}} - \frac{1}{2} \log |I + \mathbf{K}W|$. Now $\boldsymbol{\theta}^{\text{MAP}}$ is found by maximizing $\log p(\boldsymbol{\theta}|\mathcal{Y}, \mathcal{X})$ with respect to $\boldsymbol{\theta}$ and noting that $p(\mathcal{Y}|\mathcal{X})$ is independent of $\boldsymbol{\theta}$. We perform the optimization using a BFGS gradient method. The required derivatives and details are provided in [10].

The choice of particular priors is left for the simulations in Sec. 4, however, if $p(\boldsymbol{\theta})$ is the Uniform distribution, we obtain the traditional evidence optimization [9] as expected. It is noted that the complexity of the posterior inference is of the same order as standard GP regression described in [9].

3.3 Prediction

The main task is to estimate the latent function, f, with the end goal to do predictions of the observable variable y for a pair of test inputs $r \in \mathcal{X}_t$ and $s \in \mathcal{X}_t$. In this paper, we are especially interested in the discrete decision, i.e., whether $r \succ s$ or $s \succ r$. This can be obtained from both likelihood models, thus allowing for direct comparison of the two formulations in terms of predictive performance.

We first consider the predictive distribution of f which is required in both cases, and for notational convenience we omit the conditioning on $\mathcal{X}$ and $\mathcal{X}_t$. Given the GP, we can write the joint prior distribution between $\mathbf{f} \sim p\left(\mathbf{f}|\mathcal{Y}, \boldsymbol{\theta}^{\text{MAP}}\right)$ and the test variables $\mathbf{f}_t = \left[f\left(r\right), f\left(s\right)\right]^T$ as

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f}_t \end{bmatrix} = \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{K} & \mathbf{k}_t \\ \mathbf{k}_t^T & \mathbf{K}_t \end{bmatrix}\right), \tag{12}$$

where $\mathbf{k}_t$ is a matrix with elements $\mathbf{k}_{2,i} = k(s, x_i)_{\boldsymbol{\theta}^{\text{MAP}}_c}$ and $\mathbf{k}_{1,i} = k(r, x_i)_{\boldsymbol{\theta}^{\text{MAP}}_c}$ with x_i being a training input. The conditional $p\left(\mathbf{f}_t|\mathbf{f}\right)$ is obviously Gaussian as well and can be obtained directly from Eq. (12). The predictive distribution is given as $p\left(\mathbf{f}_t|\mathcal{Y}, \boldsymbol{\theta}^{\text{MAP}}\right) = \int p\left(\mathbf{f}_t|\mathbf{f}\right) p\left(\mathbf{f}|\mathcal{Y}, \boldsymbol{\theta}^{\text{MAP}}\right) d\mathbf{f}$. With the posterior approximated with the Gaussian from the Laplace approximation then $p\left(\mathbf{f}_t|\mathcal{Y}, \boldsymbol{\theta}^{\text{MAP}}\right)$ will be Gaussian too and is given as $\mathcal{N}\left(\mathbf{f}_t|\mu^*, \mathbf{K}^*\right)$ with $\mu^* = [\mu_r^*, \mu_s^*]^T = \mathbf{k}_t\mathbf{K}^{-1}\hat{\mathbf{f}}$ and

$$\mathbf{K}^* = \begin{bmatrix} \mathbf{K}_{rr}^* & \mathbf{K}_{rs}^* \\ \mathbf{K}_{sr}^* & \mathbf{K}_{ss}^* \end{bmatrix} = \mathbf{K}_t - \mathbf{k}_t^T\left(\mathbf{I} + \mathbf{W}\mathbf{K}\right)^{-1}\mathbf{W}\mathbf{k}_t,$$

where $\hat{\mathbf{f}}$ and $\mathbf{W}$ are obtained from Eq. (7). With the predictive distribution for $\mathbf{f}_t$, the final prediction of the observed variable is available from

$$p\left(y_t|\mathcal{Y}, \boldsymbol{\theta}^{\text{MAP}}\right) = \int p\left(y_t|\mathbf{f}_t, \boldsymbol{\theta}_{\mathcal{L}}^{\text{MAP}}\right) p\left(\mathbf{f}_t|\mathcal{Y}, \boldsymbol{\theta}^{\text{MAP}}\right) d\mathbf{f}_t \tag{13}$$

If the likelihood is an odd function, as in both our cases, the binary preference decision between r and s can be made directly from $p\left(\mathbf{f}_t|\mathcal{Y}\right)$. In contrast, evaluation of the integral in Eq. (13) is required for, e.g., soft decisions, reject options and sequential designs.

3.3.1 Binary Likelihood

If $p\left(\mathbf{f}_t|\mathcal{Y}, \boldsymbol{\theta}^{\text{MAP}}\right)$ is Gaussian and we consider the Probit likelihood, the integral in Eq. (13) can be evaluated in closed form as a modified Probit function given by [3]

$$P\left(r \succ s|\mathcal{Y}\right) = \Phi\left(\left(\mu_r^* - \mu_s^*\right)/\sigma^*\right) \tag{14}$$

with $\left(\sigma^*\right)^2 = 2\sigma^2 + \mathbf{K}_{rr}^* + \mathbf{K}_{ss}^* - \mathbf{K}_{rs}^* - \mathbf{K}_{sr}^*$

3.3.2 Continuous Likelihood

In the continuous case the observed variable, π, does not directly define the discrete observation which is the main focus of this work. However, a binary preference can be derived from the continuous likelihood via the predictive distribution over π. With the suggested likelihood and mean function in

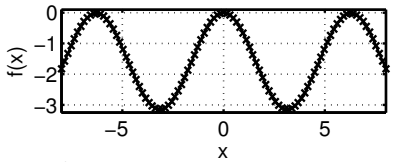


Figure 2: The Griewangk function used to evaluate the predictive performance. Crosses indicate discrete samples. The center peak is slightly higher than the two others.

Sec. 2.2 the probability of the binary choice is obtained as $P\left(r \succ s|\mathcal{Y}, \boldsymbol{\theta}_{\mathcal{L}}\right) = \int_{\pi=0}^{\pi=1/2} p\left(\pi_t|\mathcal{Y}, \boldsymbol{\theta}_{\mathcal{L}}\right)\, d\pi_t$, thus

$$P\left(r \succ s|\mathcal{Y}, \boldsymbol{\theta}^{\text{MAP}}\right) = \int p(\mathbf{f}_t|\mathcal{Y}, \boldsymbol{\theta}^{\text{MAP}})\text{Betacdf}\left(\frac{1}{2}\middle|\alpha(\mathbf{f}_t), \beta(\mathbf{f}_t)\right) d\mathbf{f}_t \tag{15}$$

In the ideal case of a noise-free user, i.e., $\nu \to \infty$, the Beta distribution reduces to a point mass at the mean defined by the Probit function. Hence, in the limit of a completely consistent user, the predictions from Eq. (15) reduces to a classical choice model with predictions that follows Eq. (14).

4 Experimental Results and Discussion

To study the performance of the models in a controlled setting, we use a synthetic dataset generated from the deterministic *Griewangk function* depicted in Fig. 2. We use the predictive performance of the binary decision to compare the learning rates of the binary response (BR) model as the baseline and the continuous bounded response (CBR) model. In each comparison, the two inputs are drawn randomly among 101 input points sampled uniformly from $x = [-8; 8]$.

The training points π_k are drawn from a Beta distribution with the parameterization from Sec. 2.2 with the Probit link function in Eq. (4), $\sigma = 1$, and the Griewangk function values as the two inputs. The noise level on the training data is defined by the parameter ν_D corresponding to ν in the CBR model. The binary decision d_k is determined by whether π_k is smaller or larger than 0.5. For evaluation, we generate an independent binary test set located equidistantly in between the training points. Initial experiments showed that in order to get a robust predictive model for all noise level, it is important to learn the ν parameter in the CBR model. The initial experiments also indicated that it is vital not to underestimate the noise, while an overestimation is not as crucial and provides overall good predictive performance. This suggests a prior with a monotonic increasing likelihood towards the highest noise level. A natural choice is a Gamma(1,η) prior with inverse scale parameter η.

The considered models, priors and parameters are listed in Table 1 where the covariance parameters, θ_c, are applied in a GP prior with a covariance function defined by the *squared exponential* kernel $k_{SE}(x, x') = \sigma_f^2 \exp(-l^{-2}\|x - x'\|^2)$. When a specific prior is not a point-mass/constant indicated by δ_x in Table 1, the hyper-parameters are estimated (MAP) either for each training set size (realistic scenario) or for $m = 500$ (ideal scenario). The latter is indicated by δ_{ideal}.

Simulation	Data Noise	$\boldsymbol{\theta}_{\mathcal{L}}$		$\boldsymbol{\theta}_c$	
	ν_D	σ	ν	σ_f	l
BR NoiseFree	No Noise	δ_1		δ_{ideal}	δ_{ideal}
BR	$\{3, 10, 30\}$	δ_1		$\mathcal{U}_1$	$\mathcal{U}_1$
CBR Noise-Free	No Noise	δ_1	$\delta_{\to\infty}$	δ_{ideal}	δ_{ideal}
CBR Ideal	$\{3, 10, 30\}$	δ_1	$\delta_{\{3,10,30\}}$	δ_{ideal}	δ_{ideal}
CBR	$\{3, 10, 30\}$	δ_1	$\mathcal{G}(1, \eta)_{\{3,10,30\}}$	$\mathcal{U}_1$	$\mathcal{U}_1$

Table 1: Simulation conditions. δ_x is a point-mass, thus the parameter is constantly equal to x. The δ_{ideal} value is learned as $m \to \infty$. $\mathcal{U}_x$ is an uniform prior over $]0; \infty[$ with the parameter initialized to x. $\mathcal{G}(1, \eta)_x$ is a Gamma prior with inverse scale parameter $\eta = 0.05$ and initialization x.

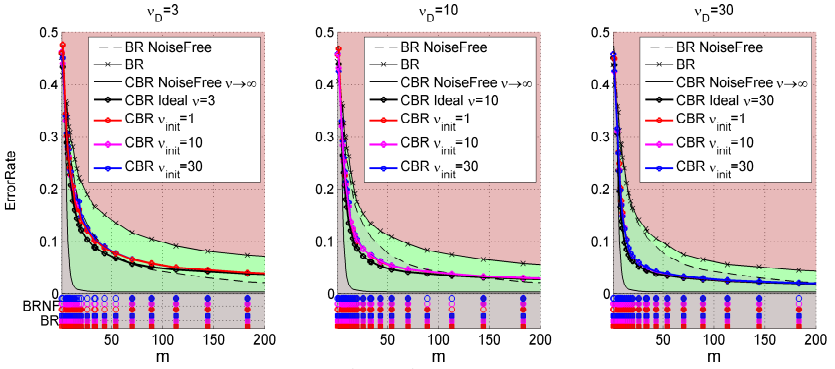


Figure 3: Mean error test rates (MER) as a function of the number of experiments over 100 different realizations of the training set generated with different ν_D. In the red and top green area MER are worse and better, respectively, than those obtained with the BR model on the noisy data. In the lower green area MER are also better than those obtained by the BR NoiseFree, and finally, the grey area corresponds to unrealistic MER better than those obtained with a CBR NoiseFree model with $\nu \to \infty$ evaluated with $\nu = 10^3$ on a noise-free data set. The six rows of markers indicate if the MER of the corresponding CBR model are significantly different from those resulting from the BR (squares) and from the BR NoiseFree (circles). If solid, the zero-hypothesis of the two means being equal is rejected at the 5% level using a paired t-test.

The learning curves from Fig. 3 show that under ideal conditions with nearly noise-free observations and a correct noise setting (Fig. 3, right plot) the CBR model outperforms the BR models as expected, since a continuous response will essentially provide more information from each experiment under ideal conditions than a binary response will. Also, in both high and moderate noise conditions (Fig. 3, left and middle plot) the CBR model with a correct noise setting (CBR Ideal) outperforms the corresponding BR model significantly in terms of learning rates and actually shows similar learning rates as the BR model under noise-free conditions. Finally and most importantly, the learning rates are only slightly lower when ν has been inferred from data via the MAP procedure

(with different initializations) than when it is specified correctly, which suggests that the parameter inference framework with independent priors is robust in real-life-scenarios without ideal model and noise conditions.

We have focused on a controlled example to highlight properties of the model and inference, leaving a real-world validation for future work. Future work also includes the extension of the mean function, Eq. (4), using a mixture of Probit functions to account for different user behavior such as centering and contraction bias. For a real-world setting, a natural extension is a suitable active learning criteria, such as the *expected value of information* framework applied recently in e.g. [5] for the BR model.

5 Conclusion and Perspectives

We have proposed a new model for preference learning with Gaussian Process priors with the main purpose to increase the learning rate compared to the standard binary model applied in [3]. We have outlined a robust and flexible inference framework for the new model based on suitable priors and the Laplace approximation. Simulations were used to present properties and performance, which showed a significant information increase from each experiment under ideal conditions as expected but more importantly also under adverse conditions. The performance is especially increased in a certain window of opportunity.

Acknowledgement: This work was supported in part by the IST Programme of the European Community, under the PASCAL2 Network of Excellence, IST-2007-216886. This publication only reflects the authors' views.

References

[1] G. R. Lockhead, "Absolute Judgments Are Relative: A Reinterpretation of Some Psychophysical Ideas.," *Review of General Psychology*, vol. 8, no. 4, pp. 265–272, 2004.

[2] L. L. Thurstone, "A law of comparative judgement.," *Psychological Review*, vol. 34, 1927.

[3] W. Chu and Z. Ghahramani, "Preference learning with Gaussian Processes," *ICML 2005 - Proceedings of the 22nd International Conference on Machine Learning*, pp. 137–144, 2005.

[4] P. Groot, T. Heskes, T. Dijkstra, and J. Kates, "Predicting preference judgments of individual normal and hearing-impaired listeners with Gaussian Processes," *IEEE Transactions on Audio, Sound, and Language Processing*, 2010.

[5] E. Bonilla, S. Guo, and S. Sanner, "Gaussian Process preference elicitation," in *Advances in Neural Information Processing Systems 23*, J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta, Eds., pp. 262–270. 2010.

[6] W. Chu and Z. Ghahramani, "Extensions of Gaussian Processes for ranking: semi-supervised and active learning," in *Workshop Learning to Rank at Advances in Neural Information Processing Systems 18*, 2005.

[7] R. D. Bock and J. V. Jones, "The measurement and prediction of judgment and choice.," 1968.

[8] S. Ferrari and F. Cribari-Neto, "Beta Regression for Modelling Rates and Proportions," *Journal of Applied Statistics*, vol. 31, no. 7, pp. 799–815, Aug. 2004.

[9] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, 2006.

[10] Bjørn Sand Jensen and Jens Brehm Nielsen, *Pairwise Judgements and Absolute Ratings with Gaussian Process Priors*, Technical Report, DTU Informatics, September 2011.

Appendix B

On Sparse Multi-Task Gaussian Process Priors for Music Preference Learning

---

Jens Brehm Nielsen, Bjørn Sand Jensen, Jan Larsen. On Sparse Multi-Task Gaussian Process Priors for Music Preference Learning. Published in *NIPS 2011 workshop on Choice Models and Preference Learning*, 2011.

On Sparse Multi-Task Gaussian Process Priors for Music Preference Learning

Jens Brehm Nielsen, Bjørn Sand Jensen and Jan Larsen
Technical University of Denmark
Department of Informatics and Mathematical Modelling
Amussens Allé, Building 305, 2800 Lyngby, Denmark,
{jenb, bjje, jl}@imm.dtu.dk

Abstract

In this paper we study pairwise preference learning in a music setting with multi-task Gaussian processes and examine the effect of sparsity in the input space as well as in the actual judgments. To introduce sparsity in the inputs, we extend a classic pairwise likelihood model to support sparse, multi-task Gaussian process priors based on the pseudo-input formulation. Sparsity in the actual pairwise judgments is potentially obtained by a sequential experimental design approach, and we discuss the combination of the sequential approach with the pseudo-input preference model. A preliminary simulation shows the performance on a real-world music preference dataset which motivates and demonstrates the potential of the sparse Gaussian process formulation for pairwise likelihoods.

1 Introduction

Preference learning is aimed at eliciting, modeling and eventually predicting human preference for a given input or normally sets of inputs. In this paper we focus on a relatively robust query type for human preference elicitation suitable for e.g. music applications, namely pairwise comparisons modeled by the likelihood function considered in [11, 1]. This basic likelihood model was first put into the flexible framework of Gaussian processes (GP) priors by Chu *et. al.* [5]. Furthermore, a general multi-task extension to the particular preference setup was proposed in Bonilla *et. al.* [3] based on the multi-task formalism originally developed by Bonilla *et. el.* [2] which supports the inclusion of collaborative or transfer learning between users. GP based models are in turn desirable models for preference learning, however, they all struggle with an inconvenient $\mathcal{O}\left(n^3\right)$ scaling in terms of the number of input instances, n, which makes their use limited for large-scale problems. A number of suggestions have been proposed to resolve this issue for the standard GP regression case.

Our objective is to extend the well-known pairwise likelihood model to allow for explicit sparsity in the input space. This is achieved by extending the pairwise likelihood model in terms of a set of pseudo-inputs (of size $l << n$) which are essentially used to integrate out the function values of the original inputs using the ideas proposed in Snelson *et. al.* [10] for the standard regression case. In effect the multi-task GP prior is now placed over the function values of the pseudo points. Posterior inference relies on a Laplace approximation, and the pseudo-inputs can be found by evidence optimization or be fixed and determined by, e.g., k-means initialization. Secondly, we outline to combine the model with the ideas of Bonilla *et. al.* [3] and include sequential experimental design to ensure that sparsity also persists in terms of the number of actual pairwise comparisons, m, besides

Revision: 2011/12/12. Acknowledgment: This work was supported in part by the IST Programme of the European Community, under the PASCAL2 Network of Excellence, IST-2007-216886. This publication only reflects the authors' views.

the sparsity in the associated number of input instances, n. Finally, we evaluate the pseudo-input model on a real-world music preference dataset, examine the multi-task transfer and learning rates and discuss limitations and further improvements of this initial evaluation.

The paper is organized as follows: In Section 2 we review the basic model, provide the pseudo-input extension and discuss option of sequential experimental design. In Section 3 we consider a toy example and present the preliminary results on the music dataset. In Section 4 we discuss the overall findings and outline a number of future research steps.

2 Model & Extensions

We describe the general setup and model in terms of

- a set $\mathcal{A}$ of n_a input instances, e.g. audio tracks, where each input instance i is described by one feature vector $x^{(a)} \in \mathbb{R}^{d_a}$, i.e., $\mathcal{A} = \{x_i^{(a)} | i = 1, ..., n_a\}$.
- a set $\mathcal{U}$ of n_u users, where each user j is described by a feature vector $x^{(u)} \in \mathbb{R}^{d_u}$, i.e., $\mathcal{U} = \{x_j^{(u)} | j = 1, ..., n_u\}$.

The task for a specific user j is to perform a forced choice between two input instances, $x_u^{(a)} \in \mathcal{A}$ and $x_v^{(a)} \in \mathcal{A}$, where $u \neq v$, resulting in a response $y \in \{-1, +1\}$, where $y = +1$ corresponds to a preference for the u'th input, and -1 corresponds to a preference for the v'th input. We acquire m such pairwise comparisons between any two input instances in $\mathcal{A}$ and with any user in $\mathcal{U}$, which results in the set of observations $\mathcal{Y} = \left\{ (y_k; x_{u_k}^{(a)}, x_{v_k}^{(a)}, j_k) | k = 1, ..., m \right\}$.

Given the two latent function values $\mathbf{f}_k = \left[f_{j_k}\left(x_{u_k}^{(a)}\right), f_{j_k}\left(x_{v_k}^{(a)}\right) \right]$ (associated with a particular user) at the two inputs, we model the observations by a likelihood function $p\left(y_k | \mathbf{f}_k, \boldsymbol{\theta}_{\mathcal{L}}\right)$. The likelihood function is defined by additional parameters $\boldsymbol{\theta}_{\mathcal{L}}$. The function f_{j_k} is an absolute, latent function preserving the preference information over the input space for a particular user j. The function parametrization admits that we directly place a Gaussian process prior on f_{j_k} allowing for a flexible predictive model for the pairwise responses of a particular user. A multi-task setting can be constructed by exploiting an observed feature vector per user. Consequently, we can think of a global latent multi-task preference function $f(x^{(a)}, x^{(u)})$ instead of several individual single-task preference functions $f_j(x^{(a)})$. The multi-task kernel formulation of a GP [2] can hence be formulated as:

$$f_j(x_i^{(a)}) = f(x_i^{(a)}, x_j^{(u)}) \sim \mathcal{GP}\left(0, k(x_i^{(a)}, \cdot) k(x_j^{(u)}, \cdot)\right) = \mathcal{GP}\left(0, k(x_{i,j}, \cdot)\right), \tag{1}$$

where we have joined the audio and user feature into one input instance, $x = \{x^{(a)}, x^{(u)}\}$, and thereby defined the unique set of inputs as $\mathcal{X} = \{\{x_i^{(a)}, x_j^{(u)}\} | i = 1...n_a, j = 1...n_u\}$. Thus, the GP framework constitutes a non-linear, yet very flexible alternative to the more traditional models such as (Generalized) Linear Models. Also, this formulation addresses the multi-task kernel only in the definition of the covariance function - everywhere else, we only think of one input x containing both user and task features simultaneously with a corresponding function value $f(x)$. This definition will be convenient later.

Given a standard Bayesian framework and assuming the likelihood factorizes we now obtain the posterior over the function, i.e.,

$$p\left(\mathbf{f} | \mathcal{X}, \mathcal{Y}, \boldsymbol{\theta}\right) \propto p\left(\mathbf{f} | \mathcal{X}, \boldsymbol{\theta}_{GP}\right) \prod_{k=1}^{m} p\left(y_k | \mathbf{f}_k, \boldsymbol{\theta}_{\mathcal{L}}\right)$$

with $\mathbf{f} = [f(x_1^{(a)}, x_1^{(u)}), f(x_1^{(a)}, x_2^{(u)}), ..., f(x_1^{(a)}, x_{n_u}^{(u)}), ..., ..., f(x_{n_a}^{(a)}, x_{n_u}^{(u)})]^\top$, $\boldsymbol{\theta}_{GP}$ contains the GP hyper-parameters and $\boldsymbol{\theta} = \{\boldsymbol{\theta}_{\mathcal{L}}, \boldsymbol{\theta}_{GP}\}$. The main computational issue in the single task GP is to calculate/approximate the posterior which poses a $\mathcal{O}\left(n_a^3\right)$ scaling challenge due to the inversion of the kernel matrix. Coupling n_u single task GPs in the covariance structure will further scale this to $\mathcal{O}([n_a n_u]^3)$. In practical preference applications, this is of course a problem and to remedy this we first consider the (standard) pairwise likelihood in Section 2.1.1 and then a sparse extension in Section 2.1.2 allowing for a sparse GP prior with less than $(n_a)(n_u)$ inputs. Finally, we suggest the sequential extension in Section 2.3.

2.1 Likelihood

2.1.1 Pairwise Likelihood (Standard)

Pairwise comparisons are typically modeled by the classic Probit choice model [11, 1], constituting the basis for the so-called pairwise likelihood function given by

$$p\left(y_k|\mathbf{f}_k,\boldsymbol{\theta}_{\mathcal{L}}\right)=\Phi\left(y_k\frac{f_{j_k}\left(x_{u_k}^{(a)}\right)-f_{j_k}\left(x_{v_k}^{(a)}\right)}{\sqrt{2}\sigma}\right), \tag{2}$$

where $\Phi(x)$ defines a cumulative Gaussian (with zero mean and unity variance), and $\boldsymbol{\theta}_{\mathcal{L}}=\{\sigma\}$. The use of a GP prior in connection with this likelihood was first proposed in [5].

2.1.2 Pairwise Likelihood with Pseudo-Inputs

We extend the standard preference model in Eq. 2 to obtain sparsity in the input space in terms of the effective number of points in the prior and posterior. We generally follow the ideas in [10], i.e., given a set of pseudo-inputs $\bar{\mathbf{X}}$, their functional values $\bar{\mathbf{f}}$ must come from a Gaussian process like the real latent data $\mathbf{f}$. Therefore, we can directly place a Gaussian process prior over $\bar{\mathbf{f}}$

$$p\left(\bar{\mathbf{f}}|\bar{\mathbf{X}}\right)=\mathcal{N}\left(\bar{\mathbf{f}}|\mathbf{0},\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}\right) \tag{3}$$

where the matrix $\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}$ is the covariance matrix of the l pseudo-inputs collected in the matrix $\bar{\mathbf{X}}=[\bar{x}_1,...,\bar{x}_l]$. Recall, that we have formulated our multi-task problem only in terms of the covariance function. Therefore, each pseudo-input $\bar{x}$ defines both a task vector $\bar{x}^{(a)}\in\mathbb{R}^{d_a}$ and a user vector $\bar{x}^{(u)}\in\mathbb{R}^{d_u}$, which are stacked to form each of the pseudo-input vectors used in $\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}$. Then the covariance matrix, $\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}$, is again found by the use of the same multi-task covariance function $k(\cdot,\cdot)$ from Eq. 1, i.e., $[\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}]_{i,j}=k(\bar{x}_i,\bar{x}_j)$[1]. The overall idea of the pseudo-input formalism is now to refine the likelihood such that the real $\mathbf{f}$ values that enter directly in the original, non-sparse likelihood function (through f_k), exist only in the form of predictions from the the pseudo-inputs $\bar{\mathbf{f}}(\bar{\mathbf{X}})$. Given the listed assumptions, we formally have that $\mathbf{f}$ and $\bar{\mathbf{f}}$ are jointly Gaussian, i.e.,

$$\begin{bmatrix}\mathbf{f}_k\\ \bar{\mathbf{f}}\end{bmatrix}=\mathcal{N}\left(\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\end{bmatrix},\begin{bmatrix}\mathbf{K}_{\mathbf{x}_k\mathbf{x}_k} & \mathbf{K}_{\bar{\mathbf{X}}\mathbf{x}_k}^{\top}\\ \mathbf{K}_{\bar{\mathbf{X}}\mathbf{x}_k} & \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}\end{bmatrix}\right), \tag{4}$$

where we define the following matrices and vectors

$$\mathbf{K}_{\mathbf{x}_k\mathbf{x}_k}=\begin{bmatrix}k(x_{u_k,j_k},x_{u_k,j_k}) & k(x_{u_k,j_k},x_{v_k,j_k})\\ k(x_{v_k,j_k},x_{u_k,j_k}) & k(x_{v_k,j_k},x_{v_k,j_k})\end{bmatrix},\mathbf{K}_{\bar{\mathbf{X}}\mathbf{x}_k}=[\mathbf{k}_{u_k},\mathbf{k}_{v_k}]$$

with $[\mathbf{k}_{u_k}]_i=k(\bar{x}_i,x_{u_k,j_k})$ and $[\mathbf{k}_{v_k}]_i=k(\bar{x}_i,x_{v_k,j_k})$. Note, that we have now formally stacked the task and user feature into one input, such that x_{u_k,j_k} and x_{v_k,j_k} contain the task feature for option u and v, respectively, together with the user feature.

From Eq. 4 it is trivial to find the conditional distribution of $\mathbf{f}_k$ given $\bar{\mathbf{f}}$, hence the likelihood can be derived in terms of $\bar{\mathbf{f}}$, i.e. $p\left(y_k|\bar{\mathbf{f}},\bar{\mathbf{X}}\right)$, by integrating over $\mathbf{f}_k$

$$p\left(y_k|x_{u_k,j_k},x_{v_k,j_k},\bar{\mathbf{X}},\bar{\mathbf{f}},\boldsymbol{\theta}\right)=\int_{\mathbf{f}_k}p\left(y_k|\mathbf{f}_k,\boldsymbol{\theta}_{\mathcal{L}}\right)p\left(\mathbf{f}_k|\bar{\mathbf{f}},\bar{\mathbf{X}}\right)d\mathbf{f}_k \tag{5}$$

$$=\int_{\mathbf{f}_k}\Phi\left(y_k\frac{f_{j_k}\left(x_{u_k}^{(a)}\right)-f_{j_k}\left(x_{v_k}^{(a)}\right)}{\sqrt{2}\sigma}\right)\mathcal{N}\left(\mathbf{f}_k|\mu_k,\boldsymbol{\Sigma}_k\right)d\mathbf{f}_k \tag{6}$$

$$=\Phi\left(y_k\frac{\mu_{u_k}-\mu_{v_k}}{\sigma_k^*}\right) \tag{7}$$

[1] Notice, that now we have introduced one more use of i and j, besides to index input and users, namely to index element of a matrix. In the following we will keep using both, but when i and j are used to index matrices and vectors, it will be clear from the notation

where $\mu_k = [\mu_{u_k}, \mu_{v_k}]^\top$, $\mu_{u_k} = \mathbf{k}_{u_k}^T \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1} \bar{\mathbf{f}}$, $\mu_{v_k} = \mathbf{k}_{v_k}^T \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1} \bar{\mathbf{f}}$ and

$$\mathbf{\Sigma}_k = \begin{bmatrix} \sigma_{u_k u_k} & \sigma_{u_k v_k} \\ \sigma_{v_k u_k} & \sigma_{v_k v_k} \end{bmatrix} = \mathbf{K}_{\mathbf{x}_k \mathbf{x}_k} - \mathbf{K}_{\bar{\mathbf{X}}\mathbf{x}_k}^\top \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1} \mathbf{K}_{\bar{\mathbf{X}}\mathbf{x}_k}$$

Furthermore, $(\sigma_k^*)^2 = 2\sigma^2 + \sigma_{u_k u_k} + \sigma_{v_k v_k} - \sigma_{u_k v_k} - \sigma_{v_k u_k}$, which all together results in the pseudo-input likelihood

$$p\left(y_k | x_{u_k,j_k}, x_{v_k,j_k}, \bar{\mathbf{X}}, \bar{\mathbf{f}}, \boldsymbol{\theta}\right) = \Phi\left(z_k\right), \quad \text{where } z_k = y_k \left(\mathbf{k}_u^T - \mathbf{k}_v^T\right) \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1} \bar{\mathbf{f}} / \sigma_k^* \tag{8}$$

2.2 Posterior - Inference & Predictions

Both likelihoods described in Section 2.1 lead to untractable posteriors and call for approximation techniques or sampling methods. Our goal in this initial study is to examine the model and its properties - not to provide the optimal approximation - and we will only explore inference based on the Laplace approximation.

2.2.1 Posterior Approximation

Inference using the Laplace approximation has also been applied in [4] for the standard model. The general solution to the approximation problem can be found by considering the unnormalized log-posterior and the resulting cost function (to be maximized) is given by

$$\psi\left(\bar{\mathbf{f}} | \mathcal{Y}, \mathcal{X}, \bar{\mathbf{X}}, \boldsymbol{\theta}\right) = \log p\left(\mathcal{Y} | \bar{\mathbf{f}}, \mathcal{X}, \bar{\mathbf{X}}, \boldsymbol{\theta}\right) - \frac{1}{2} \bar{\mathbf{f}}^T {\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}}^{-1} \bar{\mathbf{f}} - \frac{1}{2} \log |\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}| - \frac{N}{2} \log 2\pi. \tag{9}$$

where $[\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}]_{i,j} = k(x_i, x_j)_{\boldsymbol{\theta}_{\mathcal{GP}}}$. We use a damped Newton method with optional linesearch to maximize Eq. (9). The basic damped Newton step (with adaptive damping factor λ) can in this case be calculated without inversion of the Hessian (see [7])

$$\bar{\mathbf{f}}^{new} = \left(\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1} + \mathbf{W} - \lambda \mathbf{I}\right)^{-1} \left[(\mathbf{W} - \lambda \mathbf{I}) - \bar{\mathbf{f}} + \nabla \log p(\mathcal{Y} | \bar{\mathbf{f}}, \mathcal{X}, \bar{\mathbf{X}}, \boldsymbol{\theta})\right], \tag{10}$$

Using the notation $\nabla\nabla_{i,j} = \frac{\partial^2}{\partial f(x_i) \partial f(x_j)}$ we apply the definition $\mathbf{W}_{i,j} = -\sum_k \nabla\nabla_{i,j} \log p(y_k | x_{u_k,j_k}, x_{v_k,j_k}, \bar{\mathbf{X}}, \bar{\mathbf{f}}, \boldsymbol{\theta})$. When converged, the resulting approximation can be shown to be $p\left(\bar{\mathbf{f}} | \mathcal{Y}, \mathcal{X}, \bar{\mathbf{X}}, \boldsymbol{\theta}\right) \approx \mathcal{N}\left(\bar{\mathbf{f}} | \hat{\mathbf{f}}, \left(\mathbf{W} + {\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}}^{-1}\right)^{-1}\right)$. The damped Newton step requires the Jacobian and Hessian of the new pseudo-input log-likelihood, which requires the following derivatives

$$\frac{\partial}{\partial \bar{\mathbf{f}}} p\left(y_k | ...\right) = y_k \frac{\mathcal{N}\left(z_k\right)}{\sigma_k \Phi\left(z_k\right)} \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1} \left(\mathbf{k}_u - \mathbf{k}_v\right) \tag{11}$$

$$\frac{\partial^2}{\partial \bar{\mathbf{f}} \bar{\mathbf{f}}^\top} p\left(y_k | ...\right) = -y_k^2 \frac{\mathcal{N}\left(z_k\right)}{\sigma_k^2 \Phi\left(z_k\right)} \left[z_k + \frac{\mathcal{N}\left(z_k\right)}{\Phi\left(z_k\right)}\right] \cdot \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1} \left(\bar{\mathbf{k}}_u - \bar{\mathbf{k}}_v\right) \left(\bar{\mathbf{k}}_u - \bar{\mathbf{k}}_v\right)^\top \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1}. \tag{12}$$

2.2.2 Evidence / Hyperparameter Optimization

Hyperparameters are optimized based on a regularized variant of traditional evidence or maximum likelihood II (ML-II) optimization allowing for simple regularizing priors on the hyperparameters. The reguralization is primarily included for robustness and is in spirit similar to regularized EM algorithms. The details are available in [7], but for completeness we shortly review the process of evidence optimization and comments on the case of the pseudo-input model.

So far we have simply considered the hyper-parameters $\boldsymbol{\theta} = \{\boldsymbol{\theta}_{\mathcal{L}}, \boldsymbol{\theta}_{\mathcal{GP}}\}$ and pseudo-inputs $\bar{\mathbf{X}}$ as fixed paraments. However, they have a crucial influence on the model and we will resort to point estimates by iterating between the Laplace approximation with fixed hyper-parameters, i.e., finding $p\left(\bar{\mathbf{f}} | \mathcal{Y}, \mathcal{X}, \bar{\mathbf{X}}, \boldsymbol{\theta}\right)$, followed by an evidence maximization step in which $(\boldsymbol{\theta}, \bar{\mathbf{X}}) = \arg\max_{(\boldsymbol{\theta}, \bar{\mathbf{X}})} p\left(\mathcal{Y} | \boldsymbol{\theta}, \bar{\mathbf{X}}\right)$. The log-evidence, $\log p(\mathcal{Y} | \boldsymbol{\theta}, \bar{\mathbf{X}})$, has to be approximated in our case, which in terms of the existing Laplace approximation yields $\log p\left(\mathcal{Y} | \boldsymbol{\theta}, \bar{\mathbf{X}}\right) \approx \log p(\mathcal{Y} | \hat{\mathbf{f}}, \bar{\mathbf{X}}, \mathcal{X}, \boldsymbol{\theta}) - \frac{1}{2} \hat{\mathbf{f}}^T {\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}}^{-1} \hat{\mathbf{f}} - \frac{1}{2} \log |\mathbf{I} + \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}} \mathbf{W}|$. We perform the optimization step using a standard BFGS method.

The pseudo-input model poses a number of difficulties since $\bar{\mathbf{X}}$ are also to be considered hyperparameter, and the input locations can thus be optimized as outlined above. Typically, this will, as noted in [10][9], lead to a large number of local maxima providing potentially suboptimal solutions, at least when using the proposed gradient method. It is not our aim to resolve nor document this issue, and we will take a pragmatic view and simply accept evidence optimization methods as is. The pseudo-input approach can in some sense be seen as a supervised clustering of the input space, but the optimization of $\bar{\mathbf{X}}$ is heavily influences by the initializations. We recommend starting out with a fixed set of pseudo-inputs initialized by a standard unsupervised clustering, such as k-means like [9], and then attempt an evidence optimization of $\bar{\mathbf{X}}$. We will provide a demonstration of this approach.

2.2.3 Predictions

Predictions of the pairwise judgments for a new experiment $\eta = \{x_u^{(a)*}, x_v^{(a)*}, x^{(u)*}\}$ with $x_u^{(a)*} \in \mathbb{R}^{d_a}$, $x_v^{(a)*} \in \mathbb{R}^{d_a}$ and $x^{(u)*} \in \mathbb{R}^{d_u}$ is given by $p(y|\eta, \mathcal{Y}, \mathcal{X})$. Given the approximated posterior of interest, $p\left(\bar{\mathbf{f}}|\mathcal{Y}, \mathcal{X}, \boldsymbol{\theta}\right)$, the prediction can be made in closed form (see e.g. [5] in the standard case and [7] for the pseudo-input case).

2.3 Sequential Experimental Design

Sequential experiential design - also known as active learning, selective or uncertainty sampling - includes datapoints/queries in a sequential manner by selecting only the most informative experiments/instances in terms of some gain. If the gain is relevant to the task, this effectively reduces the number of real input instances, n, and the number of pairwise comparisons, m, required to obtain a certain performance level compared to random selection of datapoints. Together with the pseudo-input model proposed in Section 2.1.2 this will ensure that we obtain a sparse and close to optimal model in terms of m,n and the effective number of pseudo-inputs l. We formulate the problem as a Bayesian sequential design problem (see e.g. [8]) in terms of a gain function, $G(\cdot)$, the expectation of this gain and the currently observed data $\mathcal{D} = \{\mathcal{X}, \mathcal{Y}\}$, i.e.,

$$\eta_y = \arg\max_{\eta} \sum_{y \in \mathbb{Y}} p(y|\eta, \mathcal{D})\, G\left(y, \eta, p\left(\bar{\mathbf{f}}_{\mathcal{D}\cup\eta}|y, \eta, \mathcal{D}\right), p\left(\bar{\mathbf{f}}_{\mathcal{D}}|y, \eta, \mathcal{D}\right)\right) \tag{13}$$

If the aim is to find the instance for which the user(s) has/have highest preference, the gain can e.g. be defined as expected improvement [3]. If the aim is a generalization of the preference model for all instances and users, entropy change (reduction) is the natural choice (but not guaranteed to be optimal). The multi-task (-user) and collaborative setting does support specialized gain functions depending, e.g., on user experience, consensus and knowledge, but it is not the aim to develop such concepts here. Since the main focus of the paper is the pseudo-input formulation of the pairwise likelihood, we leave the evaluation of the sequential extension to future research, but consider it a natural part of the general sparse framework outlined.

3 Simulations & Experimental Results

3.1 Example I: Pseudo-Input in 1D

This example is primarily intended to illustrate the basics of the pseudo-input principle in the pairwise case (in a single task setting). The example is based on a deterministic function which defines the pairwise relations, specifically a cosine in $[-2\pi; 2\pi]$ illustrated at the top-left in Figure 1. The seventeen input points are distributed equidistantly throughout the interval. The pairwise dataset $\mathcal{Y}$ is then generated as a complete set of pairwise relations for all input combinations. To model this dataset, we consider three case: A standard model (Section 2.1.1), a sparse model with fixed pseudo-inputs (Section 2.1.2) and a sparse model with optimized pseudo-inputs (Section 2.1.2). The five pseudo-inputs are initialized to $\bar{\mathbf{X}} = [-5, -2, 0, 2, 5]$, i.e. not in the training set. For direct comparison between the three models, we fix the other parameters, i.e., $\boldsymbol{\theta}_{\mathcal{L}}$ and $\boldsymbol{\theta}_{\mathcal{GP}}$, and use a Squared Exponential covariance function in all three cases with variance $\sigma_f = 1$ and lengthscale $\ell = 1$. The results are presented in Figure 1.

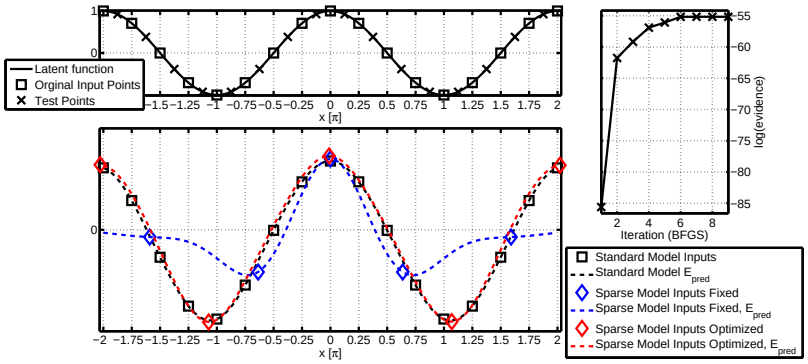


Figure 1: **Top (left)** panel shows a graph of the function from which the true underlying relations are defined. **Top (right)** panel shows the convergence of the evidence optimization. **Bottom (left)** panel shows the input points as markers used the three considered models and the predictive mean (E_{pred}) of the model as dotted graphs.

Given the equidistantly distributed input points and the full pairwise design, the standard model is almost capable of modeling the underlying function, however, the fixed model parameters limits the fit to the original model. Yet, the standard model is the best model we can expect in this case. The sparse model with fixed parameters generally has problems due to the suboptimal placement of the five pseudo-inputs. The optimized version converges to a (possible local) maximum as seen in the right panel of Figure 1 and solves the problem by moving the pseudo-inputs. This provides a better - and almost close to the standard - model despite only requiring 5 points as compared to 17.

3.2 Example II: Music Preference Data

In order to provide some initial insight into pairwise music preference learning, we consider a publicly available dataset [6]. Specifically, it consist of 10 test subjects, but only 9 with full user metadata, 30 audio tracks with 10 audio tracks per genre [2]. The genres are Classical, Heavy Metal and Rock/Pop. The design of the experiment is based on a partial version of a complete pairwise design, hence only 155 out of the 420 combinations was evaluated by each of the 10 subjects. We extract standard audio features from the audio tracks, specifically the Mel-Frequency Cepstral Coefficients, MFCCs, (26 dimensions, including delta coefficients), which we project to a 6 dimensional space using PCA. Each track is subsequently modeled by a Gaussian with mean vector, $\mu^{(a)}$, and covariance matrix, $\Sigma^{(a)}$. The feature vector is then constructed as $x^{(a)} = \left[\mu^{(a)}, diag\left(\Sigma^{(a)}\right)\right]^\top$.

We define the correlation structure of tracks by considering a general purpose covariance function for audio that easily integrates user features and metadata types for the audio, such as audio features, tags, lyrics etc. It is defined as

$$k\left(x, x'\right) = \left(\sum\nolimits_{\ell=1}^{K_a} k_\ell\left(x^{(a)}, x^{(a)\prime}\right)\right) k_u\left(x^{(u)}, x^{(u)\prime}\right), \tag{14}$$

The first factor is the sum of all the K_a covariance functions defining the correlation structure of the audio inputs, $x^{(a)}$. The second factor, or multi-task part, is a general covariance function defining the covariance function for the user metadata part, $x^{(u)}$. We include only audio features, and e.g. not tags and lyrics, thus $K_a = 1$ and apply a standard squared exponential isotropic covariance function for the audio part. The user kernel is defined by a standard squared exponential kernel between the user features (age and the three prior genre preferences) available in vector form.

[2] The small-scale nature of the dataset is not optimal, yet it has not been possibly to obtain a larger dataset containing both features (or audio) and ratings, and especially the desire to consider pairwise comparisons of music tracks seems to be a novel consideration in music preference modeling.

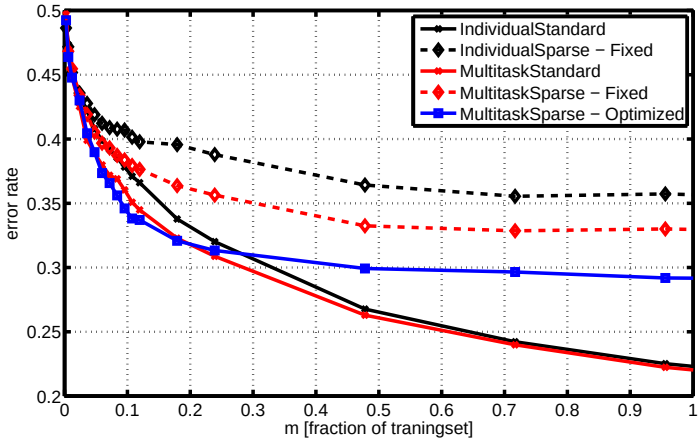


Figure 2: Learning curves averaged over 10 repetitions and 5-folds. $\bar{\mathbf{X}}$ is learned once on the full training set in each fold. A fraction of one corresponds to 80% of all comparisons. Sparse models are limited to 10% of the original number of inputs

3.2.1 Results

We concentrate on two of the most imminent questions which are the performance difference between (sparse) pseudo-input Model versus (dense) standard model, and the difference between individual modeling versus multi-task modelling.

We include a typical example of the learning curves by fixing all model parameters except the pseudo-inputs. Based on initial experiments, we fix the covariance parameters to: $\sigma^{(a)} = 3$, $\ell^{(a)} = 4$, $\sigma^{(u)} = 1.5$, and $\sigma^{(u)} = 1.5$. and the likelihood parameter, $\sigma_{\mathcal{L}} = 1$.

We consider the specific case of 27 pseudo-inputs (10% of total inputs points) in the $2 * 6 + 4 = 16$ dimensional input. This is based on a pure genre assumption, i.e., each of the nine users track preference can be described by single value pr. genre ($9 \cdot 3$). Multi-task models effectively implies more points per genre if transfer can be exploited between users. The pseudo-inputs are initialized by k-means in the full input space (all audio tracks, all user features).

To provide some insight into the generalization properties of the relatively small dataset, we use a 5-fold cross-validation (CV) scheme. In each of the five CV we use one fold as test (279 observations), and 4 fold for training (837 observations). We evaluate the learning curves for a number of training set sizes, m, by selecting a random subsets of the full set. This is done 10 times for each m.

The preliminary results presented in Fig. 2 yields a few noticeable observations. Comparing the standard multi-task versus standard individual, we observe a minor benefit in the multi-task/collaborative model versus modeling users individually, thus some (useful) transfer is present. We furthermore observe that as more and more data is observed the individual model performs almost equally well as the multi-task. This is expected and individual models will in the limit outperform a multi-task model, but the exact point at which the individual models outperforms a multi-task model is difficult to estimate beforehand.

The second point to notice is the difference between the standard multi-task and the sparse multi-task. From a m-fraction of 0.0125 the sparse model contains less points than standard model (on average) and with approximately less than a 20% of the training set, the sparse model is fully capable to compete with the standard multi-task model. After 20% of the pairwise comparisons ($m = 0.2$) approximately 80% of all real inputs points has been observed. After this point the sparse model seems to lack the flexibility to fully describe the preferences. Whether this is due to a general characterize of the music preference problem or the fixed hyperparameters is so far unexplored,

but we speculate that a full hyperparameter optimization will further minimize the gap between the sparse and the non-sparse model in this pairwise case.

The exact shape and absolute level of the learning curves are found to be sensitive to the exact prior parameters including $\bar{\mathbf{X}}$, and a robust scheme is to be derived to ensure robust and generalizable results. Despite its limitations the included case study suggests that the sparse pairwise model can provide some computation relief without scarifying all of the performance - also in the multi-task case - but there is a large number of model combinations still to be evaluated in future work.

4 Discussion & Conclusion

We derived a sparse version of the pairwise likelihood model using the pseudo-input formulation, and applied the Laplace approximation. We suggest to examine Expectation Propagation and (sequential) MCMC methods for more efficient and exact approximations. The pseudo-inputs are optimized using an evidence optimization approach which in general is challenging due to local maximum of the evidence, which is to be examined in the future. For now we rely on a "good" initialization. In the final step we suggested that the pairwise pseudo-input model should be combined with a sequential experimental design to reduce the actual number of pairwise experiments.

A synthetic example was used to show the effect of the pseudo-inputs and evidence optimization. As motivating example we presented a multi-task problem, namely a music preference problem. This typically requires a sparse approximation both in terms of input (tracks) as evaluated and in terms of the number of comparisons users have to perform, but the evaluation of the latter is considered future work on a larger dataset. We see the pseudo-input model as a useful tool in examining clustering properties of features and users in GP based preference learning, but this will probably require more elaborate inference methods and kernels.

In conclusion this workshop contribution serves primarily as a presentation of the pairwise likelihood in a pseudo-input formulation with the sequential design as an additional suggested option.

References

[1] R. D. Bock and J. V. Jones. The Measurement and Prediction of Judgment and Choice. 1968.

[2] E. Bonilla, K. Ming Chai, and C. Williams. Multi-task gaussian process prediction. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 153–160. MIT Press, Cambridge, MA, 2008.

[3] E. Bonilla, S. Guo, and S. Sanner. Gaussian Process Preference Elicitation. In J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 262–270. 2010.

[4] W. Chu and Z. Ghahramani. Extensions of Gaussian Processes for ranking: semi-supervised and active learning. In *Workshop Learning to Rank at Advances in Neural Information Processing Systems 18*, 2005.

[5] W. Chu and Z. Ghahramani. Preference Learning with Gaussian Processes. *Proceedings of the 22nd International Conference on Machine Learning (ICML)*, pages 137–144, 2005.

[6] B. S. Jensen, J. S. Gallego, and J. Larsen. A Predictive Model of Music Preference using Pairwise Comparisons - Supporting Material and Dataset. www.imm.dtu.dk/pubdb/p.php?6143.

[7] B. S. Jensen and J. B. Nielsen. Pairwise Judgements and Absolute Ratings with Gaussian Process Priors. Technical report, November 2011.

[8] D. V. Lindley. On a Measure of the Information Provided by an Experiment. *The Annals of Mathematical Statistics*, (4):986–1005, 1956.

[9] Y. Qi, A. Abdel-Gawad, and T. Minka. Sparse-posterior gaussian processes for general likelihoods. In *Proceedings of the Proceedings of the Twenty-Sixth Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-10)*, 2010.

[10] E. Snelson and Z. Ghahramani. Sparse Gaussian Processes using Pseudo-Inputs. *Advances in neural information processing*, 2006.

[11] L. L. Thurstone. A Law of Comparative Judgement. *Psychological Review*, 34, 1927.

Appendix C

Pseudo Inputs for Pairwise Learning with Gaussian Processes

Jens Brehm Nielsen, Bjørn Sand Jensen, Jan Larsen. Pseudo Inputs for Pairwise Learning with Gaussian Processes. Published in *2012 IEEE International Workshop on Machine Learning for Signal Processing*, 2012. ISSN 1551-2541. doi:10.1109/MLSP.2012.6349812.

Pseudo Inputs for Pairwise Learning with Gaussian Processes

Jens Brehm Nielsen, Bjørn Sand Jensen and Jan Larsen

DTU Informatics, Technical University of Denmark
Asmussens Alle B305, 2800 Kgs. Lyngby, Denmark
{jenb,bjje,jl}@imm.dtu.dk

Preprint

Abstract

We consider learning and prediction of pairwise comparisons between instances. The problem is motivated from a perceptual view point, where pairwise comparisons serve as an effective and extensively used paradigm. A state-of-the-art method for modeling pairwise data in high dimensional domains is based on a classical pairwise probit likelihood imposed with a Gaussian process prior. While extremely flexible, this non-parametric method struggles with an inconvenient $\mathcal{O}\left(n^3\right)$ scaling in terms of the n input instances which limits the method only to smaller problems. To overcome this, we derive a specific sparse extension of the classical pairwise likelihood using the pseudo-input formulation. The behavior of the proposed extension is demonstrated on a toy example and on two real-world data sets which outlines the potential gain and pitfalls of the approach. Finally, we discuss the relation to other similar approximations that have been applied in standard Gaussian process regression and classification problems such as FI(T)C and PI(T)C.

1 INTRODUCTION

The pairwise learning setting has several application areas such as preference learning and ranking [1], metric learning [2] and general pairwise comparison paradigms. Pairwise comparisons are naturally motivated from a perceptual point of view, where human subjects make a sequence of pairwise (subjective) preference decisions in relation to sound quality, music taste, etc. The main advantage is that pairwise relations are relatively easy for subjects to convey consistently since subjects do not need an internal reference.

The theory underlying pairwise comparisons was first formulated in a principle manner in [3] stating *The Law of Comparative Judgments* building on cognitive and perceptual ideas. The basic idea is that a choice is determined by the difference in the response from a latent stochastic process. The resulting likelihood function in its simplest form—which is also by far the most common one—was first put into the flexible framework of Gaussian processes priors in [4].

Gaussian process based models are flexible and thus desirable for pairwise learning, but struggle with an inconvenient $\mathcal{O}\left(n^3\right)$ scaling in terms of the number of input instances n. This makes their use impractical for large-scale problems. Several suggestions have been proposed to remedy this issue for the standard Gaussian process regression case by using a smaller set of inputs that is either a subset of the original input set [5, 6] or a completely new set of *pseudo inputs* [7, 8, 9]. An unifying view of the latter family of models is given in [10] and extended in [11] leading to the well-known FI(T)C and PI(T)C approximations for standard regression and classification models.

In the standard case the explicit formulation of pseudo inputs can easily and without further considerations be turned into a conditional Gaussian process prior with an easy to invert covariance matrix. However, in the pairwise case the likelihood function depends on two variables. Therefore, we cannot immediately and without consideration use the standard approximations in the covariance as done in [12]. Instead, our quest to derive a sparse approximation for pairwise problems starts from the original pseudo-input formulation presented in [7]. Using this direct approach, our objective is to extend the pairwise likelihood model to allow for explicit sparsity in input space achieved by extending the model by a set of pseudo inputs—or inducing points—of size $l \ll n$. Essentially, the pseudo inputs are used to integrate out the two original variables of the classical pairwise likelihood function. In effect the Gaussian process prior is now placed over the function values of the pseudo inputs often resulting in a considerably lower computational load. Posterior inference relies on a Laplace approximation and the pseudo inputs can be found by evidence optimization for example initialized by k-means.

We give insight and intuition about the behavior and performance of the sparse model compared with the standard model by considering the *Boston housing* data set and a *wine-quality* data set. Examination of the out-of-sample error rates is the basis for discussing the potential and limitations of the sparse model.

2 MODEL & EXTENSIONS

In this section we describe the general setup and frame the pairwise model in a Bayesian non-parametric setting. Each input instance i is described by a feature vector $\mathbf{x} \in \mathbb{R}^d$ and $\mathcal{X} = \{\mathbf{x}_i | i = 1, ..., n\}$. Next, we consider a data set $\mathcal{Y} = \{y_k; u_k, v_k | k = 1, ..., m\}$ of pairwise relations $y \in \{-1, +1\}$ between the u'th and the v'th instance of $\mathcal{X}$, hence $\mathbf{x}_{u_k}, \mathbf{x}_{v_k} \in \mathcal{X}$ [1] . The two opposite choices picking either the u'th or the v'th instance are denoted by $y = -1$ and $y = +1$, respectively.

Given two latent function values $\mathbf{f}_k = [f(\mathbf{x}_{u_k}), f(\mathbf{x}_{v_k})]^\top$, the observations are modeled by a pairwise likelihood function $p\left(y_k | \mathbf{f}_k, \boldsymbol{\theta}_\mathcal{L}\right)$ with parameter(s) $\boldsymbol{\theta}_\mathcal{L}$. The function f is an latent function, which in a Thurstonian context [13], models the mean absolute response from the internal cognitive process when the subject is exposed to an input instance. The function parametrization admits that we directly place a zero-mean Gaussian process [14] prior on f allowing for a flexible predictive model for the pairwise responses. Formally, we write $f(\mathbf{x}_i) \sim$

[1] We will without loss of generality assume that the set $\mathcal{Y}$ involves all n inputs instances in $\mathcal{X}$.

$\mathcal{GP}\left(0, k_{\boldsymbol{\theta}_{\mathcal{GP}}}(\mathbf{x}_i, \cdot)\right)$, where $k(\cdot,\cdot)$ denotes a covariance function, or kernel, with parameter(s) $\boldsymbol{\theta}_{\mathcal{GP}}$, which generally speaking restricts the smoothness of the function. The fundamental consequence of a Gaussian process is that the joint distribution of a finite set of function values $\mathbf{f} = [f(\mathbf{x}_1), f(\mathbf{x}_2), f(\mathbf{x}_3), ..., f(\mathbf{x}_n)]^\top$ has a multivariate Gaussian distribution defined by $p\left(\mathbf{f}|\mathcal{X}, \boldsymbol{\theta}_{\mathcal{GP}}\right) = \mathcal{N}\left(\mathbf{0}, \mathbf{K}_{\mathcal{X}\mathcal{X}}\right)$, where the elements of the covariance matrix are given as $[\mathbf{K}_{\mathcal{X}\mathcal{X}}]_{i,j} = k_{\boldsymbol{\theta}_{\mathcal{GP}}}(\mathbf{x}_i, \mathbf{x}_j)$. Given a standard Bayesian framework and assuming i.i.d. comparisons we now obtain the posterior over the function values

$$p\left(\mathbf{f}|\mathcal{X}, \mathcal{Y}, \boldsymbol{\theta}\right) \propto p\left(\mathbf{f}|\mathcal{X}, \boldsymbol{\theta}_{\mathcal{GP}}\right) \prod_{k=1}^{m} p\left(y_k|\mathbf{f}_k, \boldsymbol{\theta}_{\mathcal{L}}\right)$$

with $\boldsymbol{\theta} = \{\boldsymbol{\theta}_{\mathcal{L}}, \boldsymbol{\theta}_{\mathcal{GP}}\}$. The main computational issue in the Gaussian process framework is to calculate/approximate the posterior posing a $\mathcal{O}\left(n^3\right)$ scaling challenge due to the inversion of the kernel matrix.

2.1 Standard Pairwise Likelihood Function

The pairwise likelihood function described in a general pairwise context by [13] and used with Gaussian processes by e.g. [4] and [15] is given by

$$p\left(y_k|\mathbf{f}_k, \boldsymbol{\theta}_{\mathcal{L}}\right) = \Phi\left(y_k \frac{f\left(\mathbf{x}_{u_k}\right) - f\left(\mathbf{x}_{v_k}\right)}{\sqrt{2}\sigma}\right), \tag{1}$$

where $\Phi(\cdot)$ defines a cumulative Gaussian (with zero mean and unity variance) and $\boldsymbol{\theta}_{\mathcal{L}} = \{\sigma\}$. The use of a Gaussian process prior in connection with this likelihood function was first proposed in [4].

2.2 Sparse Pairwise Likelihood Function

To obtain sparsity in input space, we generally follow the ideas in [7]. Hence, given a set of pseudo inputs $\bar{\mathbf{X}}$, their functional values $\bar{\mathbf{f}}$ must originate from the same Gaussian process that was used for $\mathbf{f}$. Therefore, we can directly place a Gaussian process prior over $\bar{\mathbf{f}}$, i.e., $p\left(\bar{\mathbf{f}}|\bar{\mathbf{X}}\right) = \mathcal{N}\left(\bar{\mathbf{f}}|\mathbf{0}, \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}\right)$, where the matrix $\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}$ is the covariance matrix of the l pseudo inputs collected in the matrix $\bar{\mathbf{X}} = [\bar{\mathbf{x}}_1, ..., \bar{\mathbf{x}}_l]$.

The overall idea of the pseudo-input formalism is now to refine the likelihood function from Eq. (1) such that the real $\mathbf{f}$ values that enter directly in the original, non-sparse likelihood function (through $\mathbf{f}_k$), exist only in the form of predictions from the pseudo inputs $\bar{\mathbf{f}}(\bar{\mathbf{X}})$. Given the listed assumptions, we formally have that $\mathbf{f}$ and $\bar{\mathbf{f}}$ are jointly Gaussian, hence

$$\begin{bmatrix} \mathbf{f}_k \\ \bar{\mathbf{f}} \end{bmatrix} = \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{K}_{\mathbf{x}_k\mathbf{x}_k} & \mathbf{K}_{\bar{\mathbf{X}}\mathbf{x}_k}^\top \\ \mathbf{K}_{\bar{\mathbf{X}}\mathbf{x}_k} & \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}} \end{bmatrix}\right), \tag{2}$$

where we define the following matrices and vectors

$$\mathbf{K}_{\mathbf{x}_k\mathbf{x}_k} = \begin{bmatrix} k(\mathbf{x}_{u_k}, \mathbf{x}_{u_k}) & k(\mathbf{x}_{u_k}, \mathbf{x}_{v_k}) \\ k(\mathbf{x}_{v_k}, \mathbf{x}_{u_k}) & k(\mathbf{x}_{v_k}, \mathbf{x}_{v_k}) \end{bmatrix} \tag{3}$$

$$\mathbf{K}_{\bar{\mathbf{X}}\mathbf{x}_k} = [\mathbf{k}_{u_k}, \mathbf{k}_{v_k}] \tag{4}$$

with $[\mathbf{k}_{u_k}]_i = k(\bar{\mathbf{x}}_i, \mathbf{x}_{u_k})$ and $[\mathbf{k}_{v_k}]_i = k(\bar{\mathbf{x}}_i, \mathbf{x}_{v_k})$. From Eq. (2) it is trivial to find the conditional distribution of $\mathbf{f}_k$ given $\bar{\mathbf{f}}$, hence the sparse likelihood function can be derived in terms of $\bar{\mathbf{f}}$ by integrating over $\mathbf{f}_k$, thus

$$p\left(y_k|\mathbf{x}_{u_k}, \mathbf{x}_{v_k}, \bar{\mathbf{X}}, \bar{\mathbf{f}}, \boldsymbol{\theta}\right) = \int p\left(y_k|\mathbf{f}_k, \boldsymbol{\theta}_{\mathcal{L}}\right) p\left(\mathbf{f}_k|\bar{\mathbf{f}}, \bar{\mathbf{X}}\right) d\mathbf{f}_k$$
$$= \int \Phi\left(y_k \frac{f\left(\mathbf{x}_{u_k}\right) - f\left(\mathbf{x}_{v_k}\right)}{\sqrt{2}\sigma}\right) \mathcal{N}\left(\mathbf{f}_k|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right) d\mathbf{f}_k = \Phi\left(y_k \frac{\mu_{u_k} - \mu_{v_k}}{\sigma_k^*}\right)$$

where $\boldsymbol{\mu}_k = [\mu_{u_k}, \mu_{v_k}]^\top$, $\mu_{u_k} = \mathbf{k}_{u_k}^\top \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1} \bar{\mathbf{f}}$, $\mu_{v_k} = \mathbf{k}_{v_k}^\top \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1} \bar{\mathbf{f}}$ and

$$\boldsymbol{\Sigma}_k = \begin{bmatrix} \sigma_{u_k u_k} & \sigma_{u_k v_k} \\ \sigma_{v_k u_k} & \sigma_{v_k v_k} \end{bmatrix} = \mathbf{K}_{\mathbf{x}_k \mathbf{x}_k} - \mathbf{K}_{\bar{\mathbf{X}}\mathbf{x}_k}^\top \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1} \mathbf{K}_{\bar{\mathbf{X}}\mathbf{x}_k}$$

Furthermore, $(\sigma_k^*)^2 = 2\sigma^2 + \sigma_{u_k u_k} + \sigma_{v_k v_k} - \sigma_{u_k v_k} - \sigma_{v_k u_k}$, which all together results in the pseudo-input likelihood

$$p\left(y_k|\mathbf{x}_{u_k}, \mathbf{x}_{v_k}, \bar{\mathbf{X}}, \bar{\mathbf{f}}, \boldsymbol{\theta}\right) = \Phi\left(z_k\right), \tag{5}$$

with $z_k = y_k \left(\mathbf{k}_{u_k}^T - \mathbf{k}_{v_k}^T\right) \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1} \bar{\mathbf{f}} / \sigma_k^*$.

2.3 Inference & Predictions

The likelihood functions described in Section 2.1 and 2.2 lead to intractable posteriors and call for approximation techniques or sampling methods. Our goal in this initial study is to examine the sparse model and its properties—not to provide the optimal approximation—hence, we only explore inference based on the Laplace approximation.

2.3.1 Posterior Approximation

Inference using the Laplace approximation has also been applied in [16] for the standard model. The general solution to the approximation problem can be found by maximizing the unnormalized log-posterior $\psi\left(\bar{\mathbf{f}}|\mathcal{Y}, \mathcal{X}, \bar{\mathbf{X}}, \boldsymbol{\theta}\right) = \log p\left(\mathcal{Y}|\bar{\mathbf{f}}, \mathcal{X}, \bar{\mathbf{X}}, \boldsymbol{\theta}\right) - \frac{1}{2}\bar{\mathbf{f}}^T \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1} \bar{\mathbf{f}} - \frac{1}{2}\log|\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}| - \frac{l}{2}\log 2\pi$ with regards to $\bar{\mathbf{f}}$. For the maximization we use a damped Newton method in which the damped step (with adaptive damping factor λ) can be calculated without inversion of the Hessian

$$\bar{\mathbf{f}}^{new} = \left(\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1} + \mathbf{W} - \lambda\mathbf{I}\right)^{-1} \left[\left(\mathbf{W} - \lambda\mathbf{I}\right)\bar{\mathbf{f}} + \nabla \log p(\mathcal{Y}|\bar{\mathbf{f}}, \mathcal{X}, \bar{\mathbf{X}}, \boldsymbol{\theta})\right]. \tag{6}$$

Using the notation $\nabla\nabla_{i,j} = \frac{\partial^2}{\partial f(x_i)\partial f(x_j)}$ we apply the definition $\mathbf{W}_{i,j} = -\sum_k \nabla\nabla_{i,j} \log p(y_k|\mathbf{x}_{u_k}, \mathbf{x}_{v_k}, \bar{\mathbf{X}}, \bar{\mathbf{f}}, \boldsymbol{\theta})$. When converged, the resulting approximation can be shown to be $p\left(\bar{\mathbf{f}}|\mathcal{Y}, \mathcal{X}, \bar{\mathbf{X}}, \boldsymbol{\theta}\right) \approx \mathcal{N}\left(\bar{\mathbf{f}}|\hat{\mathbf{f}}, \left(\mathbf{W} + \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1}\right)^{-1}\right)$. The damped Newton step requires the Jacobian and Hessian of the new pseudo-input log-likelihood from Eq. (5), which require the following two derivatives

$$\frac{\partial}{\partial \bar{\mathbf{f}}} p\left(y_k|...\right) = y_k \frac{\mathcal{N}\left(z_k\right)}{\sigma_k^* \Phi\left(z_k\right)} \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1} \left(\mathbf{k}_{u_k} - \mathbf{k}_{v_k}\right) \tag{7}$$

$$\frac{\partial^2}{\partial \bar{\mathbf{f}} \bar{\mathbf{f}}^\top} p\left(y_k|...\right) = -y_k^2 \frac{\mathcal{N}\left(z_k\right)}{(\sigma_k^*)^2 \Phi\left(z_k\right)} \left[z_k + \frac{\mathcal{N}\left(z_k\right)}{\Phi\left(z_k\right)}\right]$$
$$\cdot \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1} \left(\mathbf{k}_{u_k} - \mathbf{k}_{v_k}\right) \left(\mathbf{k}_{u_k} - \mathbf{k}_{v_k}\right)^\top \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1}. \tag{8}$$

2.3.2 Evidence / Hyperparameter Optimization

So far we have simply considered the hyperparameters $\boldsymbol{\theta} = \{\boldsymbol{\theta}_{\mathcal{L}}, \boldsymbol{\theta}_{\mathcal{GP}}\}$ and pseudo inputs $\bar{\mathbf{X}}$ as fixed parameters, but their values have a crucial influence on the model performance. Here, we resort to point estimates and find (possible locally) optimal values by iterating between the Laplace approximation with fixed hyperparameters, i.e., finding $p\left(\hat{\mathbf{f}}|\mathcal{Y}, \mathcal{X}, \bar{\mathbf{X}}, \boldsymbol{\theta}\right)$, followed by an evidence maximization step in which $(\boldsymbol{\theta}, \bar{\mathbf{X}}) = \arg\max_{(\boldsymbol{\theta}, \bar{\mathbf{X}})} p\left(\mathcal{Y}|\boldsymbol{\theta}, \bar{\mathbf{X}}\right)$. The log-evidence $\log p(\mathcal{Y}|\boldsymbol{\theta}, \bar{\mathbf{X}})$ has to be approximated in our case, which in terms of the existing Laplace approximation yields

$$\begin{aligned}\log p\left(\mathcal{Y}|\boldsymbol{\theta}, \bar{\mathbf{X}}\right) \approx \log q\left(\mathcal{Y}|\bar{\mathbf{X}}, \boldsymbol{\theta}\right) = \log p(\mathcal{Y}|\hat{\mathbf{f}}, \bar{\mathbf{X}}, \mathcal{X}, \boldsymbol{\theta}) \\ - \frac{1}{2}\hat{\mathbf{f}}^T \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}^{-1} \hat{\mathbf{f}} - \frac{1}{2}\log\left|\mathbf{I} + \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}\mathbf{W}\right|. \end{aligned} \tag{9}$$

We further allow for fixed hyperpriors on the individual hyperparameters serving as regularization, which results in a procedure referenced to as MAP-II which provides more robust estimation. Consequently, the MAP-II is given by $\log q_{\text{MAP-II}}\left(\mathcal{Y}|\bar{\mathbf{X}}, \boldsymbol{\theta}\right) = \log q\left(\mathcal{Y}|\bar{\mathbf{X}}, \boldsymbol{\theta}\right) + \log p\left(\boldsymbol{\theta}, \bar{\mathbf{X}}|\xi\right)$, where ξ is a set of fixed parameters in the hyperprior.

The optimization requires the derivatives of the evidence approximation. These turn out to be rather tedious and involved, and we refer to the appendix for details. The pseudo-input model poses a number of difficulties since $\bar{\mathbf{X}}$ are also to be considered hyperparameters. Typically, this will—as noted by [7] and [17]—lead to a large number of local maxima providing potentially suboptimal solutions. It is not our aim to resolve nor document this issue, and we will take a pragmatic view and simply accept evidence optimization methods as is. Like [17] we recommend starting out with a fixed set of pseudo inputs initialized by a standard unsupervised clustering, such as k-means with restarts, followed by evidence optimization.

2.3.3 Predictions

The main task is to infer the latent function values $\bar{\mathbf{f}}$ with the end objective to make predictions of the observable variable y for a pair of test inputs $\mathbf{x}_r \in \mathcal{X}_t$ and $\mathbf{x}_s \in \mathcal{X}_t$ denoted $\mathbf{x}_t = [\mathbf{x}_r, \mathbf{x}_s]^T$. We consider the joint distribution between $\bar{\mathbf{f}} \sim p\left(\bar{\mathbf{f}}|\mathcal{Y}, \boldsymbol{\theta}\right)$ and the test variables $\mathbf{f}_t = [f(\mathbf{x}_r), f(\mathbf{x}_s)]^T$. With the posterior of $\bar{\mathbf{f}}$ approximated with the Gaussian from the Laplace approximation, the predictive distribution $p\left(\mathbf{f}_t|\mathcal{Y}, \boldsymbol{\theta}\right)$ will also be Gaussian given by $\mathcal{N}\left(\mathbf{f}_t|\mu^*, \mathbf{K}^*\right)$ with $\mu^* = [\mu_r^*, \mu_s^*]^T = \mathbf{k}_t \mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}{}^{-1}\bar{\mathbf{f}}$ and

$$\mathbf{K}^* = \begin{bmatrix} \sigma_{rr}^* & \sigma_{rs}^* \\ \sigma_{sr}^* & \sigma_{ss}^* \end{bmatrix} = \mathbf{K}_t - \mathbf{k}_t^T\left(\mathbf{I} + \mathbf{W}\mathbf{K}_{\bar{\mathbf{X}}\bar{\mathbf{X}}}\right)\mathbf{k}_t,$$

where $\mathbf{k}_t$ is the kernel between the test points and the pseudo inputs. With $p\left(\mathbf{f}_t|\mathcal{Y}, \boldsymbol{\theta}\right)$, the prediction distribution of the observed variable is given as

$$p\left(y_t|\mathcal{Y}, \boldsymbol{\theta}\right) = \int p\left(y_t|\mathbf{f}_t, \boldsymbol{\theta}_{\mathcal{L}}\right) p\left(\mathbf{f}_t|\mathcal{Y}, \boldsymbol{\theta}\right) d\mathbf{f}_t.$$

The integral can be calculated in closed form as

$$P\left(\mathbf{x}_r \succ \mathbf{x}_s|\mathcal{Y}, \boldsymbol{\theta}\right) = \Phi\left(\left(\mu_r^* - \mu_s^*\right)/\sigma^*\right)$$

with $\left(\sigma^*\right)^2 = 2\sigma^2 + \sigma_{rr}^* + \sigma_{ss}^* - \sigma_{rs}^* - \sigma_{sr}^*$.

3 SIMULATIONS & EXPERIMENTAL RESULTS

In this section we demonstrate the performance of the pseudo-input method on a toy example and provide predictive performance on two real-world data sets: *Boston housing* and *wine quality*. The main objective is not to achieve the overall best performance, but to compare the standard (GP) and the sparse (SPGP) formulations.

3.1 Toy Example

To illustrate the basics of the SPGP model, we draw a deterministic function f_{real} (see Fig. 1(a)) from a zero-mean Gaussian process with a squared exponential covariance function. This function is then used to generate a pairwise data set consisting of all possible pairwise comparisons using the function values at equidistantly distributed locations marked with black crosses in Fig. 1(a). To model this data, we consider the two models: The GP model (Sec. 2.1) and the SPGP model with optimized pseudo inputs (Sec. 2.2). The $l = 9$ pseudo inputs are initialized equidistantly in the input interval, the length scale of the covariance function $\boldsymbol{\theta}_{\mathcal{GP}} = \{\sigma_\ell\}$ and the likelihood parameter $\boldsymbol{\theta}_{\mathcal{L}} = \{\sigma\}$ are learned by evidence optimization whereas $\sigma_f = 1$ of the covariance function is fixed. The results are presented in Fig. 1(a).

We notice that the SPGP model is capable of modeling the mean and thereby the actual pairwise relationships, whereas the predictive variance differs significantly from the GP variance. This is a characteristic and expected artifact also seen in connection with the pseudo-input models for standard classification and regression.

3.2 Real World Examples

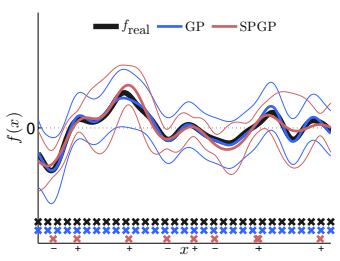
We compare the performance of the SPGP model to the GP model on two different real-world data sets.

The first data set is the well-known *Boston housing*[2] where we have constructed a full pairwise version by using all $m = 127765$ pairwise combinations of the $n = 506$ inputs base on the house price. For each input we use all available features except RAD, CHAS and NOX, thus $d = 10$.

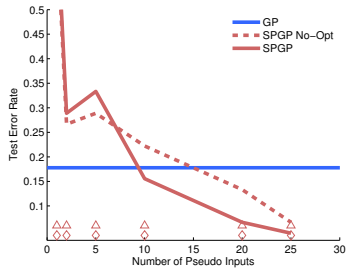
The second data set is a subset of the *wine quality*[3] which is based on user ratings of wines. The subset is based on $n = 600$ instances of wines described by $d = 11$ features. We construct the set of unique pairwise comparisons from the ratings resulting in $m = 179700$ comparisons.

We use a squared exponential covariance function for both data sets which (based on initial experimentation) is initialized with $\sigma_f = 1$ and $\sigma_\ell = 1$. The covariance parameter σ_f is fixed, whereas the likelihood parameter initialized as $\boldsymbol{\theta}_{\mathcal{L}} = \{\sigma = 1\}$ and $\boldsymbol{\theta}_{\mathcal{GP}} = \{\sigma_\ell\}$ are learned by MAP-II optimization using a uniform hyperprior and a half-student-t hyperprior with scale 6 and 4 degrees of freedom, respectively. Pseudo inputs are initialized with k-means (selecting

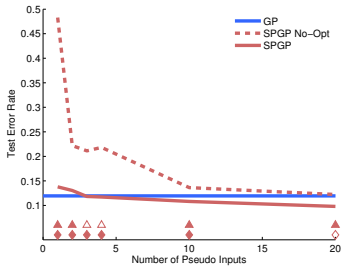
[2] `archive.ics.uci.edu/ml/datasets/Housing`
[3] `archive.ics.uci.edu/ml/datasets/Wine+Quality`



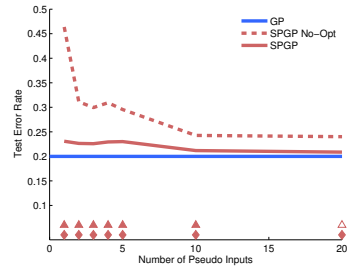
(a) Toy: $d = 1$, $n = 31$, $m = 465$, $l = 9$



(b) Toy: $d = 1$, $n = 31$, $m = 465$



(c) Boston Housing: $d = 10$, $n = 506$, $m = 127765$



(d) Wine Quality: $d = 11$, $n = 600$, $m = 179700$

Figure 1: In general, blue graphs indicate the full model (GP) and red indicate the sparse model (SPGP). In **Fig. (a)** thick graphs indicate means and thin graphs indicate one standard deviation. The black graph indicates the real (deterministic) function used to generate the full pairwise data set between the instances marked with black crosses in the bottom. The two other colors sketch the predictive distribution of the GP and SPGP models using the (pseudo) inputs at the locations marked with the corresponding color in the bottom. **Fig. (b)-(d)** display the performance of the sparse model (SPGP) evaluated on the toy example and on the two real-world data sets as a function of the number of pseudo inputs for the sparse model (red). The performance of the standard model is included as a baseline. The **solid** and **dashed** red graphs show the average test error rate for the optimized and non-optimized SPGP model, respectively. The two rows of markers indicate whether the optimized (triangle) and non-optimized (diamond) SPGP models are significant different from the GP model using the McNemar test. The markers are solid if the null hypothesis that they are equal can be rejected at the 5% significance level.

the solution with minimum total squared distance out of five random initializations). We compare two SPGP models: one where the pseudo inputs are kept fixed following the k-means initialization (this model is identified with the No-Opt tag) and one where they are further fitted using MAP-II with a uniform hyperprior. With both data sets we use 20-fold cross validation on instances, such that a minimum of two instances are held out for testing and a randomly selected quarter of all remaining pairwise comparisons between training instances

are used for training. Consequently, predictions are only performed on comparisons between instances that do not appear in the training data and the setting is thus a true predictive ranking scenario. In Fig. 1(c)-(d) we report the average error rate on the test set as a function of the number of pseudo inputs for the two SPGP models. The GP model is included as a baseline.

4 DISCUSSION

In the toy example (Fig. 1(a)) we see that the mean is well modeled by both the GP model and the SPGP model with $l = 9$ pseudo inputs, suggesting that the SPGP model performs nearly as good as the GP model. The main difference between the two models seems to be the predictive variance which differs significantly, yet this is an expected property of the sparse model. A way to improve the estimation of the predictive variance is by allowing the input instances and pseudo inputs to have different length scales [8][17].

Focusing on the predictive mean performance of the optimized SPGP model on the two real-world data sets (Fig. 1(c)-(d)), we see that a SPGP model with few pseudo inputs (as low as 1-5) performs only slightly worse than or equal to the GP model. This indicates that the two real-world problems do not constitute very complex pairwise problems. The performance is, however, highly dependent on the optimization of the locations of the pseudo inputs, seen since the non-optimized SPGP model requires more pseudo inputs due to the fixed locations. This illustrates the importance and power of the optimization.

By further adding pseudo inputs we can obtain better performance than the GP model. We believe that two effects come into play. The first effect is that the constraints induced in the SPGP model provide better regularization compared to the full Gaussian process prior meaning that it generalizes better. The second effect stems from the fact that the arbitrary placement of the pseudo inputs provides added flexibility, which effectively renders it more adequate for capturing the important regions of the underlying function when these locations are optimized appropriately. We speculate that the observed behavior is a combination of the two effects of course dependent on the application.

A further aspect to be investigated is the capability of the SPGP model to capture and approximate higher order moments of the predictive distribution. In line with previous work on the topic and with the variances observed in the toy example, we have observed fluctuating behavior of the predictive likelihoods as a function of l for the SPGP models in the two real-world examples. Whether the behavior is due to the pairwise setting, specific application or a general property of the pseudo-input formulation is an open question.

In the current sparse formulation the original function values are dependent in pairs given the exact comparisons, whereas in FI(T)C all the original function values are independent given the pseudo inputs. We plan to investigate if this difference have any practical importance and to compare the current approximation to other traditional approaches—in particular the PI(T)C approximation.

5 CONCLUSION

In this paper we have derived a sparse version of the pairwise likelihood model using the pseudo-input formulation. We applied the Laplace approximation for both posterior and evidence approximation. We observe competitive predictive performance with the sparse model using only few pseudo inputs on a toy example and on two real-world data sets. A noticeable observation is the fact that we by adding more pseudo inputs are able to obtain better performance than the full GP model in the studied applications.

Acknowledgement: This work was supported in part by the IST Programme of the European Community, under the PASCAL2 Network of Excellence, IST-2007-216886. This publication only reflects the authors' views.

References

[1] J. Fürnkranz and E. Hüllermeier, *Preference Learning*, Springer, 1st edition, 2011.

[2] B. McFee and G. Lanckriet, "Metric Learning to Rank," *ICML 2010 - Proceedings, 27th International Conference on Machine Learning*, pp. 775–782, 2010.

[3] L. L. Thurstone, "A Law of Comparative Judgement," *Psychological Review*, vol. 34, 1927.

[4] W. Chu and Z. Ghahramani, "Preference Learning with Gaussian Processes," *Proceedings of the 22nd International Conference on Machine Learning (ICML)*, pp. 137–144, 2005.

[5] N. Lawrence, M. Seeger, and R. Herbrich, "Fast Sparse Gaussian Process Methods: The Informative Vector Machine," in *Neural Information Processing Systems (NIPS)*, 2002, p. 8.

[6] L. Csato, *Gaussian Processes - Iterative Sparse Approximations*, Ph.D. thesis, Aston University, 2002.

[7] E. Snelson and Z. Ghahramani, "Sparse Gaussian Processes using Pseudo-Inputs," *Advances in neural information processing*, 2006.

[8] C. Walder, K. I. Kim, and B. Schölkopf, "Sparse Multiscale Gaussian Process Regression," in *Proceedings of the 25th international conference on Machine Learning*, 2008, pp. 1112–1119.

[9] M. Lazaro-Gredilla and A. Figueiras-Vidal, "Inter-Domain Gaussian Processes for Sparse Inference using Inducing Features," in *Advances in Neural Information Processing Systems 22*, pp. 1087–1095. 2009.

[10] J. Quiñonero-Candela and C.E. Rasmussen, "A Unifying View of Sparse Approximate Gaussian Process Regression," *The Journal of Machine Learning Research*, vol. 6, pp. 1939–1959, 2005.

[11] E. Snelson and Z. Ghahramani, "Local and Global Sparse Gaussian Process Approximations," in *Eleventh International Conference on Artificial Intelligence and Statistics, AISTATS*, 2007, pp. 524–531.

[12] J. Guiver and E. Snelson, "Learning to Rank with Softrank and Gaussian Processes," *Annual ACM Conference on Research and Development in Information Retrieval*, pp. 259–266, 2008.

[13] R. D. Bock and J. V. Jones, "The Measurement and Prediction of Judgment and Choice," 1968.

[14] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, 2006.

[15] E. Bonilla, S. Guo, and S. Sanner, "Gaussian Process Preference Elicitation," in *Advances in Neural Information Processing Systems 23*.

[16] W. Chu and Z. Ghahramani, "Extensions of Gaussian Processes for Ranking: Semi-Supervised and Active Learning," in *Workshop Learning to Rank at Advances in Neural Information Processing Systems 18*, 2005.

[17] Y. Qi, A. Abdel-Gawad, and T. Minka, "Sparse-Posterior Gaussian Processes for General Likelihoods," in *Proceedings of the Twenty-Sixth Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-10)*, 2010.

6 Appendix - Evidence Derivatives

The derivatives of Eq. (9) are slightly different compared to the standard classification case [14, Sec 5.5.1] due to the pseudo-input model because the covariance parameters enter into the likelihood, and the fact that the covariance function also depends on $\bar{\mathbf{X}}$. We outline the derivations by noting that the Eq. (9) depends both explicitly and implicitly (due to the solution of $\hat{\mathbf{f}}$) on the parameters $\boldsymbol{\theta}$. We do not differentiate between likelihood and covariance parameters and $\bar{\mathbf{X}}$. Here, we simply denote a parameter by θ_j. We can split the derivatives into an explicit and implicit part

$$\frac{\partial \log q\left(\mathcal{Y}|...\right)}{\partial \theta_i} = \left.\frac{\partial \log q\left(\mathcal{Y}|...\right)}{\partial \theta}\right|_{\text{explicit}} + \sum_j \frac{\partial \log q\left(\mathcal{Y}|...\right)}{\partial f_j}\frac{\partial f_j}{\partial \theta_i}.$$

Referring to the **explicit** term we obtain the following terms

$$\begin{aligned}
&\frac{\partial}{\partial \boldsymbol{\theta}_i}\log p\left(\mathcal{Y}|\hat{\mathbf{f}},\boldsymbol{\theta}\right)\\
&\quad\frac{\partial}{\partial \boldsymbol{\theta}_i}\hat{\mathbf{f}}^\top {\mathbf{K}_{\boldsymbol{\theta}}}^{-1}\hat{\mathbf{f}} = -\hat{\mathbf{f}}^\top\left({\mathbf{K}_{\boldsymbol{\theta}}}^{-1}\frac{\partial \mathbf{K}_{\boldsymbol{\theta}}}{\partial \boldsymbol{\theta}_i}{\mathbf{K}_{\boldsymbol{\theta}}}^{-1}\right)\hat{\mathbf{f}}\\
&\frac{\partial}{\partial \boldsymbol{\theta}_i}\log|\mathbf{I}+\mathbf{W}_{\boldsymbol{\theta}}\mathbf{K}_{\boldsymbol{\theta}}| = Tr\left[(\mathbf{I}+\mathbf{K}_{\boldsymbol{\theta}}\mathbf{W}_{\boldsymbol{\theta}})^{-1}\left(\frac{\partial \mathbf{W}_{\boldsymbol{\theta}}}{\partial \boldsymbol{\theta}_i}\mathbf{K}_{\boldsymbol{\theta}}+\mathbf{W}_{\boldsymbol{\theta}}\frac{\partial \mathbf{K}_{\boldsymbol{\theta}}}{\partial \boldsymbol{\theta}_i}\right)\right]
\end{aligned}$$

Referring to the **implicit** term we have (without any assumptions regarding the type of parameter)

$$\frac{\partial \log q\left(\mathcal{Y}|\bar{\mathbf{X}},\mathcal{X},\boldsymbol{\theta}\right)}{\partial f_j} = -\frac{1}{2}Tr\left[(\mathbf{I}+\mathbf{K}_{\boldsymbol{\theta}}\mathbf{W}_{\boldsymbol{\theta}})^{-1}\left(\mathbf{K}_{\boldsymbol{\theta}}\frac{\partial \mathbf{W}_{\boldsymbol{\theta}}}{\partial f_j}\right)\right]$$

$\frac{\partial f_j}{\partial \boldsymbol{\theta}_i}$ is found by exploiting that $\hat{\mathbf{f}} = \mathbf{K}_{\boldsymbol{\theta}}\nabla \log p\left(\mathcal{Y}|\hat{\mathbf{f}},\boldsymbol{\theta}\right)$ at the current solution leading to the following result

$$\frac{\partial f_j}{\partial \boldsymbol{\theta}_i} = (\mathbf{I}+\mathbf{K}_{\boldsymbol{\theta}}\mathbf{W}_{\boldsymbol{\theta}})^{-1}\left(\frac{\partial \mathbf{K}_{\boldsymbol{\theta}}}{\partial \boldsymbol{\theta}_i}\right)\frac{\partial \log p\left(y|\hat{\mathbf{f}},\boldsymbol{\theta}\right)}{\partial \mathbf{f}} + (\mathbf{I}+\mathbf{K}_{\boldsymbol{\theta}}\mathbf{W}_{\boldsymbol{\theta}})^{-1}\mathbf{K}_{\boldsymbol{\theta}}\frac{\partial}{\partial \boldsymbol{\theta}_i}\left(\frac{\partial \log p\left(y|\hat{\mathbf{f}},\boldsymbol{\theta}\right)}{\partial \mathbf{f}}\right)$$

We may exploit that the inverse of the common factor $(\mathbf{I}+\mathbf{K}_{\boldsymbol{\theta}}\mathbf{W}_{\boldsymbol{\theta}})$ can be computed using the Cholesky decomposition which enters robustly into the individual expressions for added numerical stability. The expression above is a general result and valid for both likelihood parameters, covariance parameters and pseudo inputs. In addition, the derivatives of the likelihood, Jacobian, Hessian and covariance function are required. One should be aware that some of the derivatives are zero depending on the actual parameter type (e.g. $\partial \mathbf{K}_{\boldsymbol{\theta}}/\partial \boldsymbol{\theta}_{\mathcal{L}}$).

The gradients are based on the current Laplace approximation. Even though we take into account implicit dependencies, there is in general no guarantee for strictly monotonic behavior, thus a robust optimization method is required. In practice we have found the BFGS implementation in the `immoptibox`[4] robust.

[4] `www2.imm.dtu.dk/%7Ehbn/immoptibox/`

Appendix D

Efficient Individualization of Hearing Aid Processed Sound

Jens Brehm Nielsen, Jakob Nielsen. Efficient Individualization of Hearing Aid Processed Sound. Published in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 398-402, 2013. ISSN 1520-6149. doi:10.1109/ICASSP.2013.6637677.

Efficient Individualization of Hearing aid Processed Sound

Jens Brehm Nielsen[1,2] & Jakob Nielsen[1]

[1]Widex A/S, Nymøllevej 6, DK-3540 Lynge,
{jeb,jnl}@widex.com

[2]DTU Informatics, Asmussens Alle B305, DK-2800 Kgs. Lyngby,
jenb@imm.dtu.dk

Preprint

Abstract

Due to the large amount of options offered by the vast number of adjustable parameters in modern digital hearing aids, it is becoming increasingly daunting—even for a fine-tuning professional—to perform parameter fine tuning to satisfactorily meet the preference of the hearing aid user. In addition, the communication between the fine-tuning professional and the hearing aid user might muddle the task. In the present paper, an interactive system is proposed to ease and speed up fine tuning of hearing aids to suit the preference of the individual user. The system simultaneously makes the user conscious of his own preferences while the system itself learns the user's preference. Since the learning is based on probabilistic modeling concepts, the system handles inconsistent user feedback efficiently. Experiments with hearing impaired subjects show that the system quickly discovers individual preferred hearing-aid settings which are consistent across consecutive fine-tuning sessions for each user.

Hearing aid personalization, Bayesian learning, Gaussian processes, Active learning, Preference learning

1 INTRODUCTION

Modern digital hearing aids (HAs) contain a vast number of adjustable parameters that offer an almost infinite number of possible settings. Different settings make the hearing aids emphasize parts of the incoming sound to make it more or less comfortable, audible, intelligible etc. for the hearing impaired (HI). The procedure of fitting the HAs to the user is performed by skilled professionals like an audiologist.

Having fitted a set of HAs to the hearing loss of the HI user to ensure audibility and intelligibility of incoming sounds, several options are still left for the audiologist to choose from. Some of those are related to the preference of the user. Fine tuning of these parameters is normally done manually by adjusting a

number of handles available in the supplied fitting software. At this point, two aspects should be considered. First, due to the large number of parameters—and thus the number of settings—a manual procedure may not be adequate for finding optimal settings for all parameters even for a fine-tuning expert like an audiologist. Secondly, the success of the fine-tuning process depends on the communication between the HA user and the audiologist. Typically, the HA user has not recognized his own preference beforehand, which may muddle the communication and result in an inadequate fine tuning.

To take full advantage of modern digital hearing aids, more sophisticated fine tuning tools are needed. These should discover the best setting for each individual in robust and time-efficient procedures to take full advantage of the flexibility of the HAs.

In this paper, an interactive system is considered that lets the HA user recognize his own preference by comparing different settings simply by listening to the resulting sounds. By letting the user report how much one setting is preferred over another in a sequence of such comparisons, the interactive system starts to learn the preference of the user. At the end, the interactive system is able to suggest which setting (or subset of settings) that is preferred by the HA user. The system builds on the assumption that each user has an *unobserved* internal representation of preference (IRP), which is a stochastic function (or process) of hearing aid settings. In the interactive system, the *mean response* of the IRP is modeled by a Gaussian process (GP) [1], which loosely speaking defines a *distribution* of functions and thus of possible mean responses of the IRP. In the remainder of the article the IRP is used to refer to the *mean response* of the IRP. The distribution of IRPs is updated iteratively each time the user compares and chooses between two HA settings using the GP framework previously proposed in [2]. To reduce the required number of comparisons needed for the system to learn the user's preference, the distribution of IRP provided by the GP is used to decide the next setting pair to compare. In the literature, this is referred to as *active learning*, and in this paper, a bivariate version of *Expected Improvement* (EI) [3] is used.

Several directions have been pursued to develop systems capable of fine-tuning settings of HAs and other devices. Some of the very first attempts used a modified simplex procedure [4], but required an unrealistic amount of preference assessments to converge. Other tournament based attempts have used genetic algorithms [5, 6], but the convergence time tend to scale badly with the number of tunable parameters. One of the most promising suggestions [7] is also probabilistic and contains at least two ideas that are similar to the ideas underlying the work presented here. Firstly, the method is also based on probabilistic modeling of the user's IRP, but does not use state-of-the-art GPs for this. These are included later in a slightly different context in for instance [8]. Secondly, the two methods also rely on probabilistic choice models that directly address the fact that humans are in general not completely consistent performing perceptual evaluations. However, the two methods are based only on forced choices (discrete decisions) using the choice model and framework from [9, 10, 11], in which subjects only select the option they prefer (discrete choice). This is in contrast to the choice model proposed in [11], in which subjects also decide how much they prefer the selected setting (continuous decision). The results in [2] give reason to believe that the additional information contained in continuous decisions reduces the number of required comparisons needed to learn a user's preferred

setting. This is really the key for the application considered in this work. It is, however, beyond the scope of this work to actually compare results obtained with discrete choices to those obtained with continuous decisions. Nevertheless, this is definitely of great interest for future research. Instead, the focus is to investigate the variability between IRP and thus the preferred setting suggested by the system using continuous decisions.

To test the fine-tuning abilities of the proposed interactive system, two adjustable parameters of a HA were fine-tuned individually to five different HI users. By comparing the results from two similar sessions with each subject, the variability of the found best setting can be investigated. The two HA parameters that were adjusted in the experiments changed how both noise reduction and speech enhancement algorithms should react to the incoming sound.

This article is organized as followed: In section 2, the interactive system is outlined and an explanation of the experiments is provided in section 3. Results are presented in section 4, and finally, section 5 contains the discussion.

2 MODELING FRAMEWORK

A user's *internal representation of preference* (IRP)—referred to as $f : \mathcal{X} \rightarrow \mathbb{R}$—is modeled by a (zero-mean) *Gaussian process* (GP) [1]. The set $\mathcal{X} = \{\mathbf{x}_i \in \mathbb{R}^d : i = 1...n\}$ is the entire set of the n possible settings of the $d = 2$ HA parameters. A GP is a non-parametric—and thus flexible—discriminative Bayesian approach, which defines a distribution of entire functions, "*any finite number of which have a joint Gaussian distribution*" [1, Def. 2.1]. This simply implies that any finite number of function values, $\mathbf{f} = [f(\mathbf{x}_1), ..., f(\mathbf{x}_n)]^\top$, have a distribution given by a multivariate Gaussian distribution as

$$p(\mathbf{f}) = \mathcal{N}(\mathbf{0}, \mathbf{K}), \tag{1}$$

with the elements of $\mathbf{K}$ given by $[\mathbf{K}]_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j)$, where $k(\cdot, \cdot)$ is a covariance function (or kernel), which generally speaking defines the *smoothness* of the functions. For an introduction to kernels, see [1, Chap. 4] or [12, Chap. 6].

The fundamental benefit from the GP is that Eq. 1 can be used as a *prior* distribution of a user's IRP before any preference assessments have been performed by the user. In the Bayesian framework, the distribution of the user's IRP is re-calculated conditioned on the preference assessment(s) that have been observed to give the *posterior* distribution of the user's IRP as

$$p(\mathbf{f}|\mathcal{Y}) \propto p(\mathcal{Y}|\mathbf{f})p(\mathbf{f}), \tag{2}$$

where $p(\mathcal{Y}|\mathbf{f})$ is the *likelihood* which is defined by a specific observational model (choice model). In this work, users assess their *degree* of preference (continuous decision) between two particular HA settings. To update the posterior (and *predictive*) distribution in the GP framework at any given point in the experiment with a particular number of performed preference assessments, the model proposed in [2] is used. The specific functional form of that observational model as well as details about inference and predictions are provided in [2] and will therefore not be presented here.

To reduce the number of preference assessments required to discover the optimal setting, *active learning* is used. Active learning can be formulated

in several ways, but the statistics provided by the GP framework makes it possible to use a slightly modified version of *Expected Improvement* (EI) [3]. In contrast to the original formulation [3], the modification also includes the correlation between function values when calculating the (modified) EI. The added correlations are directly available from the GP framework. The EI for a possible new setting $\mathbf{x}_i$ is thus calculated in closed form as

$$EI(\mathbf{x}_i) = \sigma_{EI} \cdot \phi\left(\frac{\mu_{EI}}{\sigma_{EI}}\right) + \mu_{EI} \cdot \Phi\left(\frac{\mu_{EI}}{\sigma_{EI}}\right), \tag{3}$$

where $\phi(\cdot)$ and $\Phi(\cdot)$ is the standard normal distribution and standard normal cumulative distribution functions, respectively, $\mu_{EI} = \mu_i - \mu_{\max}$ and $\sigma_{EI}^2 = \sigma_i^2 + \sigma_{\max}^2 - 2 \cdot \mathrm{cov}_{i,\max}$. Here, the $_{\max}$ index refers to the point with the *current* largest predicted IRP and the notation

$$p\left(\begin{bmatrix} f_{\max} \\ f_i \end{bmatrix}\right) = \mathcal{N}\left(\begin{bmatrix} \mu_{\max} \\ \mu_i \end{bmatrix}, \begin{bmatrix} \sigma_i^2 & \mathrm{cov}_{i,\max} \\ \mathrm{cov}_{i,\max} & \sigma_{\max}^2 \end{bmatrix}\right) \tag{4}$$

has been used for the two-variate marginal of the predictive normal distribution given by the GP framework.

Typically in active learning theory, an explicit trade-off between exploration (of unseen regions of input space) and exploitation (of "known" regions of input space) must be made. Generally, a system will exhibit slow convergence with too much emphasis on exploration, but will quickly get stuck in a sub-optimal solution, if too much emphasis is put on exploitation. In this work, the next proposed setting to compare with the current best one is sampled from a multinomial distribution, where the probability of a given setting is proportional to its EI given by Eq. 3. This was done to put slightly more emphasis on exploration.

3 Measurement Procedure

To illustrate the behavior of the suggested interactive fine-tuning system, an experiment with five (native danish) HI subjects was conducted. To obtain an indicate of the expected variability in the proposed settings for individual HI users between consecutive fine-tuning sessions, the experiment consisted of both a test session and a re-test session. The two sessions were conducted on two separate days.

In each of the two sessions, each subject conducted thirty comparisons between pairs of HA settings. Subjects wore (experimental) hearing aids fitted (binaurally) in advance to compensate for each individual's hearing loss, and listened to running speech in car noise played back over loudspeakers. Via a graphical user interface (see Fig. 1), the user could switch between the two current HA settings and report their degree of preference. The users were not instructed to focus on particular parts of the sound or on particular attributes, but were only provided with an introduction to the acoustical scene reflected in the sound file. The hearing loss of each individual subject is found in Fig. 2

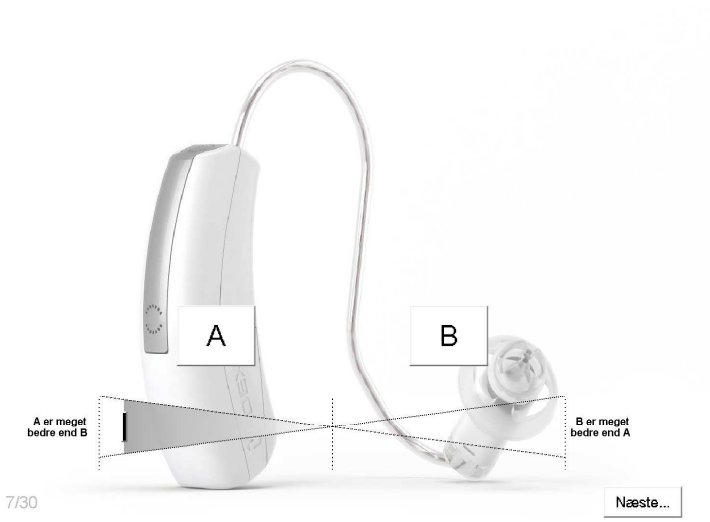


Figure 1: The (danish) graphical user interface used in the experiments. The buttons, 'A' and 'B', were used to switch between the two current settings. The slider—currently positioned far to the left—was used to indicate the degree of preference between the two settings by how far it was positioned towards either of the two settings, 'A' or 'B'. No preference was indicated by leaving the slider at the center. After positioning the slider, the user continued to the next comparison be clicking the button in the lower-right corner.

4 Results

In Fig. 3, the predictions of the IRP for both the test and re-test sessions for each of the five test subjects are depicted. Since the IRPs are unit free, the reader should be aware that the colors cannot be compared across subjects, and similarly, high-preference regions should in general not be interpreted as being "good", but only as being "better than" blue or green. Hence, the predicted IRP only reflects relative properties.

Considering that a parameter change from one end of the space to the other is extremely subtle, the predicted high- and low-preference regions between test and re-tests within each subject are consistent, except for subject 5. The results with subject 5 do, however, coincide with what subject 5 expressed after the sessions, namely that the subject was unable to hear any differences between any of the pairs of presented settings. For this reason, subject 5 chose only occasionally to move the slider, and when the subject did, the subject moved it as little as possible.

A statistic significance test using forced choices was performed to prove significance between the most and least preferred settings discovered by the system in the two sessions. Options 1 and 2 were mixed randomly in eleven trials, yet this only proved significance ($p < 0.005$) for subject 3.

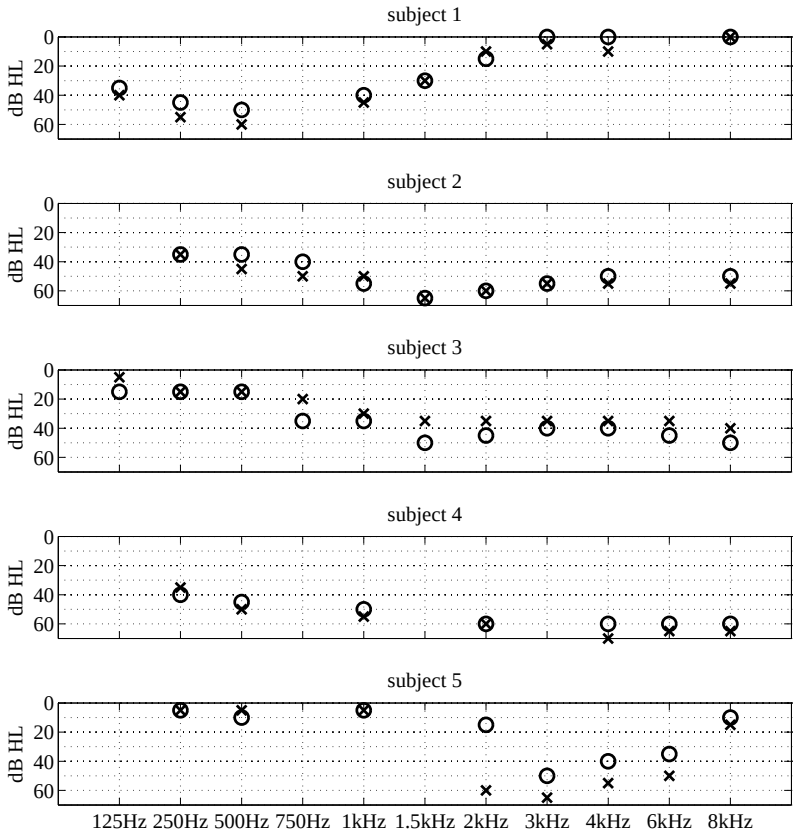


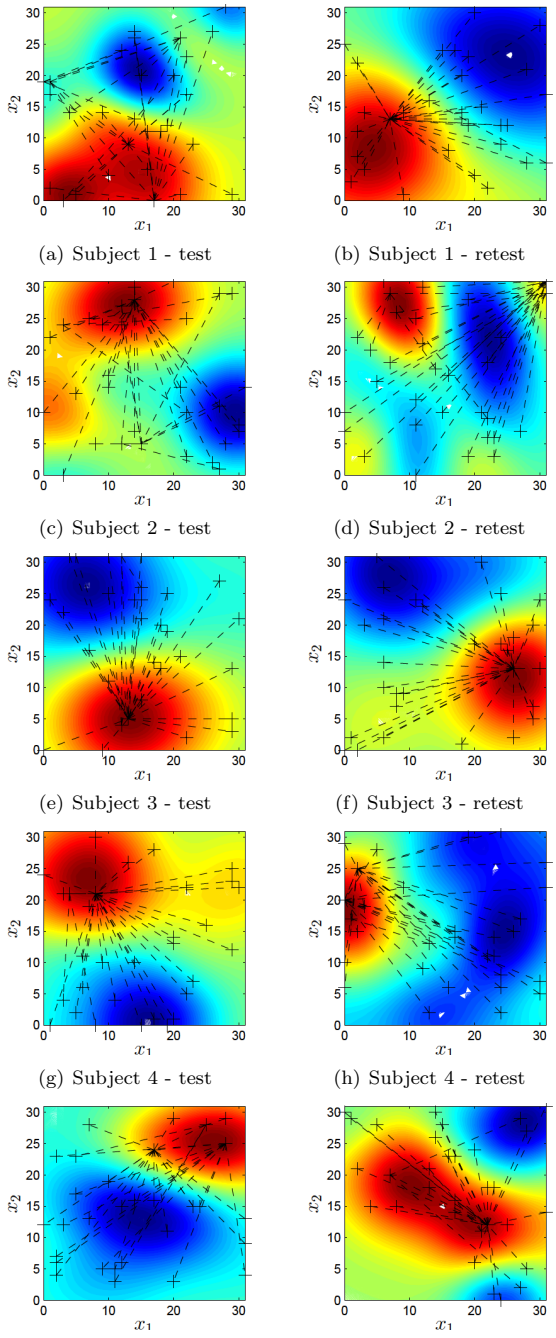
Figure 2: Audiograms of each individual subject. Crosses and circles correspond to the left and right ear, respectively.

The number of assessments that each subject needs to perform before the algorithm discovers a steady preferred setting is visualized in Fig. 4 (see caption for an explanation). Note, that subject 5 has not been included, since subject 5 did not prefer any setting over others and hence did not convergence.

5 Summary and Discussion

Overall, the reproducibility of the found preferred settings is satisfactory given the subtle differences between parameter settings and is found well before the 30th assessment. However, since the perceptual differences between settings are very subtle, it was not possible to prove or to reject significance of the preferred settings overall. However, the apparent good reproducibility indicates that the found preferred settings are actually a result of the subjects' individual preferences and not a results of a random effect.

The variability in the preferred settings across users from the results in Fig. 3 corroborates previous findings in the literature [13] that individual preferences



(a) Subject 1 - test (b) Subject 1 - retest

(c) Subject 2 - test (d) Subject 2 - retest

(e) Subject 3 - test (f) Subject 3 - retest

(g) Subject 4 - test (h) Subject 4 - retest

(i) Subject 5 - test (j) Subject 5 - retest

Figure 3: IRP as a function of the two HA parameters, x_1 and x_2, predicted by the fine-tuning algorithm after 30 comparisons for the test (left column) and re-test (right column) sessions. Red and blue colors indicate high and low preference regions, respectively. Crosses connected with a dashed line indicate comparisons. Note, the IRPs are unit-free.

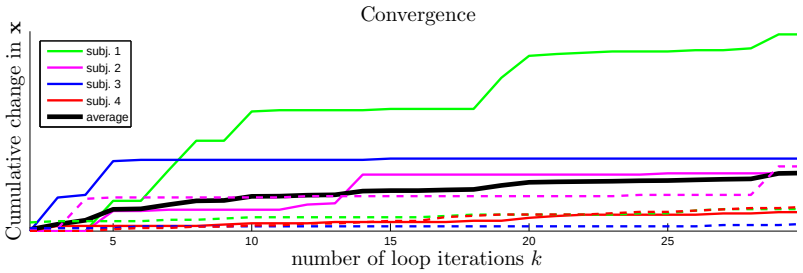


Figure 4: The cumulative euclidean change in the location of the maximum point of the predicted IPR after a each new assessment as a function of the number of assessments.

among HA users do exist, and the system proposed here discovers such preferences before the subjects have performed twenty comparisons in a worst case scenario (see Fig. 4). In case of parameter settings that are perceptually easier to distinguish, the required number of comparisons would presumably be even smaller.

The results presented here are preliminary and serve merely to visualize how the system works. In future work, especially the scaling issue with respect to the number of required comparisons in relation to the number of adjustable parameters is of interest. Also, a similar experiment should be conducted in the future, but with parameter settings that are easier to distinguish from each other, to verify that settings that are suggested by the system to be preferred are significantly different from settings that are not suggested to be preferred. Next, better convergence measures based on the actual statistics provided by the probabilistic modeling framework should be studied. One possible suggestion could be the mean of the EI across settings. Finally, investigation of suitable metrics for expressing the similarity between test/re-test results would be interesting. One (Bayesian) suggestion could be based on the likelihood of the test data given the re-test data or vice versa.

References

[1] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, 2006.

[2] B. S. Jensen, J. B. Nielsen, and J. Larsen, "Efficient Preference Learning with Pairwise Continuous Observations and Gaussian Processes," *IEEE Workshop MLSP, Beijing*, September 2011.

[3] D. R. Jones, "A Taxonomy of Global Optimization Methods Based on Response Surfaces," *Journal of Global Optimization*, vol. 21, no. 4, pp. 345–383, 2001.

[4] F. K. Kuk and N. M. C. Pape, "The reliability of a modified simplex procedure in hearing aid frequency-response selection," *J Speech Hear Res*, vol. 35, no. 2, pp. 418–429, 1992.

[5] D. Baskent, C. L. Eiler, and B. Edwards, "Using genetic algorithms with subjective input from human subjects: implications for fitting hearing aids and cochlear implants.," *Ear and hearing*, vol. 28, no. 3, pp. 370–80, June 2007.

[6] E. A. Durant, G. H. Wakefield, D. J. Van Tasell, and M. E. Rickert, "Efficient perceptual tuning of hearing aids with genetic algorithms," *Speech and Audio Processing, IEEE Transactions on*, vol. 12, no. 2, pp. 144–155, 2004.

[7] T. Heskes and B. de Vries, "Incremental utility elicitation for adaptive personalization," in *Proceedings of the 17th Belgium-Netherlands Conference on Artificial Intelligence, Brussels*. 2005, pp. 127–134, Citeseer.

[8] Adriana Birlutiu, Perry Groot, and Tom Heskes, "Multi-task preference learning with Gaussian processes," in *Proceedings of the 17th European Symposium on Artificial Neural Networks (ESANN)*, 2009, pp. 123–128.

[9] L. L. Thurstone, "A Law of Comparative Judgement," *Psychological Review*, vol. 34, 1927.

[10] R. D. Bock and J. V. Jones, "The Measurement and Prediction of Judgment and Choice," 1968.

[11] W. Chu and Z. Ghahramani, "Preference Learning with Gaussian Processes," *Proceedings of the 22nd International Conference on Machine Learning (ICML)*, pp. 137–144, 2005.

[12] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.

[13] K. H. Arehart, J. M. Kates, M. C. Anderson, and L. O. Harvey, "Effects of noise and distortion on speech quality judgments in normal-hearing and hearing-impaired listeners.," *The Journal of the Acoustical Society of America*, vol. 122, no. 2, pp. 1150–64, Aug. 2007.

Appendix E

Bounded Gaussian Process Regression

Bjørn Sand Jensen, Jens Brehm Nielsen, Jan Larsen. Bounded Gaussian Process Regression. Published in *2013 IEEE International Workshop on Machine Learning for Signal Processing*, 2013. ISSN 1551-2541. doi:10.1109/MLSP.2013.6661916.

Errata

- In the equation before Eq. (7), the denominator should be

$$\mathcal{N}\left(\Phi^{-1}(y^*)\right),$$

and not $\Phi(\Phi^{-1}(y^*))$.

Bounded Gaussian Process Regression

Bjørn Sand Jensen, Jens Brehm Nielsen and Jan Larsen

Department of Applied Mathematics and Computer Science,
Technical University of Denmark,
Matematiktorvet Building 303B, 2800 Kongens Lyngby, Denmark
{bjje,jenb,janla}@dtu.dk

Preprint

Abstract

We extend the Gaussian process (GP) framework for *bounded* regression by introducing two bounded likelihood functions that model the noise on the dependent variable explicitly. This is fundamentally different from the implicit noise assumption in the previously suggested warped GP framework. We approximate the intractable posterior distributions by the Laplace approximation and expectation propagation and show the properties of the models on an artificial example. We finally consider two real-world data sets originating from perceptual rating experiments which indicate a significant gain obtained with the proposed explicit noise-model extension.

1 Introduction

Regression is typically defined as learning a mapping from a possible multi-dimensional input to an effectively unbounded one-dimensional observational space, i.e., the space of the dependent variable. However, in many regression problems the observational space is clearly bounded. Examples of such problems include prediction of betting odds, data compression ratios and ratings from perceptual experiments. When the observational space is bounded, modeling the observations with a distribution having infinite support such as the Gaussian distribution, is clearly incorrect from a probabilistic point of view. In this work we will extend the GP framework to allow for principle modeling of such observations.

Gaussian processes (GPs) are currently considered a state-of-the-art Bayesian regression method due to its flexible and non-parametric nature. However, *bounded* regression with GPs has only indirectly been addressed by mapping or *warping* the bounded observations onto a latent unbounded space in which the observational noise can be assumed to be Gaussian [1]. Hereby, the observational model is only modeled implicitly through the warping function. In contrast, we consider observational models or likelihood functions that make assumptions about the noise directly in the observational space, and thus, model the observational noise explicitly.

Possibly, the simplest way to derive a bounded likelihood function is to use a truncated distribution. A natural choice is to use the truncated version of the Gaussian distribution considered in this work. Alternatively, a bounded likelihood function could be derived from a distribution that only has finite support. Of this type, we will consider the beta distribution and derive a bounded likelihood function based on a re-parameterization. For both models we perform inference and predictions based on the Laplace approximation and expectation propagation (EP).

Employing a toy example, we compare the predictive distributions of warped GPs with regression based on the bounded likelihood functions mentioned above. We show that, as expected, the model with the correct noise assumption provides the best expected predictive negative log likelihood (or, alternatively, generalization error). Two examples are used to justify the models in real-world regression scenarios and they show that the two likelihood models provide better model fits compared to the warped GP.

2 Gaussian Process Regression

A Gaussian process (GP) is a stochastic process defined as a collection of random variables, any finite subset of which must have a joint Gaussian distribution. In effect, we may place the GP as a prior over any finite set of functional values $\mathbf{f} = [f_1, f_2, ..., f_n]^\top$, where $f_i = f(\mathbf{x}_i)$, resulting in a finite multivariate (zero-mean) Gaussian distribution over the set as $p(\mathbf{f}|\mathcal{X}, \boldsymbol{\theta}_c) = \mathcal{N}(\mathbf{f}|\mathbf{0}, \mathbf{K})$, where each element of the covariance matrix $[\mathbf{K}]_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j)_{\boldsymbol{\theta}_c}$ is given by a covariance function $k(\cdot,\cdot)_{\boldsymbol{\theta}_c}$ with parameters $\boldsymbol{\theta}_c$, and where $\mathcal{X} = \{\mathbf{x}_i | i = 1, ..., n\}$ denotes the set of inputs. The GP is effectively used as a prior over functions in non-parametric Bayesian regression frameworks where either the outputs or a likelihood can be parameterized by a smooth and continuous function $f(\cdot)$. In the simplest case the set of observations, $\mathcal{Y} = \{y_i | i = 1, ..., n\}$, consists of the functional values themselves with added i.i.d Gaussian noise with variance σ_n^2. Hereby, the likelihood function is a standard Gaussian likelihood function parameterized by $f(\cdot)$ defining the mean. Hence, $p(y_i|f_i, \boldsymbol{\theta}_\mathcal{L}) = \mathcal{N}(y_i|f_i, \sigma^2)$.

Bayes formula gives us—regardless of the likelihood function—the posterior distribution,

$$p(\mathbf{f}|\mathcal{Y}, \mathcal{X}, \boldsymbol{\theta}) = \frac{p(\mathcal{Y}|\mathbf{f}, \boldsymbol{\theta}_\mathcal{L}) p(\mathbf{f}|\mathcal{X}, \boldsymbol{\theta}_c)}{p(\mathcal{Y}|\mathcal{X}, \boldsymbol{\theta})},$$

where it is typically assumed that the likelihood factorizes over instances such that $p(\mathcal{Y}|\mathbf{f}, \boldsymbol{\theta}_\mathcal{L}) = \prod_{i=1}^n p(y_i|f_i, \boldsymbol{\theta}_\mathcal{L})$. The denominator, $p(\mathcal{Y}|\mathcal{X}, \boldsymbol{\theta})$, is called the *marginal likelihood* or *evidence* given as $p(\mathcal{Y}|\mathcal{X}, \boldsymbol{\theta}) = \int p(\mathcal{Y}|\mathbf{f}, \boldsymbol{\theta}_\mathcal{L}) p(\mathbf{f}|\mathcal{X}, \boldsymbol{\theta}_c) d\mathbf{f}$. In empirical Bayesian methods the evidence is used to learn point estimates of both likelihood function and prior parameters $\boldsymbol{\theta} = \{\boldsymbol{\theta}_c, \boldsymbol{\theta}_\mathcal{L}\}$.

Provided that the likelihood is Gaussian, both the posterior and predictive distribution will be Gaussian (processes) available in closed form [2, Chapter 2]. However, not all real-world problems actually justify the observations to be Gaussian distributed. As mentioned, we consider *bounded* observations, meaning that they in contrast to Gaussian distributed observations do not have infinite support.

3 Bounded Likelihood Functions

We consider a set $\mathcal{Y} = \{y_i | i = 1, ..., n\}$ of bounded responses $y_i \in]a, b[$ to an input $\mathbf{x}_i$. In the following we will present three different observational models for this type of response. The first is the warped GP [1], where the likelihood describes warped observations rather than the bounded responses directly. Following this, we propose two different likelihood functions that directly model the bounded responses in a principle probabilistic fashion by assuming particular distributions of the observations defining the noise in the original bounded domain.

3.1 Warping

Snelson *et. al* [1] learn a warping, that transforms the original data $\mathcal{Y}$ into a form where the data is modeled by a traditional GP with a Gaussian noise model. Here, we will not consider how to learn the correct warping, but instead use a fixed warping that transforms the bounded responses y_i into unbounded versions z_i. Several warping functions would apply, but to allow for direct comparison of all the models we use the inverse cumulative Gaussian (probit) $\Phi^{-1}(\cdot)$—with zero mean and unity variance—such that $z_i = \Phi^{-1}(y_i)$. The resulting model will be referred to as GP-WA.

3.2 Truncated Distributions

The simplest route to a bounded likelihood function is to use distributions with infinite support and truncate them to the bounded domain. There are a number of relevant distributions including the truncated student-t and of course the truncated Gaussian (TG) distribution, see e.g. [3]. As a representative for this type of bounding approach, we consider the TG and define the corresponding likelihood function as

$$\mathcal{L}_{TG} \equiv p\left(y_i | f_i, \boldsymbol{\theta}_{\mathcal{L}}\right) = \frac{\nu \mathcal{N}\left(\nu\left(y_i - \mathrm{M}\left(f_i\right)\right)\right)}{\Phi\left(\nu\left(b - \mathrm{M}\left(f_i\right)\right)\right) - \Phi\left(\nu\left(a - \mathrm{M}\left(f_i\right)\right)\right)}, \tag{1}$$

where the distribution is parameterized by the mode $\mathrm{M}(f_i)$ and the domain limits a and b which we assume to be 0 and 1, respectably[1]. The mean of the TG distribution is given by

$$\mu\left(f_i\right) = \mathrm{M}\left(f_i\right) + \frac{1}{\nu} \frac{\mathcal{N}\left(\nu\left(a - \mathrm{M}\left(f_i\right)\right)\right) - \mathcal{N}\left(\nu\left(b - \mathrm{M}\left(f_i\right)\right)\right)}{\Phi\left(\nu\left(b - \mathrm{M}\left(f_i\right)\right)\right) - \Phi\left(\nu\left(a - \mathrm{M}\left(f_i\right)\right)\right)}. \tag{2}$$

Eq. 2 in effect leaves two parametrization options in the sense that we may select the non-parametric function, $f(\cdot)$, to parameterize either the mode or the mean function. Both options are valid from a modeling perspective, but the easiest parametrization is by far the mode, $\mathrm{M}(f_i)$. For prediction speed it may be beneficial to indirectly parameterize the mean, but then the (unique) solution to the mode given the mean must be found numerically or approximately. The numerical approach will severely limit the effectiveness of the posterior approximation and in this work we will therefore focus on the mode parametrization for

[1] We note that the truncated student-t has the same form as the TG and can easily be realized using the methods and implementations presented in this work.

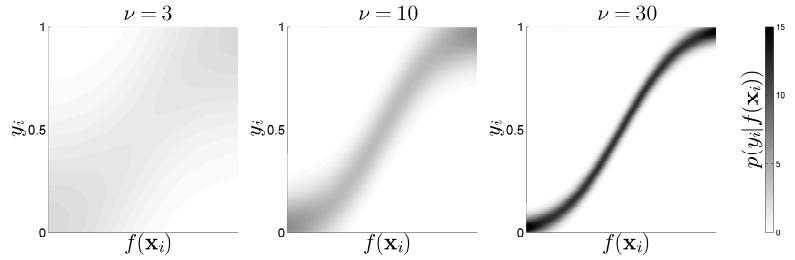


Figure 1: Illustration of the proposed TG likelihood function with $p(y_i|f_i)$ shown as a gray-scale level. Left: $\nu = 3$, Middle: $\nu = 10$ and Right: $\nu = 30$.

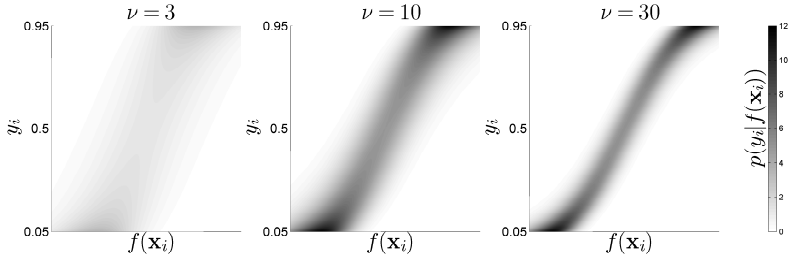


Figure 2: Illustration of the proposed beta likelihood function with $p(y_i|f_i)$ shown as a gray-scale level. Left: $\nu = 3$, Middle: $\nu = 10$ and Right: $\nu = 30$.

the TG. Thus, the likelihood function in Eq. 1 is parameterized by the mode as follows $\mathrm{M}(f_i) = \Phi(f_i)$ and the resulting model depicted in Fig. 1 will be referred to as GP-TG

3.3 Beta

A distribution that imposes bounded responses in a completely natural manner is the beta distribution which has also been applied in standard parametric settings [4, 5]. The beta distribution is therefore an obvious distribution for the bounded observations and we select a parametrization which expresses the shape parameters, α, β, of the beta distribution, $\mathrm{Beta}(\alpha, \beta)$, in terms of the mean μ such that

$$\alpha = \nu\mu, \qquad \beta = \nu\,(1-\mu)\,.$$

We then parameterize the mean μ of the beta distribution by the cumulative Gaussian, such that $\mu(f_i) = \Phi(f_i)$. The re-parameterized beta likelihood depicted in Fig. 2 is thereby given by

$$\mathcal{L}_{\mathrm{BE}} \equiv p(y_i|f_i, \boldsymbol{\theta}_{\mathcal{L}}) = \mathrm{Beta}(y_i|\nu\Phi(f_i), \nu\,(1-\Phi(f_i))),$$

and will be referred to as the GP-BE model. Note, that the ν parameter is an (inverse) dispersion parameter.

4 Approximate Inference and Prediction

For the GP-WA model the likelihood is effectively Gaussian, hence, inference is analytical tractable [1]. However, neither the GP-TG model nor the GP-BE model have analytical tractable posterior distributions. Instead, we must resort to approximations. We consider two different approximate inference schemes—the Laplace approximation and expectation propagation (EP). Both methods approximate the posterior distribution $p(\mathbf{f}|\mathcal{Y},\mathcal{X},\boldsymbol{\theta})$ with a single Gaussian $q(\mathbf{f})$. In the following we briefly give an overview of the two approximate inference schemes in relations to the bounded likelihood functions. For more details on the approximation schemes see for instance [2].

4.1 Laplace Approximation

Possibly, the simplest inference method is the Laplace approximation in which a multivariate Gaussian distribution is used to approximate the posterior, such that $p(\mathbf{f}|\mathcal{X},\mathcal{Y},\theta) \approx q(\mathbf{f}) = \mathcal{N}(\mathbf{f}|\hat{\mathbf{f}},\mathbf{A}^{-1})$, where $\hat{\mathbf{f}}$ is the mode of the posterior and $\mathbf{A}$ is the Hessian of the negative log posterior at the mode. The mode is found as $\hat{\mathbf{f}} = \arg\max_{\mathbf{f}}\ p\left(\mathbf{f}|\mathcal{Y},\mathcal{X},\boldsymbol{\theta}\right) = \arg\max_{\mathbf{f}}\ p\left(\mathcal{Y}|\mathbf{f},\boldsymbol{\theta}_{\mathcal{L}}\right) p\left(\mathbf{f},\mathcal{X},\boldsymbol{\theta}_{c}\right)$. The general solution to the problem can be found by considering the un-normalized log posterior and the resulting cost function which is to be maximized, is given by

$$\psi\left(\mathbf{f}|\mathcal{Y},\mathcal{X},\boldsymbol{\theta}\right) = \log p\left(\mathcal{Y}|\mathbf{f},\mathcal{X},\boldsymbol{\theta}_{\mathcal{L}}\right) - \frac{1}{2}\mathbf{f}^{T}\mathbf{K}^{-1}\mathbf{f} - \frac{1}{2}\log|\mathbf{K}| - \frac{N}{2}\log 2\pi,$$

where $\mathbf{K}_{i,j} = k(\mathbf{x}_i,\mathbf{x}_j)_{\boldsymbol{\theta}_c}$. The maximization can be solved with a standard Newton-step algorithm given by

$$\hat{\mathbf{f}}^{new} = \left(\mathbf{K}^{-1} + \mathbf{W}\right)^{-1} \cdot \left[\mathbf{W}\hat{\mathbf{f}} + \nabla \log p(\mathcal{Y}|\mathbf{f},\mathcal{X},\boldsymbol{\theta}_{\mathcal{L}})\right],$$

where the Hessian $\mathbf{W} = -\nabla\nabla_{\mathbf{f}} \log p(\mathcal{Y}|\mathbf{f})$ is diagonal with elements defined by the second derivative of the log-likelihood function $[\mathbf{W}]_{i,i} = -\frac{\partial^2 \log p(y_i|f_i)}{\partial f_i^2}$. When converged, the resulting approximation is

$$p\left(\mathbf{f}|\mathcal{Y},\mathcal{X},\boldsymbol{\theta}\right) \approx \mathcal{N}\left(\mathbf{f}|\hat{\mathbf{f}},\boldsymbol{\Sigma}\right), \quad \text{where } \boldsymbol{\Sigma} = \left(\mathbf{W} + \mathbf{K}^{-1}\right)^{-1}.$$

Approximating the posterior of $\mathbf{f}$ by the Laplace approximation requires the first two derivatives of the log likelihood. For the TG we will report the general derivatives applicable for any truncated likelihood function based on symmetric densities for which the truncated density can be written as the TG, i.e. in the form

$$p\left(y_i|f_i\right) = \frac{r\left(g\left(y_i|f_i\right)\right)}{s\left(g\left(b|f_i\right)\right) - s\left(g\left(a|f_i\right)\right)}, \tag{3}$$

where we for the TG model defines $g\left(c|f_i\right) = \nu\left(c - \mathrm{M}\left(f_i\right)\right)$. The resulting derivatives for the TG likelihood requires the following partial derivatives

$$\begin{aligned}
\frac{\partial r\left(\cdot\right)}{\partial f_i} &= \nu^2 g\left(y_i\right)\mathcal{N}\left(g\left(y_i\right)\right)\mathcal{N}\left(f_i\right), \\
\frac{\partial^2 r(\cdot)}{\partial^2 f_i} &= \nu^2\mathcal{N}\left(g\left(y_i\right)\right)\mathcal{N}\left(f_i\right)\left[-\nu\mathcal{N}\left(f_i\right) + g\left(y_i\right)\left(\nu g\left(y_i\right)\mathcal{N}\left(f_i\right) - f_i\right)\right], \\
\frac{\partial s\left(\cdot\right)}{\partial f_i} &= -\nu\mathcal{N}\left(g\left(b\right)\right)\mathcal{N}\left(f_i\right) \quad \text{and} \\
\frac{\partial^2 s(\cdot)}{\partial^2 f_i} &= -\nu\mathcal{N}\left(g\left(b\right)\right)\mathcal{N}\left(f_i\right)\left[\nu g\left(b\right)\mathcal{N}\left(f_i\right) - f_i\right],
\end{aligned}$$

which enter into the derivatives of Eq. 3. The two required partial derivatives for the beta distribution are given by

$$\begin{aligned}
\frac{\partial \log \mathrm{Beta}(y_i|\cdot)}{\partial f_i} &= \nu\mathcal{N}\left(f_i\right)\cdot\left[\log(y_i) - \log(1-y_i) - \psi(\alpha) + \psi(\beta)\right] \quad \text{and} \\
\frac{\partial^2 \log \mathrm{Beta}(y_i|\cdot)}{\partial f_i^2} &= -\nu^2\mathcal{N}\left(f_i\right)\cdot\Big[\mathcal{N}\left(f_i\right)\cdot\left(\psi^{(1)}(\alpha) + \psi^{(1)}(\beta)\right) \\
&\qquad + \frac{f_i}{\nu}\cdot\left(\log(y_i) - \log(1-y_i) - \psi(\alpha) + \psi(\beta)\right)\Big],
\end{aligned}$$

where $\psi(\cdot)$ and $\psi^{(1)}(\cdot)$ are the digamma function of zero'th and first order, respectively.

4.2 Expectation Propagation

EP also approximates the posterior distribution with a single multivariate Gaussian distribution $q(\mathbf{f}) = \mathcal{N}(\mathbf{f}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ by factorizing the likelihood by n Gaussian factors $t_i(f_i|\tilde{Z}_i, \tilde{\mu}_i, \tilde{\Sigma}_i) = \tilde{Z}_i\mathcal{N}(f_i|\tilde{\mu}_i, \tilde{\Sigma}_i)$, where $i = 1, ..., n$. The EP approximation to the full posterior is thus given by

$$p(\mathbf{f}|\mathcal{Y}, \mathcal{X}, \boldsymbol{\theta}) \approx q(\mathbf{f}) = \mathcal{N}(\mathbf{f}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = p(\mathbf{f}, \mathcal{X})\mathcal{N}(\mathbf{f}|\tilde{\boldsymbol{\mu}}, \tilde{\boldsymbol{\Sigma}})\prod_{i=1}^{n}\tilde{Z}_i,$$

where the means $\tilde{\mu}_i$ and variances $\tilde{\Sigma}_i$ have been collected into the vector $\tilde{\boldsymbol{\mu}}$ and diagonal matrix $\tilde{\boldsymbol{\Sigma}}$, respectively. The mean and covariance of the approximation are given by

$$\boldsymbol{\mu} = \boldsymbol{\Sigma}\tilde{\boldsymbol{\Sigma}}^{-1}\tilde{\boldsymbol{\mu}}, \qquad \boldsymbol{\Sigma} = \left(\mathbf{K}^{-1} + \tilde{\boldsymbol{\Sigma}}^{-1}\right)^{-1}.$$

EP updates each factor t_i in turn by first removing the factor to yield what is called the *cavity distribution* $q_{-i}(f_i) = \mathcal{N}(f_i|\mu_{-i}, \Sigma_{-i})$, where $\mu_{-i} = \Sigma_{-i}([\boldsymbol{\Sigma}]_{i,i}^{-1}\mu_i - \tilde{\Sigma}_i^{-1}\tilde{\mu}_i)$ and $\Sigma_{-i} = ([\boldsymbol{\Sigma}]_{i,i}^{-1} - \tilde{\Sigma}_i^{-1})^{-1}$. Secondly, the factor t_i is updated by projecting the cavity distribution multiplied with the true likelihood term onto a univariate Gaussian. The projection is effectively done

by solving the following three integrals

$$Z_i = \int p(y_i|f_i)\mathcal{N}(f_i|\mu_{-i}, \Sigma_{-i})df_i, \tag{4}$$

$$\frac{dZ_i}{d\mu_{-i}} = \frac{d}{d\mu_{-i}} \int p(y_i|f_i)\mathcal{N}(f_i|\mu_{-i}, \Sigma_{-i})df_i$$
$$= \int p(y_i|f_i) \frac{d}{d\mu_{-i}} \left\{\mathcal{N}(f_i|\mu_{-i}, \Sigma_{-i})\right\} df_i, \tag{5}$$

$$\frac{d^2Z_i}{d\mu_{-i}^2} = \frac{d^2}{d\mu_{-i}^2} \int p(y_i|f_i)\mathcal{N}(f_i|\mu_{-i}, \Sigma_{-i})df_i$$
$$= \int p(y_i|f_i) \frac{d^2}{d\mu_{-i}^2} \left\{\mathcal{N}(f_i|\mu_{-i}, \Sigma_{-i})\right\} df_i. \tag{6}$$

Neither the beta likelihood nor the TG likelihood yield analytical tractable solutions for these three integrals, but the one-dimensional integrals can be solved numerically for the EP inference.

4.3 Predictive Distributions

Naturally, we want to predict future values of both the latent functional value f^* and data label y^*. For all models the posterior distribution over $\mathbf{f}$ is effectively Gaussian[2]. Hence, the predictive distribution $p(f^*|\mathcal{Y}, \mathcal{X}, \mathbf{x}^*) = \mathcal{N}(f^*|\mu^*, \sigma_*^2)$ of latent functional values is Gaussian and is derived just as in the standard cases in a straight forward manner (see e.g. [2, Chapter 2-3]).

The predictive distribution of future targets $p(y^*|\mathcal{Y}, \mathcal{X}, \mathbf{x}^*)$ involves computing the integral

$$p(y^*|\mathcal{Y}, \mathcal{X}, \mathbf{x}^*) = \int p(y^*|f^*)\mathcal{N}(f^*|\mu^*, \sigma_*^2)df^*.$$

For the GP-WA, the predictive distribution has a closed-form solution [1]

$$p_{\text{GP-WA}}(y^*|\mathcal{Y}, \mathcal{X}, \mathbf{x}^*) = \frac{\mathcal{N}(\Phi^{-1}(y^*)|\mu^*, \sigma_*^2)}{\mathcal{N}(\Phi^{-1}(y^*))}.$$

In case of the GP-BE and GP-TG the predictive distribution is not given in closed form. Instead, the integral must be computed using numerical methods. Predictions of the mean, $\mathbb{E}(y) \in \,]\,0; 1\,[$, are in the bounded case given by

$$\mathbb{E}_{p(y^*|\cdot)}\{y^*\} = \int_0^1 y^* p(y^*|\mathcal{Y}, \mathcal{X}, \mathbf{x}^*)dy^* \tag{7}$$
$$= \int \mathcal{N}(f^*|\mu^*, \sigma_*^2) \int_0^1 y^* p(y^*|f^*)dy^* df^*$$
$$= \int \mathcal{N}(f^*|\mu^*, \sigma_*^2)\mathbb{E}_{p(y^*|f^*)}\{y^*\}df^*. \tag{8}$$

Given the cumulative Gaussian warping, Eq. 7 can be solved analytically for the GP-WA model. In Eq. 8 the mean of the likelihood occurs, which in the beta

[2]For the warped GP the posterior is exactly Gaussian, whereas we for the two other models have approximated—either by Laplace or EP—the posterior with a Gaussian.

case is parameterized by a cumulative Gaussian and given the specific choice of warping this results in a closed form solution expressed by[3]

$$\mathbb{E}_{\text{GP-WA}}\{y^*\} = \mathbb{E}_{\text{GP-BE}}\{y^*\} = \Phi\left(\frac{\mu^*}{\sqrt{1+(\sigma^*)^2}}\right).$$

In case of the GP-TG model, Eq. 7 has no analytical form and must be solved by one-dimensional numerical approximation.

5 Simulation Example

In order to illustrate the difference between the warped and bounded likelihood approaches we consider an artificial example with added noise. It is generated by drawing a one-dimensional function from a zero-mean Gaussian process with a squared exponential (SE) kernel with length scale, $\sigma_l = 1$, and noise variance $\sigma_f = \exp(1)$. Three different types of noise are then added: The first type (WA) is i.i.d Gaussian noise added directly on f and transformed through $\Phi(\cdot)$ which corresponds to the noise assumption in the warped GP. In the second case (TG), f is transformed through $\Phi(\cdot)$ before adding noise based on the mode-parameterized TG distribution, thus corresponding to the noise assumption of the TG likelihood. In the third case (BE), we add noise based on the mean-parameterized beta distribution.

In order to visualize the special nature of bounded responses and the difference between the models, we have illustrated the WA noise case in Fig. 3, where all three bounded models are evaluated. Both the Laplace approximation and EP have been used for inference for the beta and TG model. The hyper-parameters are in all cases optimized using evidence maximization. The main difference of the three models occurs at the domain boundaries, where the GP-WA model concentrates the entire mass almost at the boundary. The predictive distribution of the GP-TG model generally has a similar shape over the entire domain with its mean always spaced significantly far from the boundary, whereas the GP-BE can also have its mean very close to the boundary as for the GP-WA model, but still retain mass away from the boundary. No significant differences between the two inference schemes are evident. Since the EP scheme requires numerical solutions to the integrals in Eq. 4-6, the Laplace approximation will be used in the reminder of this article.

We evaluate the ability of the models to model different noise distributions by comparing the predictive log likelihood for the previously mentioned dataset based on the Laplace approximation. A second example is added in which the function is drawn from a GP with a periodic covariance function. The predictive log likelihood for both examples is reported in Tab. 5 and is the average over ten realizations of the noise. As expected, we see that the model corresponding to the added noise type always results in the lowest negative likelihood, indicating a better model fit.

[3]Keep in mind that although there is an equal sign between the predictive mean of the cumulative-warped and the beta model, the means will in general be different due to difference in the *latent* predictive distributions of the GP.

	Squared Exponential ($\sigma_f^2 = 2$, $\ell = 1$)		
	WA	TG	BE
GP-WA	**-129.8** (6.4)	-82.0 (7.3)	-165.3 (31.1)
GP-TG	-91.0 (19.6)	**-96.8 (4.5)**	-81.8 (14.5)
GP-BE	-119.8 (7.7)	-91.2 (6.3)	**-195.2 (24.6)**
	Periodic ($\sigma_f^2 = 3$, $\ell = 0.8$, $\lambda = 5$)		
	WA	TG	BE
GP-WA	**-93.2 (6.9)**	-80.6 (11.0)	-70.8 (10.4)
GP-TG	-76.2 (10.3)	**-91.6 (9.4)**	-66.0 (12.9)
GP-BE	-88.5 (3.8)	-84.5 (7.8)	**-99.8 (15.5)**

Table 1: Expected predictive negative log likelihood (and standard deviation) for each of the three models (GP-WP, GP-TG, GP-BE)) evaluated on a specific function with additive noise from ten random realizations of the noise for each corresponding noise types: WA, BE and TG. The noise free function is drawn from a GP prior with the indicated covariance functions and parameter values (defined in [6])

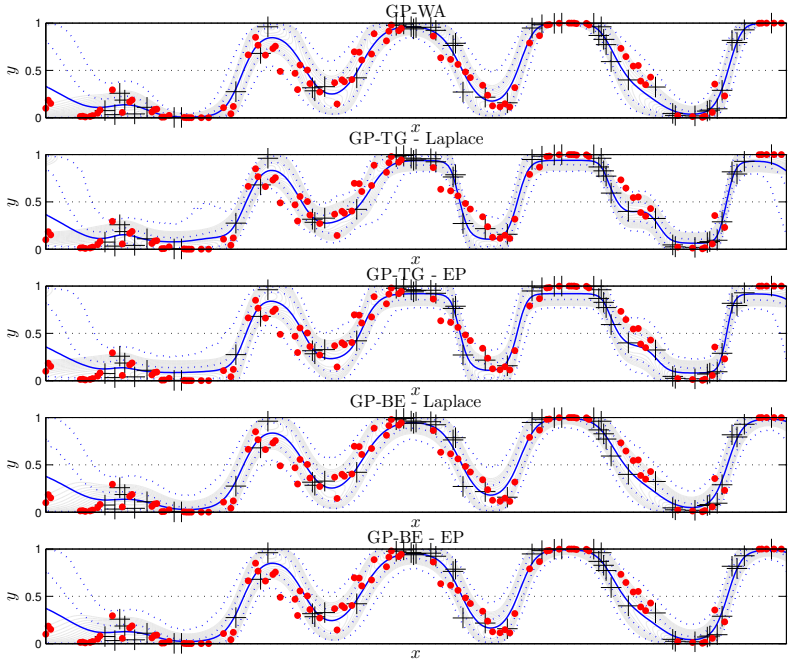


Figure 3: Predictive distributions for the three models: GP-WA, GP-TG and GP-BE. For GP-TG and GP-BE both Laplace and EP inference are shown, where training data: $+$, test examples: $\cdot$, predictive mean: $-$ and 68% and 95% percentiles: $\cdots$. Also, contours of the predictive distribution are shown in gray, where the intensity reflects probability mass concentration.

6 Perceptual Audio Evaluations

In order to demonstrate the difference between the three considered models in a real-world scenario, we have tested the three models on two datasets consisting of subjective ratings performed while listening to audio through a hearing aid (HA) compressor with different settings.

The first dataset [7], HA-I, contains six compression ratio settings (including one without compression) and three release-time settings. This results in sixteen non-trivial combinations of the settings with $\mathbf{x}^s \in \mathbb{R}^2$, that are rated three times by each of the seven test subjects, u, while listening to a speech signal. The dataset also contains an complete six point audiogram on both left and right ear, $\mathbf{x}^u \in \mathbb{R}^{2\times 6}$, of the hearing impaired test subjects. The audio signal resulting from each compressor setting is represented by standard audio features, namely thirty Mel frequency cepstral coefficients, $\mathbf{x}^a \in \mathbb{R}^{30}$. Thus, for one setting, s, each test subject, u, rated the audio signal, a. This results in a collection of inputs for this specific rating which we collect in $\mathbf{x} = \{\mathbf{x}^u, \mathbf{x}^a, \mathbf{x}^s\}$. We use the multi-task kernel formulation [8] and define the covariance function as $k(\mathbf{x}_i, \mathbf{x}_j) = k^u_{\text{SE-ARD}}\left(\mathbf{x}^u_i, \mathbf{x}^u_j\right)\left(k^a_{SE}\left(\mathbf{x}^a_i, \mathbf{x}^a_j\right) + k^s_{SE}\left(\mathbf{x}^s_i, \mathbf{x}^s_j\right)\right)$ where all covariance functions are squared exponential (SE), the first one with automatic relevance determination (SE-ARD).

The second dataset [9], HA-II, contains three input settings related to the compression ratio, attack time and release time of a HA dynamic range compressor, thus $\mathbf{x}^s \in \mathbb{R}^3$. Four subjects have rated 50 combinations of inputs in relation to general preference while listening to a speech-in-background-noise signal. The dataset does not contain any data describing the subjects, hence we use only a single squared exponential covariance function.

We initialize the hyper-parameters in the (common) covariance function to the same value for all models, but initialize the likelihood noise parameter with multiple values in a grid pattern after which all the hyper-parameters are optimized using evidence maximization. We then report the performance of the model which yields the largest evidence after maximization. For the purpose of comparing the three models, we will simply consider the Laplace approximation and a retest scenario in which we train on a random repetition and test on another repetition for each setting. We repeat this three times and evaluate the resulting predictive likelihood and mean square error (MSE). The results are listed in Tab. 6. We note from the negative predictive log likelihood that the beta distribution provides a better fit to the noise compared to the other two models given the two real-world datasets presented here.

7 Discussion and Conclusion

In the present work, we outlined two bounded likelihood functions for bounded Gaussian process regression which in contrast to previous work make explicit assumptions about the noise in the bounded observation space. In the two considered examples we found the beta model to be better than the two other models in terms of the predictive log likelihood. These results together with the artificial examples support the application of all three models in the non-parametric Gaussian process framework. However, the optimal model obviously depends on the actual noise distribution in a given application. We therefore

		GP-WA	GP-TG	GP-BE
HA-I	-log $p(y^*)$	-66.1	-96.1	**-101.2**
	MSE	0.013	0.001	0.010
HA-II	-log $p(y^*)$	-7.7	-9.3	**-14.1**
	MSE	0.031	0.030	0.035

Table 2: **HA-I** Mean square error (MSE) and expected predictive negative log likelihood over 10 random sets. We find a significant difference in log likelihood at the 5% level between GP-TG and the two other models but not between GP-TG and GP-BE. For MSE the only significant difference is between GP-TG and GP-BE. **HA-II** Mean square error (MSE) and negative log likelihood over 10 folds. Considering the negative log likelihood only the GP-BE is significantly better than the GP-WA in a paired t-test. There is no significant difference between GP-TG and the other models. The GP-BE is significantly different in terms of MSE than the two others.

foresee addition and inclusion of other noise models based on other distribution with finite support.

Implementations of the various likelihoods are available [10] for use in the `gpml` toolbox [6] and can easily be extended to support more advanced link functions [11], which will make the models (both the bounded and the warped) even more flexible. In particular, we suggest to use a mixture of cumulative Gaussian link functions which do not complicate predictions significantly. Furthermore, we suggest to evaluate the performance of the deterministic approximations by the use of MCMC-sampling methods.

In conclusion, we have extended the Gaussian process framework to include bounded likelihood functions allowing for explicit specification of the likelihood model in applications where bounded observations are present and support an explicit noise model.

Acknowledgement: This work was supported in part by the Danish Council for Strategic Research of the Danish Agency for Science Technology and Innovation under the CoSound project, case number 11-115328.

References

[1] E. Snelson, C. E. Rasmussen, and Z. Ghahramani, "Warped Gaussian Processes," in *Advances in Neural Information Processing Systems*, vol. 16. MIT Press, 2004.

[2] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, 2006.

[3] N. L. Johnson, S. Kotz, and N. Balakrishnan, *Continuous Univariate Distributions*, vol. 1 & 2, Wiley, 2nd edition, 1994-1995.

[4] S. Ferrari and F. Cribari-Neto, "Beta Regression for Modelling Rates and Proportions," *Journal of Applied Statistics*, vol. 31, no. 7, pp. 799–815, Aug. 2004.

[5] M. Smithsen and J. Verkuilen, "A Better Lemon-squeezer? Maximum Likelihood Regression with Beta-distributed Dependent Variables.," *Australian Journal Psychology*, vol. 57, pp. 98–98, 2005.

[6] C. E. Rasmussen and H. Nickisch, "Matlab GPML Toolbox," 2010.

[7] E. Schmidt, *Hearing Aid Processing of Loud Speech and Noise Signals: Consequences for Loudness Perception and Listening Comfort.*, Ph.D. thesis, Technical University of Denmark, 2006.

[8] E. Bonilla, K. M. Chai, and C. Williams, "Multi-task gaussian process prediction," in *Advances in Neural Information Processing Systems*, vol. 20, pp. 153–160. MIT Press, 2008.

[9] J. B. Nielsen, "Preference based personalization of hearing aids," M.Sc. Thesis, 2010.

[10] J. B. Nielsen and B. S. Jensen, "Bounded Gaussian Process Regression - Supplementary Material," `www.imm.dtu.dk/pubdb/p.php?6683`, 2013.

[11] T. C. Martins Dias and C. A. R. Diniz, "The use of Several Link Functions on a Beta Regression Model: a Bayesian Approach.," *AIP Conference Proceedings*, vol. 1073, no. 1, pp. 144, 2008.

Appendix F

Personalized Audio Systems - a Bayesian Approach

Jens Brehm Nielsen, Bjørn Sand Jensen, Toke Jansen Hansen, Jan Larsen. Personalized Audio Systems - A Bayesian Approach. Published in *AES 135th Convention*, paper 9000, 2013. ISBN 978-0-937803-95-0.

Personalized Audio Systems - a Bayesian Approach

Jens Brehm Nielsen[1,2], Bjørn Sand Jensen[1],
Toke Jansen Hansen[1] and Jan Larsen[1]

Technical University of Denmark, DTU Compute,
Matematiktorvet 303B, DK-2800 Kongens Lyngby,
{jenb, bjje, tjha, janla}@dtu.dk

Widex A/S, Nymøllevej 6, DK-3540 Lynge, jeb@widex.com

Preprint

Abstract

Modern audio systems are typically equipped with several user adjustable parameters unfamiliar to most users listening to the system. To obtain the best possible setting, the user is forced into multi-parameter optimization with respect to the users's own objective and preference. To address this, the present paper presents a general inter-active framework for personalization of such audio systems. The framework builds on Bayesian Gaussian process regression in which a model of the users's *objective function* is updated sequentially. The parameter setting to be evaluated in a given trial is selected by model-based sequential experimental design. A Gaussian process model is proposed which incorporates correlation among particular parameters providing better modeling capabilities compared to a standard model. A five-band equalizer is considered for demonstration purposes, in which the parameters are optimized using the proposed framework. Twelve test subjects obtain a personalized setting with the framework, and these settings are significantly preferred to those obtained with random experimentation.

1 Introduction

The ever increasing number of features and processing possibilities in many modern multimedia systems, such as personal computers, mobile phones, hearing aids and home entertainment systems, has made it possible for users to customize these systems significantly. A downside in this trend is the large number of user-adjustable parameters which makes it a daunting and complex task to actually adjust/optimize the systems optimally. This is because users have to navigate in a high-dimensional parameter space, which makes it extremely difficult for users to find even a local optimum. For audio systems, the optimization is further complicated by perceptual and cognitive aspects of the

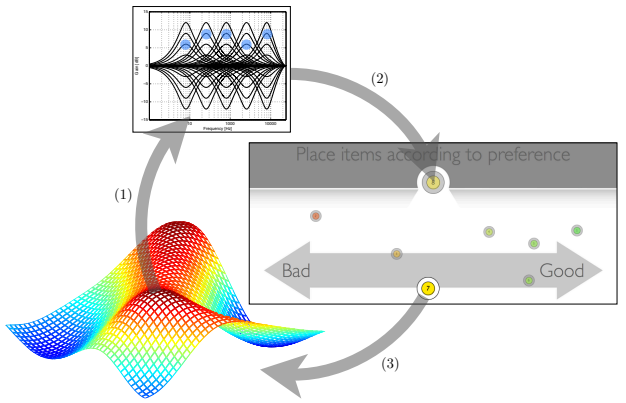


Figure 1: A conceptual overview of the interactive system. At step (1) we draw a new EQ from the current estimate of the user's objective function. Next, at step (2) this particular EQ is associated with a *ball*, in this case number *eight*, in the visualized user interface. Finally, after the user has rated the new EQ, the objective function is updated to reflect current positions of all previous *balls*, this update occurs at step (3). We emphasize that the user at any time may select between previously sampled EQ by clicking the *balls*, making the current song play through the newly selected EQ.

human auditory and cognitive system, which result in a significant spread in users's opinions concerning the adjustment of a particular system. It is therefore of great interest to find and evaluate fast and flexible tools for robustly optimizing user-adjustable parameters, with the aim to rapidly obtain a truly personalized audio system setting.

Prime examples of complex audio systems are hearing aids, where hundreds of parameters make up a unique and personal experience. It is therefore natural that this field has considered ways to learn an optimal setting based on preference (Kuk *et. al.* [8] and Baskent *et. al.* [1]), although these are currently based on non-probabilistic methods. Recently—and the closest related to our approach—Birlutiu *et. al.* [4, 3] have proposed two probabilistic approaches driven mainly by a *multi-task* formulation utilizing the information transfer among users, to learn a *complete* preference model accounting for *all* preference relations. For the purpose of optimizing parameters, it is not efficient to learn a complete model over a high-dimensional parameter space, because the model is only required to be accurate around possible optimal parameters.

In audio reproduction systems—like home entertainment and professional mixing equipment—preference learning approaches are relatively unknown, despite the clear evidence that personalization may be beneficial in for example equalization (Paterson [11] and Zhang *et. al.* [18]). Existing approaches such as Reed [13], Pardo *et. al.* [10], and Sabin *et. al.* [14], are based on non-probabilistic approaches, thus neglecting the highly stochastic nature of perceptual responses.

In this work we focus on audio reproduction systems and the outlined task of optimizing multiple parameters in such systems for an individual user listening to the output of the system. For this purpose, we propose and consider a

combination of robust Bayesian modeling, an engaging user interface for user feedback and global optimization techniques (active learning) in an interactive loop visualized in Fig. 1. The loop constitutes a general framework where the inherent uncertainty in user feedback is addressed from a Bayesian viewpoint in which the belief in the user's (unknown) objective function is modeled with (warped) Gaussian process (GP) regression [15]. The framework uses an intuitive and simple graphical user interface for obtaining user ratings, which allows the user to listen to previously rated settings thus serving as anchors/references for future ratings. In contrast to standard practice, we do however not only allow the user to listen to previous settings, but we also allow the user to change all the ratings of previous settings, if for some reason a new setting would change e.g. the span of the scale. This is possible, since we are constantly updating our regression model to reflect the belief about the user's objective function given the ratings obtained so far. Finally, we propose to use a sequential optimization technique to rapidly find a (possibly local) optimum of the user's objective function. The sequential design takes advantage of the Bayesian formulation by including the belief about the user's objective function. This significantly reduces the required number of settings that the user should rate in order to find an optimum.

We furthermore consider the fact that certain parameters may be correlated with respect to the user's objective. An example could be the compression ratio, the attack time and the release time in a compressor. To exploit such correlation and obtain better modeling capabilities, we suggest a specific model which assumes correlation between specific input parameters.

To demonstrate the potential of the framework for personal audio system optimization, we use a five-band constant-Q equalizer (EQ) as the running example, because the parameters (gains) in an EQ is something that we (as professionals) more or less all can relate to. We are aware that any audio-engineering professional will probably be able to quickly tune the five parameters of the EQ to his own objective. However, this is actually not a very typical scenario. Typically, users of home entertainment systems are untrained, and thus, have very little intuition about the parameters that they have the opportunity to tune and close to no intuition at all about the interplay *between* parameters. Hence with for instance five parameters controlling a virtual surround sound system with virtual base enhancement, most users would seek an optimal setting using trial and error (random experimentation). This is the premises in which the EQ example should be considered and the EQ is just convenient for demonstration purposes.

Through model comparison, we first show that the model with assumed correlation between input parameters improves the modeling capabilities compared to a traditional GP model without assumed correlation. The analysis is performed on real-world data, where 21 test subjects have rated different randomly chosen settings of the EQ. Even for this EQ with relative few bands—which is thus perceptually well separated—we would expect the gains in adjacent bands to be somewhat correlated with regards to the user's objective. Secondly, we evaluate the usefulness of the entire framework in a real-world experiment where personalization of the EQ have been conducted for twelve test subjects. As the EQ has over fifty-nine thousands unique settings, the hypothesis is that the preferred setting will be hard to find (for the typically untrained user) without an efficient sequential design approach and correspondingly good modeling

capabilities. The results from the real-world listening experiments focusing on the statistical difference between random experimentation and sequential experimental design, show a clear advantage of the sequential design approach.

Our contribution is thus three fold: First in Sec. 2, we propose a general personalization framework with an intuitive user interface (Sec. 2.3), a principled modeling approach using warped Gaussian processes extended to expect correlation between adjacent input parameters (Sec. 2.1) and a sequential design approach (Sec. 2.2). Secondly in Sec. 3.2, we show that the GP model extension provides better modeling capabilities for our specific purpose. Thirdly, we evaluate the entire framework by a listening experiment in a real-world interactive scenario and outline the results in Sec. 3.3. A discussion is provided in Sec. 4 and the paper is concluded in Sec. 5.

2 Personalization Framework

The proposed personalization framework uses an interactive loop to discover the user's preferred setting of a particular audio system, where we as an example use the EQ. The interactive loop is visualized in Fig. 1. The loop can conceptually be divided into three parts: a preference modeling part, a sequential design part and an interface part. The preference modeling part presents how a user's objective function over EQ settings is learned based on user ratings. The sequential design part covers how to choose new EQ settings to be rated based on what the model currently predicts. Finally, the interface part covers the design of the graphical user interface, such that it is both intuitive and easy to use for the users. The three parts are described in the following three sections.

2.1 Preference Modeling

We represent each system setting as a $d = 5$ dimensional vector of parameters, $\mathbf{x} = [x_1, ..., x_d]^\top$. Next, we assumed that the user's objective is an unobserved real-valued stochastic function (or process), such that each unique setting $\mathbf{x}_i$ has a corresponding real-valued function value, $f(\mathbf{x}_i)$, expressing the user's preference for the particular setting. This function is to be learned—and subsequently maximized—trough a number of experiments where we observe the user's expressed preference by a rating on a bounded scale, $y \in]0; 1[$, where 0 is *Bad* and 1 is *Good* (see interface (2) on Fig. 1). At some point the user has evaluated n such distinct system settings $\mathbf{x}_i \in \mathbf{X}$ collected in $\mathbf{X} = \{\mathbf{x}_i | i = 1, ..., n\}$, with a related set of n responses denoted $\mathbf{Y} = \{y_i | i = 1, ..., n\}$.

We model the function mapping from settings, $\mathbf{x}_i$, to ratings, y_i, by a so-called *warped* Gaussian process [15]. A standard Gaussian process (GP) is a stochastic process defined as a collection of random variables, any finite subset of which must have a joint Gaussian distribution [12]. In effect, the GP is placed as a prior over any finite set of functional values $\mathbf{f} = [f_1, f_2, ..., f_n]^\top$, where $f_i = f(\mathbf{x}_i)$, resulting in a finite multivariate Gaussian distribution over the set as $\mathbf{f}|\mathbf{X} \sim \mathcal{N}(\mathbf{0}, \mathbf{K_{XX}})$, where each element of the covariance matrix $\mathbf{K_{XX}}$ is given by a covariance function $k(\cdot, \cdot)$ such that $[\mathbf{K_{XX}}]_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j)$. The GP prior can be used in non-parametric Bayesian regression frameworks where the likelihood function can be parameterized by a smooth and continuous function $f(\cdot)$.

However, our regression setup is special due to the bounded nature of the ratings. We therefore use a warped Gaussian process in which the original ratings in $\mathbf{Y}$ are transformed into a form where the data is modeled by a traditional Gaussian noise model [12, Chapter 2]. Several warping functions would apply, but a natural choice is the inverse cumulative Gaussian (probit) $\Phi^{-1}(\cdot)$—with zero mean and unity variance—such that observations are warped as $z_i = \Phi^{-1}(y_i)$.

The final model is defined by,

$$\begin{aligned}
\sigma_s|\theta_s &\sim \mathcal{U}(0,\infty)\\
\sigma_\ell|\theta_\ell &\sim \mathcal{U}(0,\infty)\\
\sigma|\theta_\ell &\sim \mathcal{U}(0,\infty)\\
f_i|\sigma_s,\sigma_\ell &\sim \mathcal{GP}\left(m\left(\mathbf{x}_i\right), \mathrm{k}\left(\mathbf{x}_i,\cdot\right)_{\sigma_s,\sigma_\ell}\right)\\
z_i|f_i &\sim \mathcal{N}\left(f_i,\sigma\right) \qquad (1)\\
z_i &= \Phi^{-1}\left(y_i\right), \qquad (2)
\end{aligned}$$

where σ_ℓ is the length scale of the covariance function, σ_s is the standard deviation of the latent function, and σ is the noise standard deviation (in latent space). $\mathcal{U}(a,b)$ denotes a uniform hyper prior on the open interval from a to b, i.e. an improper and *non-informative* prior. Alternatively, so-called *weakly-informative* hyper priors would apply—especially over the length scale σ_ℓ—such as the half-student-t hyper prior [5, 16], which could be applied to provide a more robust inference and prediction scheme avoiding the GP model to fit hyperplanes with only few observations. We note that the observation noise, σ, can be included in the covariance function.

Given this model, the main question remains regarding the covariance (or kernel) function, which effectively defines the smoothness of the function. We consider two covariance functions based on the general form of the squared exponential kernel [12]

$$k\left(\mathbf{x}_i,\mathbf{x}_j\right) = \sigma_s \exp\left(-\frac{1}{\sigma_\ell}\left(\mathbf{x}_i-\mathbf{x}_j\right)^\top \mathbf{\Lambda}^{-1}\left(\mathbf{x}_i-\mathbf{x}_j\right)\right). \qquad (3)$$

In the first case, $\mathbf{\Lambda}$ is the identity matrix leading to the well-known (isotropic) squared exponential covariance function $k_{\mathrm{iso}}\left(\mathbf{x}_i,\mathbf{x}_j\right) = \sigma_s \exp\left(-\frac{1}{\sigma_\ell}\left\|\mathbf{x}_i-\mathbf{x}_j\right\|^2\right)$. In the second case, $\mathbf{\Lambda}$ is a general positive semi-definite matrix defining a correlation between parameters (input space) as explicit prior information. Here will denote this variant as the Mahalanobis covariance function[1], $k_{\mathrm{mah}}\left(\mathbf{x}_i,\mathbf{x}_j\right)$ and set

$$\mathbf{\Lambda}_{\mathrm{mah}} = \begin{bmatrix} 1 & 0.5 & 0.2 & 0 & 0\\ 0.5 & 1 & 0.5 & 0.2 & 0\\ 0.2 & 0.5 & 1 & 0.5 & 0.2\\ 0 & 0.2 & 0.5 & 1 & 0.5\\ 0 & 0 & 0.2 & 0.5 & 1 \end{bmatrix}. \qquad (4)$$

The effect of the two options on the EQ example will be evaluated with reference to the standard case as `iso` and the Mahalanobis case as `mah`.

[1] Sometimes also referred to as a anisotropic (squared exponential) covariance functions [12].

We turn to a standard GP inference scheme [12] in which the covariance and likelihood parameters, $\sigma_s, \sigma_\ell, \sigma$, are approximated by point estimates by maximizing the marginal likelihood (or evidence) using a BFGS method and where the posterior $p(\mathbf{f}|\mathbf{Y}, \mathbf{X})$ is analytical tractable [15]. For the BFGS methods, the parameters are always initialized as $\sigma_s = 1, \sigma_\ell = 1, \sigma = 1$. The predictive mean and (co)variance of the latent function, $\mathbb{E}(\mathbf{f}^*)$ and $\mathbb{V}(\mathbf{f}^*)$, are given in standard form [12] as

$$\mathbb{E}\{\mathbf{f}^*\} = \mathbf{K}_{\mathbf{X}\mathbf{X}^*}^\top \left[\mathbf{K}_{\mathbf{X}\mathbf{X}} + \sigma_i^2 \mathbf{I}\right]^{-1} \Phi^{-1}(\mathbf{Y}) \tag{5}$$

$$\mathbb{V}\{\mathbf{f}^*\} = \mathbf{K}_{\mathbf{X}^*\mathbf{X}^*} - \mathbf{K}_{\mathbf{X}\mathbf{X}^*}^\top \left[\mathbf{K}_{\mathbf{X}\mathbf{X}} + \sigma_i^2 \mathbf{I}\right]^{-1} \mathbf{K}_{\mathbf{X}\mathbf{X}^*} \tag{6}$$

where $\mathbf{K}_{\mathbf{A}\mathbf{B}}$ is the kernel matrix containing either evaluations between training inputs, $\mathbf{A} = \mathbf{B} = \mathbf{X}$, test inputs, $\mathbf{A} = \mathbf{B} = \mathbf{X}^*$, or between training and test inputs, $\mathbf{A} = \mathbf{X}, \mathbf{B} = \mathbf{X}^*$.

The predictive distribution and in particular the predictive uncertainty is a clear advantage of the probabilistic GP framework, since the predictive mean and predictive (co)variance can be used to determine the information gain in including a new candidate point into the model as considered in the next section.

2.2 Sequential Experimental Design

Classical experimental designs such as Latin Squares or random experimentation [9] become increasingly infeasible in high dimensions. As an alternative, we propose to use sequential design approaches which, by greedy selection of the most informative next sample, potentially achieve much faster convergence than fixed designs [7].

The main purpose is to define a selection criterion which finds the optimal of the (unknown) objective function. The applied criterion is a slightly modified version of the so-called *Expected Improvement* (EI) [7], a known criterion in the design of computer experiment (DACE) community. The expected improvement is for each candidate point, $\mathbf{x}_j$, defined as,

$$\mathrm{EI}(\mathbf{x}_j) = \sigma_{EI} \cdot \mathcal{N}\left(\frac{\mu_{EI}}{\sigma_{EI}}\right) + \mu_{EI} \cdot \Phi\left(\frac{\mu_{EI}}{\sigma_{EI}}\right), \tag{7}$$

where $\mathcal{N}(\cdot)$ is the standard Normal distribution and $\Phi(\cdot)$ is the standard cumulative Gaussian as before. Given the predictive distribution the EI is given by,

$$\mu_{\mathrm{EI}} = \mu_j - \mu_{\max}$$
$$\sigma_{\mathrm{EI}}^2 = \sigma_j^2 + \sigma_{\max}^2 - 2\sigma_{j,\max}$$

where μ_j and σ_j is the predictive mean and variance of the test point and $\mu_{\max}$ and σ_{max} is the predictive mean and variance of the current maximum of the objective function (using the predictive mean as the predictor), i.e., the current best setting, all of which originate from Eq. 5-6. The covariance between the two function values, $\sigma_{j,\max}$, requires correlated predictions which we refrain from due to computation burden, thus $\sigma_{j,\max} = 0, \forall \mathbf{x}_j$. Hence, the selection of a new point to evaluate is given by

$$\mathbf{x}_{new} = \arg\max_{\mathbf{x}_j} \mathrm{EI}(\mathbf{x}_j)$$

which is then included in the current set of training points and evaluated by the user through the user interface. We refer to this as the `active` configuration, where the very first setting for the user to evaluate is chosen randomly. A random configuration `rnd` is included in which samples are selected randomly to provide a baseline method.

The interactive framework leaves four strategies to be investigated experimentally: `rnd-iso`, `rnd-mah`, `active-iso` and `active-mah`.

2.3 Interface

When applying absolute ratings, it is important to define anchor and/or reference points [2]. This allows users to compare stimuli with a fixed reference, such that each rating is *calibrated* both with respect to previous ratings, but also with respect to yet unobserved stimuli, which might redefine the end points of the rating scale. To address these two issues a graphical user interface similar to [10] is designed. Users can listen to previous settings (references) and are allowed to change previous ratings based on the new one. Obviously, this means that ratings are neither directly comparable across users nor between iterations. However, it is not of particular interest to use ratings across users to formulate one single optimal setting, but instead we are interested in personalized settings—one for each user.

3 Experiment

To evaluate the different model configurations and experimental designs in a real-world scenario, an experiment was conducted, in which the five gains of the EQ are to be optimized by the four different versions of the proposed framework. The procedure and results are described in the following section.

3.1 Procedure

The experiment consisted of three parts: (1), (2) and (3) as visualized in Fig. 2. During part (1), the subjects rate ten randomly chosen balls to learn how to use the interface and to get an impression of the stimuli (EQ processed music). Part (2) consisted of three sessions for which the order of sessions was balanced across test subjects. In each of the three sessions a particular model (`iso` or `mah`) and sequential design (`rnd` or `active`) are used to find a personalized setting of the EQ for the test subject. Finally in part (3), the preferred settings, found by each of the four combinations of models and sequential designs after 10, 15, 20, 25 and 30 presented settings, are determined by which model predicted the setting that is rated highest (in the tournament - see Fig. 2). Each tournament (as defined in Fig. 2) was repeated twice resulting in ten tournaments for which the sequence was randomized.

The sound was played back to the test subjects through Sennheiser HD650 headphones and a FirestoneAudio FUBAR DACIII headphone amplifier at constant level. The output level was furthermore loudness normalized to the same level using a A-weighting filter, with the purpose to make the rating process easier for the test subjects, such that the test subject primarily focus on the

tonal qualities—not the loudness. An interval of 31.9 seconds in the beginning of the track "Sleeping with the Light on" by Teitur was used as the music piece.

3.2 Model Analysis

The interactive loop outlined in Sec. 2 has two critical blocks which influence the convergence of the optimization procedure: 1) the GP-model predictions of the subject's objective function at all inputs given only the rated inputs, and 2) the sequential design approach. In this section we only seek to determine which GP model that best suits our purpose without the influence of the sequential design approach. We do this by evaluating the two GP models—`iso` and `mah`—in terms of their predictive performance on random data sets for 21 test subjects. In machine learning and statistics, cross-validation is typically used to get an unbiased measure of the predictive performance. Since the random data sets for each test subject contain only 30 ratings, we use leave-one-out cross validation (LOO-CV) [12] to get an effectively unbiased measure of the true predictive performance.

Performance is typically defined as an error measure through a cost function, such as the sum-of-squared error function. However, such error functions only include the absolute deterministic errors made by the model on noisy data without additionally considering if the model actually fits the noise correctly.

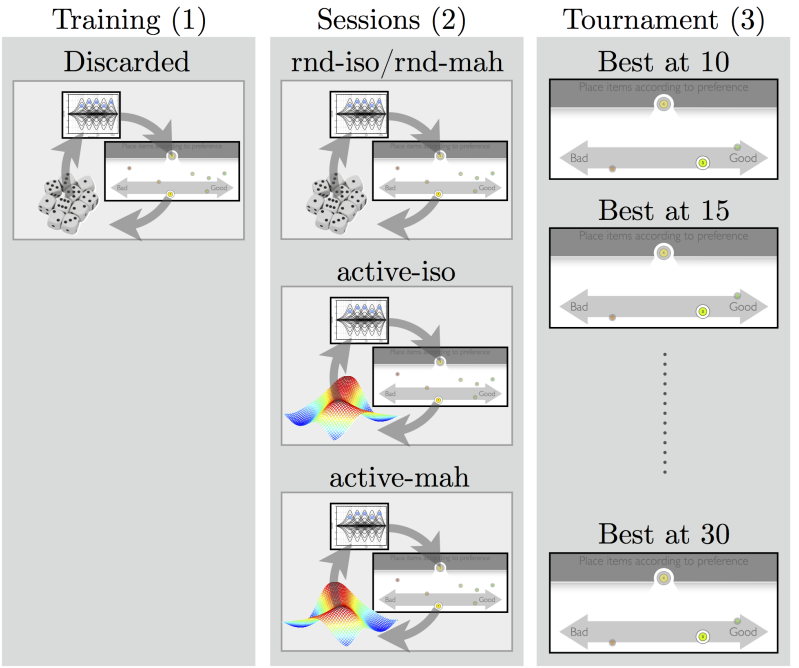


Figure 2: Visualization of the experiment with its 3 sessions: (1) Training, (2) Sessions and (3) Tournament.

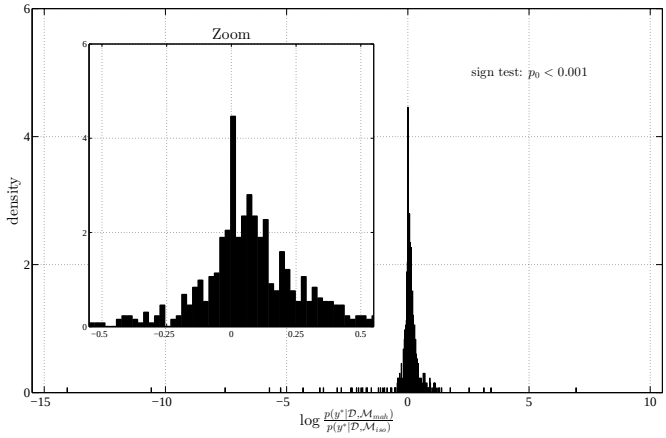


Figure 3: Predictive log-likelihood ratio (Bayes factor) over all leave-one-out cross-validation splits for all twenty-one test subjects. The p_0-value gives the probability of the null-hypothesis that the median is equal to zero (the to models are equally well) with the alternative hypothesis that the median is larger than zero (the Mahalanobis model is better than the isotropic) using an non-parametric sign test.

For the sequential design approach to work efficiently, the model should both fit the data and account for the noise in the data as well as possible. To capture this in the performance measure, typically, the predictive likelihood $p(y^*|\mathcal{D}, \mathcal{M})$ of the unseen data points y^* given the model $\mathcal{M}$ and the observed data $\mathcal{D}$ is used.

A proper Bayesian and statistical way of comparing two models [17, 6] is to compare the (predictive) likelihood ratio $p(y^*|\mathcal{D}, \mathcal{M}_{mah})/p(y^*|\mathcal{D}, \mathcal{M}_{iso})$ between the two different models—`mah` and `iso`. This is also referred to as the *Bayes factor* [6]. A (log) Bayes factor larger than zero favors the model denoted in the nominator, whereas a (log) Bayes factor less than zero favors the model in the denominator.

For each of the 21 random data sets—one for each test subject— LOO-CV is used and the (log) Bayes factor is calculated for each LOO-CV split. This gives a total of 21×30 Bayes factor estimates shown in a histogram in Fig. 3. We see that on average, the `mah` model performs the best probabilistic predictions of test subjects's individual ratings and thus appears to be the most suitable model due to the assumed correlation between adjacent parameters. A non-parametric sign test shows that this is significant (sample size of 630).

3.3 Sequential Design Analysis

The results are summarized in Fig. 4(a). The illustrated p_0-values gives the significance level for which the null-hypothesis, that the total number of active wins is equal to the total number of random wins at each tournament point (#examples), can be accepted.

Averaged across test subjects and repetitions, the sequential design is significantly better than random design after any given number of examples, as shown by the p_0–values. This is without distinguishing between the two applied covariance functions. It demonstrates the potential of the Bayesian model and active learning methods in audio applications. It is furthermore noted that a standard fixed design will approximate the random configuration in this high-dimensional space.

The second aspect is if the more informative *Mahalanobis* (`mah`) prior results in a more accurate model with only a few ratings available. This is generally not the case, although the specific Mahalanobis model possesses better generalization capabilities compared to the isotropic model as shown in Sec. 3.2.

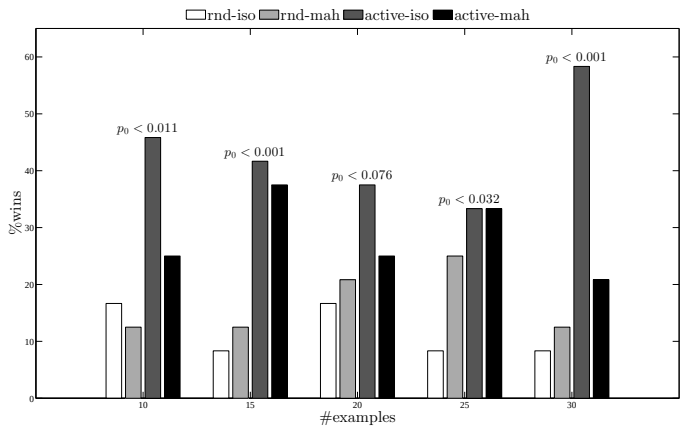
4 Discussion and Future Work

The results presented in this paper has focused first on verifying that the proposed Mahanolobis model is suitable in this context, and secondly, demonstrating that the sequential design approach actually performs as expected (and better than random). There are however many possibilities for further evaluation and development.

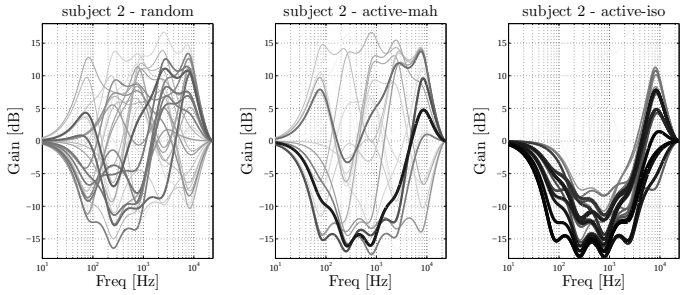
In regards to the specific prior, we believe despite the lack of evidence in the present paper, that the Mahalanobis covariance function will be found suitable in several audio applications—including the EQ example used here. We speculate that at least two additions would improve the performance of the Mahalanobis model in the suggested framework. Firstly, the modeling capabilities could be improved by a parametrization of the correlation structure in the Mahalanobis kernel by a small set of parameters, which could then be inferred from data. The latter is easily accomplished in the GP framework by evidence maximization. Secondly, the sequential design criterion (Sec. 2.2) does not in its current form fully exploit the correlation between predictive function values for different settings. To include this correlation the covariance matrix between all unique settings must be calculated. Calculating these is currently computational infeasible. To overcome this and exploit the modeled correlation in the sequential design criterion, a greedy-gradient approach is currently being developed and tested with regards to find a (possible local) optimum of the (correlated) EI.

The Gaussian process modeling approach will in general benefit from the addition of weakly informative hyper-priors over especially the length scale parameter, σ_ℓ. This will result in a more robust inference scheme in which unrealistic hyper-plane predictions of the user's objective function would be avoided, thus aiding the sequential design. This is currently being introduced into the modeling framework.

The current evaluation is based on an absolute paradigm with adjustable anchors in terms of previous ratings. For the user, it can however be quite demanding to keep track of all ratings, when there are several items (*balls*) present, which leads to inconsistent ratings. The GP based personalization framework is easily extendable with other paradigms such as pairwise comparisons or more general ranking based approaches. It is speculated that a more robust paradigm (with respect to user feedback) may further aid the optimization process.



(a) Learning Curve



(b) Ratings

Figure 4: (a): The percentage of times the predicted preferred setting by each of the four models wins over the other models across test subjects at each of the five tournament points. The p_0-values is for accepting the null-hypothesis that the two active sequential design approaches is equal to the two random approaches using a binomial test. (b): Actual ratings of different EQ settings from the three Sessions for test subject 2. The EQ curves are the imposed gain and the color and thickness of the EQ curves both indicate the rating, where think/dark black is a good ratings ($y \to 1$) and thin/light gray is a bad ratings ($y \to 0$).

Finally, it is the ambition to evaluate the proposed framework on a larger population, which could be accomplished by embedding the current personalization framework in a web application allowing evaluation on a larger scale.

5 Conclusion

We have proposed a framework for obtaining personalized systems—in particular audio systems—which utilizes a Bayesian probabilistic modeling approach in combination with sequential experimental design. This improves the high-dimensional preference optimization procedure in comparison to random (equivalent to manual) experimentation. The solutions found by the sequential approach is significantly preferred by the test subjects over the solutions found by random experimentation. The results do not support any advantage of using the more informative Gaussian process prior with the Mahalanobis kernel compared to the less informative Gaussian process prior with the isotropic kernel. Supported by the demonstrated modeling capabilities of the Mahalanobis kernel, it is nevertheless believed that future additions to the framework would be able to exploit these possibilities and hence improve the performance of the framework.

Acknowledgment

This work was supported in part by the Danish Council for Strategic Research of the Danish Agency for Science Technology and Innovation under the CoSound project, case number 11-115328. This publication only reflects the authors' views.

References

[1] D. Baskent, C. L Eiler, and B. Edwards. Using genetic algorithms with subjective input from human subjects: Implications for fitting hearing aids and cochlear implants. *Ear and Hearing*, 28(3):370–380, 2007.

[2] S. Bech and N. Zacharov. *Perceptual Audio Evaluation - Theory, Method and Application*. Wiley, July 2006.

[3] A. Birlutiu, P. Groot, and T. Heskes. Multi-task preference learning with an application to hearing aid personalization. *Neurocomputing*, 73(9-9):1177, 2010.

[4] A. Birlutiu, P. Groot, and T. Heskes. Efficiently learning the preferences of people. *Machine Learning*, (July 2010), May 2012.

[5] A. Gelman. Prior distributions for variance parameters in hierarchical models. *Bayesian Analysis*, 1(3):515–533, 2006.

[6] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian data analysis*. Chapman and Hall/CRC, 2004.

[7] D. R Jones. A Taxonomy of Global Optimization Methods Based on Response Surfaces. *Journal of Global Optimization*, 21(4):345–383, 2001.

[8] F.K. Kuk and N.M.C. Pape. The reliability of a modified simplex procedure in hearing-aid frequency-response selection. *Journal of Speech and Hearing Research*, 35(2):418–429, 1992.

[9] D.C. Montgomery. *Design and analysis of experiments*. Wiley, 2009.

[10] B. Pardo, D. Little, and D. Gergle. Building a personalized audio equalizer interface with transfer learning and active learning. *Proceedings of the second international ACM workshop on Music information retrieval with user-centered and multimodal strategies - MIRUM '12*, page 13, 2012.

[11] J. Paterson. The Preset Is Dead; Long Live the Preset. In *Audio Engineering Society Convention 130*, 2011.

[12] C.E. Rasmussen and C.K.I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[13] D. Reed. Capturing perceptual expertise: a sound equalization expert system. *Knowledge-Based Systems*, 14(1-2):111–118, March 2001.

[14] A. Sabin and B. Pardo. Rapid learning of subjective preference in equalization. In *Audio Engineering Society Convention 125*, 2008.

[15] E. Snelson, C. E. Rasmussen, and Z. Ghahramani. Warped gaussian processes. *Advances in Neural Information Processing Systems (NIPS)*, (16):337–344, 2004.

[16] J. Vanhatalo and A. Vehtari. Sparse log Gaussian processes via MCMC for spatial epidemiology. In *JMLR Workshop and Conference Proceedings*, volume 1, pages 73–89, 2007.

[17] A. Vehtari and J. Ojanen. A survey of bayesian predictive methods for model assessment, selection and comparison. *Statistics Surveys*, 6:142, 2012.

[18] D. Zhang, H. Xia, T. Chua, and G.A. Maguire. Impact of personalized equalization curves on music quality in dichotic listening. *Digital Audio Effects - DAFx '12*, pages 1–7, 2012.

Appendix G

Hearing Aid Personalization

Jens Brehm Nielsen, Bjørn Sand Jensen, Jakob Nielsen, Jan Larsen. Hearing Aid Personalization. Published in *NIPS 2013 Workshop on Personalization*, 2013.

Errata

- Eq. (7) should be

$$\boldsymbol{\mu}_* = \mathbf{k}_*^\top \mathbf{K}^{-1} \hat{\mathbf{f}}$$

Hearing Aid Personalization

Jens Brehm Nielsen[*], **Jakob Nielsen**
Widex A/S
Nymøllevej 6
3560 Lynge, Denmark
{jeb,jnl}@widex.com

Bjørn Sand Jensen, Jan Larsen
DTU Compute
Matematiktorvet 303B
2800 Kgs. Lyngby, Denmark
{bjje,janla}@dtu.dk

Abstract

Modern digital hearing aids require and offer a great level of personalization. Today, this personalization is not performed based directly on what the user actually perceives, but on a hearing-care professional's interpretation of what the user explains *about what is perceived*. In this paper, an interactive personalization system based on Gaussian process regression and active learning is proposed, which personalize the hearing aids based directly on what the user perceives. Preliminary results demonstrate a significant difference between a truly personalized setting obtained with the proposed system and a setting obtained by the current practice.

1 Introduction

Hearing aids (HAs) [1] are fitted by predetermined rules (prescriptions [1, Chapter 10]) given frequency-dependent hearing thresholds—called an audiogram—of the hearing-impaired user. These rules are based on years of practical experience and research of the human auditory system, however nobody knows exactly how the fitted HAs sound like, except of course, the user. From empirical studies, it is well-known [1, Chapter 12], that users with the same audiogram may benefit from—and prefer—very different HA settings. Therefore, a hearing-care professional with years of experience often needs to manually fine-tune the HAs beyond the predetermined prescription. This fine tuning is typically based on oral feedback from the user [1, Chapter 12]. In effect, this feedback is the user's oral translation of the perception using a *description* meaningful to the subject. This description, however, might not necessarily give meaning to the hearing-care professional. It is believed that HA users would benefit greatly if the HAs were adjusted and personalized based directly on how the devices sounds—and not on a poorly aligned translation thereof. In this paper, a machine-learning based personalization system is proposed, which adjusts hearing aid settings based on user feedback, which mimics what the individual actually hears. From the user's perspective, the feedback is returned as a *degree-of-preference* rating between two different hearing aid settings. This is an intuitive way of expressing what is perceived while inducing a low cognitive load compared to conveying an oral response to a single setting. The feedback is used to learn a Gaussian process regression model of the user's latent objective function—the optimum of which corresponds to the truly personalized setting. To quickly find this optimum, the GP model is repeatedly updated based on the feedback from the user and subsequently used to select the next comparison to present to the user using active learning. Fast convergence is an absolute requirement, because even quarters of an hour of careful listening is a very demanding task—especially for most hearing aid users.

2 Personalization System

The personalization system is an *interactive loop* visualized in Fig. 1. The loop essentially contains three parts: I) A modeling part where the user's objective function is modeled by a Gaussian process

[*]DTU Compute, Matematiktorvet 303B, 2800 Kgs. Lyngby, Denmark jenb@dtu.dk

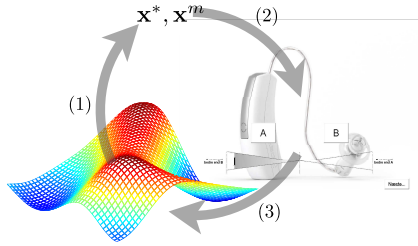


Figure 1: (1): a new optimal setting is determined based on the current (probabilistic) estimate of the subject's objective function. (2): the optimal setting is compared to the setting which maximizes the current estimate of the subject's objective function, and the subject assesses the *degree of preference* between the two settings. (3): the estimate of the subject's objective function is updated based on the recent assessment.

based on the feedback obtained, II) an active learning part setting op the next comparison based on the current state of the model, and III) an user interface part.

2.1 Part I: Modeling the User's Objective from Feedback

The modeling of the subject's objective function is performed in a Bayesian non-parametric setup based on Gaussian Processes (GPs) [2]. In the following, GP regression from *degree of preference* observations will be explained. The GP framework is based on previous work found in [3].

2.1.1 Gaussian Process Prior

A Gaussian process (GP) defines a prior, $f(\mathbf{x}) \sim \mathcal{GP}\left(0, k(\mathbf{x}, \cdot)_{\boldsymbol{\theta}_\mathcal{C}}\right)$, over functions, $f : \mathbb{R}^D \rightarrow \mathbb{R}, \mathbf{x} \mapsto f(\mathbf{x})$, where $k(\cdot, \cdot)_{\boldsymbol{\theta}_\mathcal{C}}$ is a covariance function or kernel with parameters $\boldsymbol{\theta}_\mathcal{C}$. In this paper, a squared exponential (SE) kernel with individual length scales λ_d for each input dimension (ARD) will be used, hence $\boldsymbol{\theta}_\mathcal{C} = \{\sigma_f, \lambda_1, ..., \lambda_D\}$. Given a finite set of function values (random variables), $\mathbf{f} = [f(\mathbf{x}_1), ..., f(\mathbf{x}_n)]^\top$ for $\mathcal{X} = \{\mathbf{x}_i \in \mathbb{R}^D | i = 1, ..., n\}$, the GP defines a joint distribution over the function values as $p(\mathbf{f}|\mathcal{X}, \boldsymbol{\theta}_\mathcal{C}) = \mathcal{N}(\mathbf{0}, \mathbf{K})$, where $[\mathbf{K}]_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j)_{\boldsymbol{\theta}_\mathcal{C}}$. By specifying the likelihood $p(\mathcal{Y}|\mathbf{f}, \boldsymbol{\theta}_\mathcal{L})$ of some set of observations $\mathcal{Y}$ given the finite set of function values $\mathbf{f}$ the posterior distribution over the function values $\mathbf{f}$ is given by Bayes formula

$$p(\mathbf{f}|\mathcal{Y}, \mathcal{X}, \boldsymbol{\theta}) = \frac{p(\mathcal{Y}|\mathbf{f}, \boldsymbol{\theta}_\mathcal{L}) p(\mathbf{f}|\mathcal{X}, \boldsymbol{\theta}_\mathcal{C})}{p(\mathcal{Y}|\mathcal{X}, \boldsymbol{\theta})} = \frac{p(\mathcal{Y}|\mathbf{f}, \boldsymbol{\theta}_\mathcal{L}) p(\mathbf{f}|\mathcal{X}, \boldsymbol{\theta}_\mathcal{C})}{\int p(\mathcal{Y}|\mathbf{f}, \boldsymbol{\theta}_\mathcal{L}) p(\mathbf{f}|\mathcal{X}, \boldsymbol{\theta}_\mathcal{C}) d\mathbf{f}}, \tag{1}$$

where the hyper-parameters $\boldsymbol{\theta} = \{\boldsymbol{\theta}_\mathcal{L}, \boldsymbol{\theta}_\mathcal{C}\}$ contain both likelihood and covariance parameters.

2.1.2 Beta likelihood

Following previous work [3], GP regression from pairwise continuous observations (*degree of preference*) is performed with a likelihood function based on a re-parameterized beta distribution. Consider a set of pairwise observations $\mathcal{Y} = \{y_k \in (0, 1) | k = 1, ..., m\}$ of the *degree of preference* between two distinct inputs $u_k, v_k \in \{1, ..., n\}$, implying that $\mathbf{x}_{u_k}, \mathbf{x}_{v_k} \in \mathcal{X}$. With this formulation, an dominant preference for the first option u_k is reflected by $y_k \rightarrow 0$, whereas an dominant preference for the second option v_k is reflected by $y_k \rightarrow 1$ reflects. No preference is indicated by $y_k = 0.5$. A suitable likelihood function $p(y_k|\mathbf{f}_k)$ can now be constructed given the function values for the two input instances $\mathbf{f}_k = [f(\mathbf{x}_{u_k}), f(\mathbf{x}_{v_k})]^\top$, by re-parameterizing the beta distribution, $\text{Beta}\left(\,\cdot\,; \alpha, \beta\right)$, as $p(y_k|\mathbf{f}_k, \boldsymbol{\theta}_\mathcal{L}) = \text{Beta}\left(y_k; \nu\zeta(\mathbf{f}_k, \sigma), \nu(1 - \zeta(\mathbf{f}_k, \sigma))\right)$,
where $\boldsymbol{\theta}_\mathcal{L} = \{\nu, \sigma\}$ is the set of likelihood parameters, ν is a dispersion parameter around the mean $\zeta(\mathbf{f}_k, \sigma)$, which is defined by

$$\zeta(\mathbf{f}_k, \sigma) = \Phi\left(\frac{f(\mathbf{x}_{v_k}) - f(\mathbf{x}_{u_k})}{\sqrt{2}\sigma}\right), \tag{2}$$

where $\Phi(\cdot)$ is the standard normal cumulative density function—with zero mean and unit variance—and σ is a slope parameter. By assuming that observations are independent given the latent function values $\mathbf{f}$, the likelihood can be written as $p(\mathcal{Y}|\mathbf{f}, \boldsymbol{\theta}_\mathcal{L}) = \prod_{k=1}^m p(y_k|\mathbf{f}_k, \boldsymbol{\theta}_\mathcal{L})$, which is plugged into Eq. 1 together with the GP prior from Eq. 2.1.1 to complete the Bayesian model.

2.1.3 Inference and Prediction

The Gaussian process model outlined above is not analytical tractable due to the Beta-like likelihood function from Eq. 2.1.2. Instead, approximate inference based on the Laplace approximation [2, Section 3.4] is performed as in [3], giving

$$p(\mathbf{f}|\mathcal{Y},\mathcal{X},\boldsymbol{\theta}) \approx q(\mathbf{f}|\mathcal{Y},\mathcal{X},\boldsymbol{\theta}) = \mathcal{N}\left(\hat{\mathbf{f}}, \left(\mathbf{W}+\mathbf{K}^{-1}\right)^{-1}\right) \tag{3}$$

where $\hat{\mathbf{f}}$ is the maximum of the posterior (mode) and $[\mathbf{W}]_{i,j} = -\sum_{k=1}^{m} \frac{\partial^2 \log p(y_k|\mathbf{f}_k\boldsymbol{\theta}_{\mathcal{L}})}{\partial f(\mathbf{x}_i)\partial f(\mathbf{x}_j)}$. Note, that unlike traditional classification and regression problems, $\mathbf{W}$ does not become diagonal due to the pairwise structure. For further details, see [4].

For (hyper) parameter optimization, traditional ML-II optimization [2, Chapter 5.2] results in large length scales with few observations (< 20) available. This is an undesirable property in combination with active learning. Therefore, a half-student's-t prior is placed on critical hyper-parameters, resulting in the evidence $q(\mathcal{Y}|\mathcal{X},\boldsymbol{\theta})$ of the Laplace approximation being augmented with a extra term (see [5] for similar use). The resulting MAP-II scheme for hyper parameter optimization is therefore:

$$\boldsymbol{\theta}^{\text{MAP-II}} = \arg\max_{\boldsymbol{\theta}} \left\{\log q(\mathcal{Y}|\mathcal{X},\boldsymbol{\theta}) + \log p(\boldsymbol{\theta})\right\}, \tag{4}$$

where $\sigma_f \sim \delta(\sigma_f = 4)$, $\lambda_d \sim$ half-St$(\cdot|6,100)$ and $\sigma, (\nu-2) \sim$ half-St$(\cdot|6,10)$ with

$$\text{half-St}(z;\xi,s) \propto \left(1 + \frac{1}{\xi}\left(\frac{z}{s}\right)^2\right)^{-(\xi+1)/2}. \tag{5}$$

The predictive distribution $p(\mathbf{f}_*|\mathcal{Y},\mathcal{X},\mathcal{X}_*,\boldsymbol{\theta})$ of the function values $\mathbf{f}_* = [f(\mathbf{x}_1^*),...,f(\mathbf{x}_o^*)]^\top$ at new input locations $\mathcal{X}_* = \{\mathbf{x}_l^* \in \mathbb{R}^D | l = 1,...,o\}$ is given by

$$p(\mathbf{f}_*|\mathcal{Y},\mathcal{X},\mathcal{X}_*,\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*), \tag{6}$$

$$\boldsymbol{\mu}_* = \mathbf{k}_*^\top \left(\mathbf{W}+\mathbf{K}^{-1}\right)\hat{\mathbf{f}} \tag{7}$$

$$\boldsymbol{\Sigma}_* = \mathbf{K}_* - \mathbf{k}_*^\top \left(\mathbf{I}+\mathbf{W}\mathbf{K}\right)^{-1}\mathbf{W}\mathbf{k}_*. \tag{8}$$

Predicting preference relations y_* are not of interest in the present paper, but are considered in [4].

2.2 Part II: Efficient Sequential Design for Faster personalization

In most machine learning algorithms sequential design (or active learning) aims at maximizing the generalization performance of a model in terms of a specific measure of performance. In this work, the generalization performance is not of particular importance. Instead, the aim is to find the maximum—ideally the global one—of the unknown objective function. For this, a bivariate version of the *expected improvement* [6] (EI) is used given by

$$EI = \sigma_I \phi\left(\frac{\mu_I}{\sigma_I}\right) + \mu_I \Phi\left(\frac{\mu_I}{\sigma_I}\right) \tag{9}$$

with $\mu_I = [\boldsymbol{\mu}_*]_l - [\boldsymbol{\mu}_*]_{\max}$, and $\sigma_I^2 = [\boldsymbol{\Sigma}_*]_{l,l} + [\boldsymbol{\Sigma}_*]_{\max,\max} - 2\cdot[\boldsymbol{\Sigma}_*]_{l,\max}$. The EI is optimized with a gradient descent method with 5 random initializations. By using only 5 random initializations, a little more exploration is build into the sequential designs for robustness.

2.3 Part III: Interface

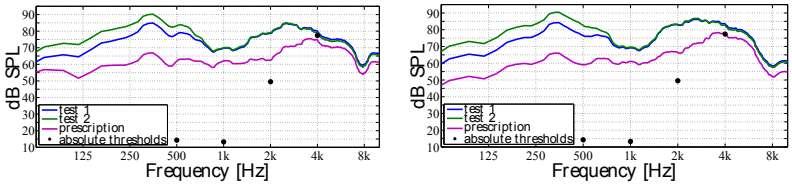
The system relies on the *degree-of-preference* paradigm discussed earlier, and the user interface (PC screen) presents two options to the user, A and B, as illustrated in Fig. 1. The user can now listen to both options, and finally select to which degree A or B is preferences by dragging the sliders to either side.

3 Preliminary Results

The feasibility of the system was evaluated in an experiment where the personalization system was used to find the preferred settings of HAs with several HA users. The preliminary results [1] in Fig. 2

[1] A full analysis of the results is currently in preparation

show a long-term spectra of the sound pressure level (SPL) at the eardrum of a HA user wearing HAs while listening to a piece of music. Each spectrum corresponds to a particular four parameter setting of the HAs. The spectra labeled *test 1* and *test 2* correspond to two HA settings obtained for the user with the personalized system. The spectrum labeled "prescription" corresponds to the setting resulting from current practice using the user's audiogram and the prescription. In a separate test, it was validated that the setting of "Test 2" is significantly ($p_0 < 0.05$) preferred over the setting resulting from the prescription. The system takes about 10 minutes to discover the preferred setting.



(a) Left Ear (b) Right Ear

Figure 2: KEMAR measurements of long-term power spectra of the sound pressure level at the eardrum of a HA user wearing HAs while listening to a piece of music. The user's thresholds at four distinct frequencies are marked with black dotes.

4 Discussion & Conclusion

In this paper, a machine learning based personalization system has been proposed directly addressing a fundamental issue of hearing aid personalization, namely, that the fine-tuning process should be based *directly* an what the hearing impaired perceives. The proposed personalization system appears to be both fast and robust in finding personalized HA settings, that are significantly preferred over standard prescription based first-fit settings. Hence, the system could possibly be a useful fine-tuning supplement in clinics. The system could easily be extended to support other types of user feedback with the Gaussian process framework, such as rankings [7] or absolute scorings [8], instead of pairwise comparisons, although in general, the latter is probably preferable due to its low cognitive load. The proposed Gaussian process based framework is applicable for other than personalization. By changing the active learning criterion to for instance BALD [9], the framework could be used to *generalize*—in constrast to *optimize*—the latent objective function over all settings.

References

[1] H. Dillon, *Hearing Aids*, Boomerang Press, 2nd edition, 2012.
[2] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, 2006.
[3] B. S. Jensen, J. B. Nielsen, and J. Larsen, "Efficient Preference Learning with Pairwise Continuous Observations and Gaussian Processes," *IEEE Workshop MLSP, Beijing*, September 2011.
[4] B. S. Jensen and J. B. Nielsen, "Pairwise Judgements and Absolute Ratings with Gaussian Process Priors," Tech. Rep., November 2011.
[5] J. Vanhatalo and A. Vehtari, "Sparse log gaussian processes via mcmc for spatial epidemiology," in *JMLR Workshop and Conference Proceedings*, 2007, vol. 1, pp. 73–89.
[6] D. R. Jones, "A Taxonomy of Global Optimization Methods Based on Response Surfaces," *Journal of Global Optimization*, vol. 21, no. 4, pp. 345–383, 2001.
[7] Wei Chu and Zoubin Ghahramani, "Extensions of gaussian processes for ranking: semi-supervised and active learning," in *NIPS workshop on Learning to Rank*. 2005, pp. 29–34, Citeseer.
[8] B. S. Jensen, J. B. Nielsen, and J. Larsen, "Bounded gaussian process regression," in *IEEE International Workshop on Machine Learning for Signal Processing*, sep 2013.
[9] N. Houlsby, F. Huszár, Z. Ghahramani, and M. Lengyel, "Bayesian Active Learning for Classification and Preference Learning," *ArXiv e-prints*, Dec. 2011.

Appendix H

Perception-based Personalization of Hearing Aids using Gaussian Processes and Active Learning

---

Jens Brehm Bagger Nielsen, Jakob Nielsen, Jan Larsen. Perception-based Personalization of Hearing Aids using Gaussian Processes and Active Learning. Published in *IEEE/ACM Transactions on Audio, Speech and Language Processing*, January 2015, volume 23, issue 1, page 162-173, ISSN 2329-9290. doi:10.1109/TASLP.2014.2377581.

Perception-based Personalization of Hearing Aids using Gaussian Processes and Active Learning

Jens Brehm Bagger Nielsen[1,2], Jakob Nielsen[1] and Jan Larsen[1,2]

[1]Widex A/S, Nymøllevej 6, DK-3540 Lynge,
{jeb,jnl}@widex.com

[2]Department of Applied Mathematics and Computer Science,
Technical University of Denmark,
Matematiktorvet Building 303B, DK-2800 Kongens Lyngby,
{jenb,janla}@dtu.dk

preprint

Abstract

Personalization of multi-parameter hearing aids involves an initial fitting followed by a manual *knowledge-based* trial-and-error fine-tuning from ambiguous verbal user feedback. The result is an often sub-optimal HA setting whereby the full potential of modern hearing aids is not utilized. This article proposes an interactive hearing-aid personalization system that obtains an optimal individual setting of the hearing aids from direct perceptual user feedback. Results obtained with ten hearing-impaired subjects show that ten to twenty pairwise user assessments between different settings—equivalent to 5-10 min.—is sufficient for personalization of up to four hearing-aid parameters. A setting obtained by the system was significantly preferred by the subject over the initial fitting, and the obtained setting could be reproduced with reasonable precision. The system may have potential for clinical usage to assist both the hearing-care professional and the user.

Hearing Aids, Personalization, Individualization, Gaussian Process (GP), Active Learning, Pairwise Comparisons.

1 Introduction

The complexity of digital signal processing algorithms in hearing-aids (HAs) has increased in the past two decades due to continuous refinement of existing HA algorithms and the addition of new ones. Consequently, the number of associated algorithm parameters has increased and will continue to do so in the future. Algorithm parameters control how the incoming sound is processed by the multitude of algorithms and thereby how the sound is presented to the user. In practice, the multi-parameter adjustment—traditionally referred to as *fitting*—is done in fitting software supplied by the HA company: A restricted

set of meta parameters is available, that controls the entire set of algorithm parameters. The rules defining the mapping from meta parameters in the fitting software to algorithm parameters are covered by a so-called fitting rationale or prescription. Every HA company has their own fitting rationales for their specific HAs. Typically, generic rationales, such as NAL [1] or DSL [2], are available in the software as an option as well. The overall objective of any fitting rationale is to compensate for the user's reduced ability to hear and comprehend speech. A hearing deficit is typically quantified by measuring the reduction in pure-tone hearing threshold level (HTL) in one-octave frequency bands from 500 Hz to 4 kHz relative to normal hearing (NH) [3, Chapter 10]. A user's *audiogram* refers to the pure-tone HTL difference between the user and NH. The HTL differences are specified in *dB of hearing level* (dB HL) [3, Chapter 10]. A 10 dB HL at 500 Hz indicates that the sound pressure level (SPL) at 500 Hz needs to be 10 dB louder compared to the HTL of a normal hearing subject for the user to detect the pure tone. Hence, an audiogram could directly be converted to HA gains in one-octave frequency band. However, due to loudness recruitment [4, Chapter 4-III] and reduced dynamic range [5, Chapter 1] among other factors, it is inappropriate to set gains directly matching the audiogram [5, Chapter 10]. Instead, the audiogram is used as *target gains* for the fitting, implying that the actual HA gain will not compensate fully for the reduced sensitivity. A rationale converts the target gains (or audiogram) into band-dependent and input-level-dependent (non-linear) HA gains.

HA fitting is carried out by a hearing-care professional (HCP) who measures the audiogram e.g. at, 500 Hz, 1 kHz, 2 kHz, and 4 kHz, which thus results in four target gains for the fitting. The target gains—one *set* of meta-parameters—are used to set algorithm parameters like compression ratio, gain of the linear region, and knee-points of the multi-band dynamic processor embedded in modern digital hearing aids. The goal of the fitting is to ensure audibility and optimal speech intelligibility without compromising the user's preferences.

Besides the measured target gains, there are typically additional meta parameters that the HCP can or must adjust related to e.g. noise reduction, multi-channel beamforming, further tailoring of the dynamic compressor etc. [5, Chapter 12]. The HCP will consult the hearing-impaired (HI) client about HA use, rehabilitation, and preferences when adjusting meta-parameters; but they can only be adjusted manually based on the user's often ambiguous descriptions about the perceived sound. The HI client typically finds it difficult to explain his preferences towards sound, hence, it is very challenging to determine the best setting. Furthermore, manual fine tuning is time consuming and thus expensive to perform. In summary, this result in an imminent risk of not exploiting the full potential of modern digital HAs. This provides a great potential for new fine-tuning methods or paradigms which aim at optimal settings for individual users in robust and time-efficient manners.

In this paper, a machine-learning based *interactive HA personalization system* (IHAPS) is proposed. IHAPS optimizes multiple parameters based directly on the user's perception of the sound and not based on a derived verbal ambiguous description. By the active user process of listening to and comparing HA settings, IHAPS enables the user to recognize his preference towards the sound. Active engagement also leads to greater psychological ownership, and thereby to better outcome of the entire hearing impairment therapy [6, 7].

In IHAPS, it is assumed that a user's perception is encoded by an unobserved

internal response function (IRF). Hence, when a user compares two stimuli, the magnitudes of the IRF for the two stimuli determine which of the two stimuli the user prefers or judges to be the best. The IRF cannot be measured directly, and is assumed to be stochastic due to multiple uncontrollable factors. Furthermore, a user's judgments are not fully consistent. Consequently, a user's IRF can only be *estimated* given a set of user assessments of particular stimuli. A particular HA setting, $\mathbf{x}_i$, determines the acoustical stimulus. Hence, the IRF is a function, $f(\mathbf{x})$, of the d HA parameters, $\mathbf{x} = [x_1, ..., x_d]^\top$. Note, that IHAPS can be used both for optimization of meta parameters and of algorithm parameters directly. In the remainder of this article, no distinction between algorithm parameters and meta-parameters will be made. Instead, HA parameters will be considered, which can cover both meta- and algorithm parameters. In IHAPS, the IRF is modeled by a non-parametric Bayesian regression method, viz. a *Gaussian process* (GP) [8], which defines a distribution over flexible non-linear functions, $f(\mathbf{x})$. Users assess settings in a pairwise-comparison paradigm, whereby users do not need to memorize previous ratings, thus resulting in a reduced cognitive load. However, to minimize the number of assessments required to estimate the user's IRF, the user does not only choose which of two particular HA settings that is preferred (forced choice), but also assesses the *degree* of which the setting is preferred over the alternative [9]. For a given set of such *degree-of-preference* assessments (observations), the distribution of a user's IRF is updated [9] and the setting associated with the largest value of the estimated (mean) IRF is suggested as the optimal setting for the user. Hence, from a modeling perspective, the task is to perform global multi-parameter optimization of the user's unobserved IRF with respect to the d HA parameters, $\mathbf{x}$. In IHAPS, global optimization is performed with minimal number of assessments by use of a sequential design in which *active learning* is used to suggest the next two settings to be compared. In summary, IHAPS sequentially loops the following three steps: (1) active learning to determine the optimal next settings to be compared given the current estimate of the user's IRF; (2) user's assessment of the degree-of-preference between the two compared settings; and (3) update of the user's estimated IRF given all past assessments—including the most recent one. When converged or stopped, the suggested optimal setting is given by the setting that maximizes the estimated IRF.

For demonstrating *solely* the potential of IHAPS, two similar studies are conducted in which HA personalization is performed in the case of two and four parameters, respectively. Preliminary results from the four-parameter study have briefly been described in [10]. Both studies considered a music scenario, because music evokes a user's immediate opinion of the quality of the HA-produced sound. Other scenarios, such as a speech scenario, could have been considered as done in [11], but to evaluate IHAPS without several external effects influencing the analysis, the music scenario was considered most suitable. For a real-life application of IHAPS, a multitude of scenarios are relevant including several different stimuli to mimic each scenario. However, these mixed conditions are irrelevant for demonstrating the potential of IHAPS.

Several directions have been pursued for personalization of HAs using for instance a modified simplex procedure [12] or genetic algorithms [13, 14]. However, these initial attempts require unreasonably many assessments to converge, and scale badly with the number of tunable parameters. Almost a decade ago, a probabilistic Bayesian approach was proposed [15], which reads similar to the

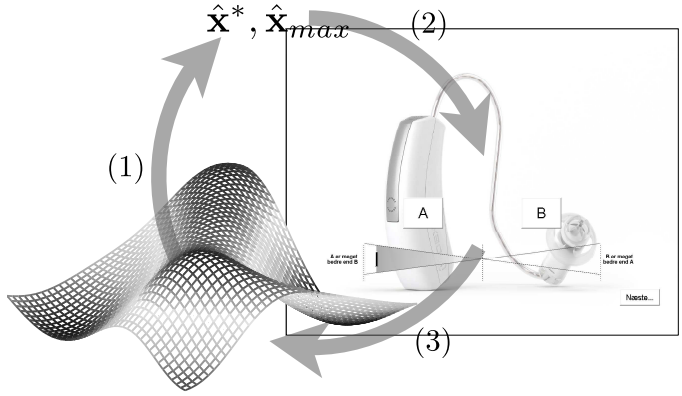


Figure 1: A conceptual overview of the interactive system. At step (1) a new optimal setting is determined based on the current (probabilistic) estimate of the subject's IRF. Next, at step (2) the optimal setting is compared to the setting which maximizes the current estimate of the subject's IRF, and the subject assesses the *degree of preference* between the two settings using a GUI (see Fig. 5). Finally at step (3), the estimate of the subject's IRF is updated based on the recent assessment.

approach proposed in the present paper. However, two fundamental aspects of the approach in [15] are different: Firstly, it assumes that the user's IRF has a known parameterized functional form, which is difficult to qualify in practice. Secondly, assessments are provided in a pairwise forced-choice paradigm using classical choice models [16, 17]. Using an artificial example, Jensen *et al* [9] show that a forced-choice paradigm requires more assessments than the degree-of-preference paradigm. The non-parametric GP approach using the forced choice paradigm [18] has been considered for instance in [19].

2 Personalization System

IHAPS is based on an interactive loop visualized in Fig. 1. The loop essentially contains three parts: (A) Modeling, (B) active learning, and (C) user-interaction.

2.1 Modeling of the User's Internal Response Function with Gaussian Processes

Modeling of the user's IRF is performed in a Bayesian non-parametric framework based on GPs, see e.g. [8]. In the following, the different steps of the non-standard GP framework used in IHAPS to perform regression based on *degree-of-preference* assessments are described. The GP framework is based on previous

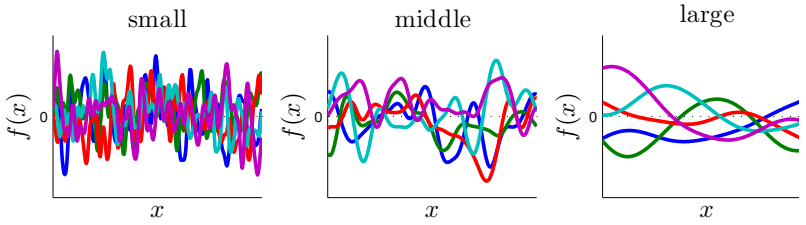


Figure 2: Examples of sampled functions from a Gaussian process with different setting of the smoothness parameter λ, see Eq. 2. For a more thorough treatment of GP smoothness, see [8, Sec. 2.3 & 2.6, Chap. 5]

work [9].

2.1.1 Gaussian Process Priors

A Gaussian process (GP) is a Bayesian non-parametric regression technique, which defines a prior over functions, $f : \mathbb{R}^d \rightarrow \mathbb{R}, \mathbf{x} \mapsto f(\mathbf{x})$, captured in the notation

$$f(\mathbf{x}) \sim \mathcal{GP}\left(0, k(\mathbf{x}, \mathbf{x}')_{\boldsymbol{\theta}_\mathcal{C}}\right), \tag{1}$$

where $k(\cdot,\cdot)_{\boldsymbol{\theta}_\mathcal{C}}$ is the *covariance function*[1] with parameters $\boldsymbol{\theta}_\mathcal{C}$. Generally speaking, the covariance function defines the smoothness of the functions. A commonly used covariance function is the isotropic *squared exponential* (SE) given by

$$k_{SE}(\mathbf{x}, \mathbf{x}') = \sigma_f \exp\left(-\frac{1}{2\lambda}(\mathbf{x} - \mathbf{x}')^\top(\mathbf{x} - \mathbf{x}')\right). \tag{2}$$

A GP is defined as a *collection of random variables, any finite number of which have a joint Gaussian distribution* [8, Definition 2.1], such that a *finite* collection of function values, $\mathbf{f} = [f(\mathbf{x}_1), ..., f(\mathbf{x}_n)]^\top$, for a corresponding set of inputs, $\mathcal{X} = \{\mathbf{x}_i \in \mathbb{R}^d | i = 1, ..., n\}$, has a distribution given by

$$p(\mathbf{f}|\mathcal{X}, \boldsymbol{\theta}_\mathcal{C}) = \mathcal{N}\left(\mathbf{f}|\mathbf{0}, \mathbf{K}\right), \tag{3}$$

where each entry in the $n \times n$ covariance matrix $\mathbf{K}$ is given by $[\mathbf{K}]_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j)_{\boldsymbol{\theta}_\mathcal{C}}$ and $\mathcal{N}\left(\mathbf{z}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$ denotes the multi-variate normal probability density function[2]. Functions sampled from a GP prior with different settings of the *smoothness* parameter, λ, are depicted in Fig. 2. By specifying the likelihood, $p(\mathcal{Y}|\mathbf{f}, \boldsymbol{\theta}_\mathcal{L})$, of some set of observations, $\mathcal{Y}$, given the finite collection of function values, $\mathbf{f}$, the posterior distribution over the function values $\mathbf{f}$ is given by Bayes formula:

$$p(\mathbf{f}|\mathcal{Y}, \mathcal{X}, \boldsymbol{\theta}) = \frac{p(\mathcal{Y}|\mathbf{f}, \boldsymbol{\theta}_\mathcal{L})p(\mathbf{f}|\mathcal{X}, \boldsymbol{\theta}_\mathcal{C})}{p(\mathcal{Y}|\mathcal{X}, \boldsymbol{\theta})} \tag{4}$$

$$= \frac{p(\mathcal{Y}|\mathbf{f}, \boldsymbol{\theta}_\mathcal{L})p(\mathbf{f}|\mathcal{X}, \boldsymbol{\theta}_\mathcal{C})}{\int p(\mathcal{Y}|\mathbf{f}, \boldsymbol{\theta}_\mathcal{L})p(\mathbf{f}|\mathcal{X}, \boldsymbol{\theta}_\mathcal{C})d\mathbf{f}}, \tag{5}$$

[1] In the literature, several expressions are used for the covariance function, such as *kernel function* or simply *kernel*.

[2] In this paper, $\mathbf{z} \sim \mathcal{N}\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$ and $p(\mathbf{z}) = \mathcal{N}\left(\mathbf{z}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$ are equivalent.

where the hyper parameters, $\boldsymbol{\theta} = \{\boldsymbol{\theta}_{\mathcal{L}}, \boldsymbol{\theta}_{\mathcal{C}}\}$, contain both likelihood and covariance parameters.

2.1.2 Likelihood Function

In previous work [9], modeling of continuous bounded responses is performed with a likelihood function based on a re-parameterized beta distribution specifically applicable in cases where observations are pairwise degree-of-preference assessments. Thus, the framework is specifically applicable for the present work.

Progressing as in [9], consider a set of pairwise observations, $\mathcal{Y} = \{y_k \in (0,1) | k = 1, ..., m\}$, of the *degree of preference* between any two distinct inputs, $u_k, v_k \in \{1, ..., n\}$, implying that $\mathbf{x}_{u_k}, \mathbf{x}_{v_k} \in \mathcal{X}$. An increasing preference for the first option, u_k, is reflected by $y_k \to 0$, whereas an increasing preference for the second option, v_k, is reflected by $y_k \to 1$. No preference is indicated by $y_k = 0.5$. A suitable likelihood function, $p(y_k|\mathbf{f}_k)$, is now constructed given the function values for the two input instances, $\mathbf{f}_k = [f(\mathbf{x}_{u_k}), f(\mathbf{x}_{v_k})]^\top$, by re-parameterizing the beta distribution, $\text{Beta}(\,\cdot\,|\alpha, \beta)$, as

$$p(y_k|\mathbf{f}_k, \boldsymbol{\theta}_{\mathcal{L}}) = \text{Beta}\left(y_k|\nu\zeta(\mathbf{f}_k, \sigma), \nu(1 - \zeta(\mathbf{f}_k, \sigma))\right), \tag{6}$$

where $\boldsymbol{\theta}_{\mathcal{L}} = \{\nu, \sigma\}$ is the set of likelihood parameters. ν is a dispersion parameter around the mean, $\zeta(\mathbf{f}_k, \sigma)$. The mean is defined by

$$\zeta(\mathbf{f}_k, \sigma) = \Phi\left(\frac{f(\mathbf{x}_{v_k}) - f(\mathbf{x}_{u_k})}{\sqrt{2}\sigma}\right), \tag{7}$$

where $\Phi(\cdot)$ is the standard normal cumulative density function—with zero mean and unit variance—and σ is a slope parameter. The likelihood function is visualized in Fig. 3.

By assuming that observations are independent given the latent function values $\mathbf{f}$, the likelihood is written as

$$p(\mathcal{Y}|\mathbf{f}, \boldsymbol{\theta}_{\mathcal{L}}) = \prod_{k=1}^{m} p(y_k|\mathbf{f}_k, \boldsymbol{\theta}_{\mathcal{L}}), \tag{8}$$

which is plugged into Eq. (4) together with the GP prior from Eq. (3) to completely specify the Bayesian model.

2.1.3 Posterior Inference and Model Training

The Gaussian process model described above is analytically intractable due to the integral in Eq. (5). Therefore, approximate inference is performed based on the Laplace approximation following [9].

The idea of the Laplace approximation [8, Section 3.4] is to approximate the intractable posterior, $p(\mathbf{f}|\mathcal{Y}, \mathcal{X}, \boldsymbol{\theta})$, from Eq. (4) with a Gaussian $q(\mathbf{f}|\mathcal{Y}, \mathcal{X}, \boldsymbol{\theta})$ of the form

$$p(\mathbf{f}|\mathcal{Y}, \mathcal{X}, \boldsymbol{\theta}) \approx q(\mathbf{f}|\mathcal{Y}, \mathcal{X}, \boldsymbol{\theta}) = \mathcal{N}\left(\mathbf{f}|\hat{\mathbf{f}}, \mathbf{A}^{-1}\right), \tag{9}$$

where $\hat{\mathbf{f}}$ is the posterior maximum (mode) and $\mathbf{A}$ is the Hessian of the negative log posterior at the mode. The mode is found by maximizing the unnormalized

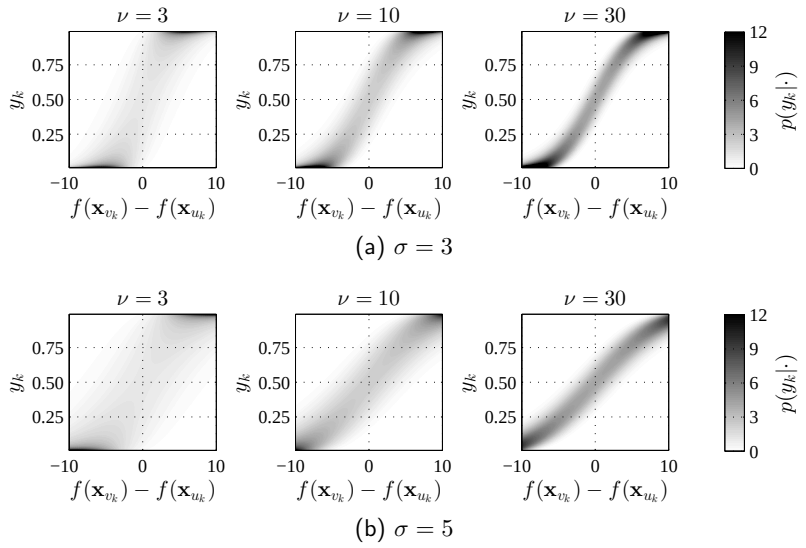


Figure 3: Visualization of the Beta likelihood $p(y_k|\mathbf{f}_k, \boldsymbol{\theta}_\mathcal{L})$ function for three different settings of the dispersion parameter ν and two different settings of the slope parameter σ.

log-posterior given by

$$\psi\left(\mathbf{f}|\mathcal{Y},\mathcal{X},\boldsymbol{\theta}\right) = \log p\left(\mathcal{Y}|\mathbf{f},\boldsymbol{\theta}_\mathcal{L}\right) - \frac{1}{2}\mathbf{f}^\top\mathbf{K}^{-1}\mathbf{f} - \frac{1}{2}\log|\mathbf{K}| - \frac{n}{2}\log 2\pi, \tag{10}$$

with a Newton method. The Newton step is given by

$$\mathbf{f}^{new} = \left(\mathbf{K}^{-1} + \mathbf{W}\right)^{-1}\left[\mathbf{W}\mathbf{f} + \nabla\log p(\mathcal{Y}|\mathbf{f},\boldsymbol{\theta}_\mathcal{L})\right], \tag{11}$$

where $[\mathbf{W}]_{i,j} = -\sum_{k=1}^{m}\nabla\nabla_{i,j}\log p\left(y_k|\mathbf{f}_k,\boldsymbol{\theta}_\mathcal{L}\right)$ defining $\nabla\nabla_{i,j} \equiv \frac{\partial^2}{\partial f(\mathbf{x}_i)\partial f(\mathbf{x}_j)}$. Note, that unlike traditional classification and regression problems, $\mathbf{W}$ is not diagonal due to the pairwise structure. For derivatives and further details, see [20].

When Eq. (11) has converged, the approximation is simply

$$p(\mathbf{f}|\mathcal{Y},\mathcal{X},\boldsymbol{\theta}) \approx q(\mathbf{f}|\mathcal{Y},\mathcal{X},\boldsymbol{\theta}) = \mathcal{N}\left(\mathbf{f}|\hat{\mathbf{f}},\left(\mathbf{W}+\mathbf{K}^{-1}\right)^{-1}\right) \tag{12}$$

Traditionally, training of GPs are performed by optimizing the marginal likelihood, $p(\mathcal{Y}|\mathcal{X},\boldsymbol{\theta})$, from Eq.(4) with respect to the hyper parameters, $\boldsymbol{\theta}$. This is referred to as ML-II optimization [8, Chapter 5.2]. In the present paper, a slightly different scheme is used, in which the optimization is regularized by hyper priors, $p(\boldsymbol{\theta})$, over the parameters in what is a maximum-a-posterior-like (MAP-II) scheme following [8, Chapter 5.2]. More precisely, the parameters $\boldsymbol{\theta}^{\text{MAP-II}}$ in the trained GP are given by

$$\begin{aligned}\boldsymbol{\theta}^{\text{MAP-II}} &= \arg\max_{\boldsymbol{\theta}}\log q(\boldsymbol{\theta}|\mathcal{Y},\mathcal{X}) \\ &\approx \arg\max_{\boldsymbol{\theta}}\log p(\boldsymbol{\theta}|\mathcal{Y},\mathcal{X}) = \arg\max_{\boldsymbol{\theta}} p(\boldsymbol{\theta}|\mathcal{Y},\mathcal{X}),\end{aligned} \tag{13}$$

where the intractable log posterior over the parameters, $\log p(\boldsymbol{\theta}|\mathcal{Y},\mathcal{X})$, is approximated by $\log q(\boldsymbol{\theta}|\mathcal{Y},\mathcal{X})$, which is—up to a normalization constant—given by

$$\log q(\boldsymbol{\theta}|\mathcal{Y},\mathcal{X}) \propto \log q(\mathcal{Y}|\mathcal{X},\boldsymbol{\theta}) + \log p(\boldsymbol{\theta}). \tag{14}$$

In Eq. (14), $q(\mathcal{Y}|\mathcal{X},\boldsymbol{\theta})$ is the Laplace approximation to the (intractable) marginal likelihood, $p(\mathcal{Y}|\mathcal{X},\boldsymbol{\theta})$, from Eq. (4), resulting in

$$\log q(\boldsymbol{\theta}|\mathcal{Y},\mathcal{X}) \propto \log p(\mathcal{Y}|\hat{\mathbf{f}},\boldsymbol{\theta}_{\mathcal{L}}) - \frac{1}{2}\hat{\mathbf{f}}^{\top}\mathbf{K}^{-1}\hat{\mathbf{f}} - \frac{1}{2}\log|\mathbf{I}+\mathbf{K}\mathbf{W}| + \log p(\boldsymbol{\theta}). \tag{15}$$

Hence, training of the GP model consists of the following two steps which are looped until convergence[3]:

1: With fixed hyper parameters, $\boldsymbol{\theta}^{\text{MAP-II}}$, repeat Eq. (11) to find the mode of the Laplace approximation and use Eq. (12) to approximate the posterior.

2: Given the approximate posterior from Eq. (12), optimize the right hand side of Eq. (15) with respect to $\boldsymbol{\theta}$ using a BFGS gradient method to obtain the hyper parameters $\boldsymbol{\theta}^{\text{MAP-II}}$.

The specific choices of kernel, $k(\cdot,\cdot)_{\boldsymbol{\theta}_{\mathcal{C}}}$, and hyper priors, $p(\boldsymbol{\theta})$ are specific to each experiment, and are therefore explained as part of Sec. 3 and Sec. 4.

2.1.4 Predictions

The overall goal of almost any machine learning algorithm once trained, is to make prediction for future inputs. In a regression setting, predictions consist of predicting the function values, $\mathbf{f}_* = [f(\mathbf{x}_1^*),...,f(\mathbf{x}_o^*)]^{\top}$, at new input locations, $\mathcal{X}_* = \{\mathbf{x}_l^* \in \mathbb{R}^d | l = 1,...,o\}$. In a Bayesian framework, the variables are considered stochastic. Therefore in a Bayesian framework, predictions are formulated in terms of the predictive distribution, $p(\mathbf{f}_*|\mathcal{Y},\mathcal{X}_*)$, from which different statistics about the variables can be accessed, such as the mean value, $\boldsymbol{\mu}^*$, and (co)variance, $\boldsymbol{\Sigma}^*$.

Given the GP, the joint prior distribution between the predictive and training function values is given by

$$\begin{bmatrix}\mathbf{f}\\ \mathbf{f}_*\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\end{bmatrix}, \begin{bmatrix}\mathbf{K} & \mathbf{K}_*\\ \mathbf{K}_*^{\top} & \mathbf{K}_{**}\end{bmatrix}\right), \tag{16}$$

where $[\mathbf{K}_{**}]_{l,r} = k(\mathbf{x}_l^*,\mathbf{x}_r^*)$ and $[\mathbf{K}_*]_{i,l} = k(\mathbf{x}_i,\mathbf{x}_l^*)$. Given Eq. (16), the conditional distribution of $\mathbf{f}_*|\mathbf{f}$ is Gaussian, hence

$$p(\mathbf{f}_*|\mathcal{Y},\mathcal{X},\mathcal{X}_*,\boldsymbol{\theta}) = \int p(\mathbf{f}_*|\mathbf{f},\mathcal{X},\mathcal{X}_*,\boldsymbol{\theta})q(\mathbf{f}|\mathcal{Y},\mathcal{X},\boldsymbol{\theta})d\mathbf{f} = \mathcal{N}(\mathbf{f}_*|\boldsymbol{\mu}_*,\boldsymbol{\Sigma}_*) \tag{17}$$

which is an integral over the product of two Gaussian distribution, which is again Gaussian. The solution is found in for instance [18, Eq. (17)-(18)] and is given as

$$\boldsymbol{\mu}_* = \mathbf{K}_*^{\top}\mathbf{K}^{-1}\hat{\mathbf{f}} \tag{18}$$

$$\boldsymbol{\Sigma}_* = \mathbf{K}_{**} - \mathbf{K}_*^{\top}\left(\mathbf{K}+\mathbf{W}^{-1}\right)^{-1}\mathbf{K}_* = \mathbf{K}_{**} - \mathbf{K}_*^{\top}\left(\mathbf{I}+\mathbf{W}\mathbf{K}\right)^{-1}\mathbf{W}\mathbf{K}_* \tag{19}$$

[3]The similarity with the Expectation-Maximization algorithm is that step 1 can be recognized as the E-Step, and step 2 as the M-step.

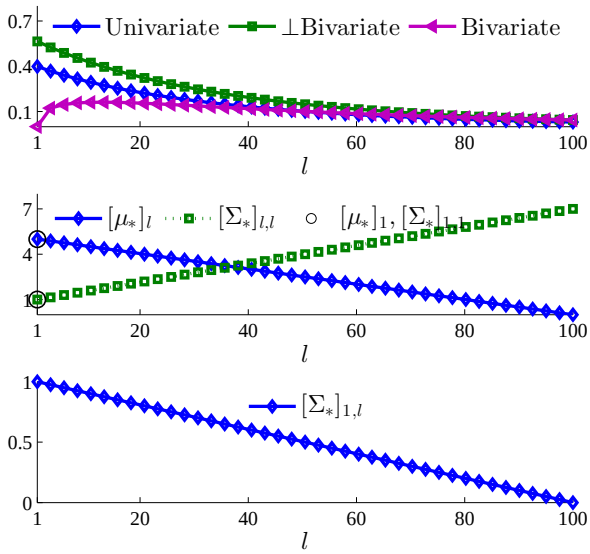


Figure 4: Illustration of the difference between the univariate and bivariate EI. The current maximum (indicated by circles) is at $l = 1$, which is also a possible query. **Top**: EI for the standard version (Univariate), a bi-variate version neglecting covariance (⊥Bivariate) and a full bivariate version incorporating covariance (Bivariate). **Middle**: mean and variance for query points, $\mathbf{x}_l^*$. **Bottom**: covariance between query $\mathbf{x}_l^*$ and maximum $\mathbf{x}_1$ used in the (full) bivariate EI. Note that the full bivariate EI is zero at the current maximum; hence, avoids querying this point again.

where the last expression is not given in [18], but is numerically more stable, because it avoids inverting $\mathbf{W}$. Eq. (18) is used as the estimator of the user's IRF. In addition, the covariance from Eq. (19) is utilized to formulate an active learning criterion used to select the next input $\hat{\mathbf{x}}^*$ actively, to constitute the next $(k+1)$ comparison.

Prediction of preference relations, y_*, can be done but is not of particular interest in the present paper, see further [20].

2.2 Sequential Design for Optimization

Sequential design (or active learning) is used to reduce the required number of training examples by sequentially including new *informative* training examples with respect to some criterion[4]. Traditionally, active learning is used when labeling of data is expensive and is done sequentially.

As for most machine learning algorithms, typically, sequential designs aim at maximizing the generalization performance of a model often formulated in terms of a specific criterion. A Bayesian criterion is the expected reduction in posterior

[4]Some active learning methods, such as query-by-committee, do not have an explicit criterion, but this is beyond the scope of the present article to discuss.

Shannon entropy after inclusion of a new training example [21]. In this work, the generalization performance is not of particular importance. Instead, the aim is to find a maximum—ideally the global one—of the unknown IRF modeled by the GP. For this, a novel bivariate version of the *expected improvement* [22] is used. Expected improvement (EI) is derived by defining improvement, I, as the difference in function values between the current maximum $\hat{f} \equiv f(\hat{\mathbf{x}}_{\max})$ (typically only among the training cases $\mathcal{X}$) and a query point $f_l^* \equiv f(\mathbf{x}_l^*)$

$$I \equiv f_l^* - \hat{f}. \tag{20}$$

Now, EI is the expectation of the (positive) improvement (which is normally distributed in the present model)

$$\begin{aligned} EI &\equiv \mathbb{E}_{p(I)}\{\max(I,0)\} = \int_0^\infty I p(I) dI = \int_0^\infty I \mathcal{N}\left(I|\mu_I, \sigma_I^2\right) dI \\ &= \sigma_I \mathcal{N}\left(\frac{\mu_I}{\sigma_I}\middle| 0,1\right) + \mu_I \Phi\left(\frac{\mu_I}{\sigma_I}\right) \end{aligned} \tag{21}$$

In standard EI, $\hat{f}$ is not considered stochastic, hence the distribution $p(I)$ is just a univariate normal with mean $\mu_I = [\boldsymbol{\mu}_*]_l - \hat{f}$ and variance $\sigma_I^2 = [\boldsymbol{\Sigma}_*]_{l,l}$ [22]. In this article, the joint distribution between the query and maximum is taken into consideration. In this case, $p(I)$ is the difference between two dependent normal distributed random variables, and is thus given by

$$\begin{aligned} p(I) &= \mathcal{N}\left(I|\mu_I, \sigma_I^2\right), \quad \text{where} \\ \mu_I &= [\boldsymbol{\mu}_*]_l - [\boldsymbol{\mu}_*]_{\max} = [\boldsymbol{\mu}_*]_l - \hat{f} \\ \sigma_I^2 &= [\boldsymbol{\Sigma}_*]_{l,l} + [\boldsymbol{\Sigma}_*]_{\max,\max} - 2 \cdot [\boldsymbol{\Sigma}_*]_{l,\max}. \end{aligned} \tag{22}$$

The difference between the univariate and bivariate EI, i.e., whether to including the covariance between the query and the maximum (last term in Eq. (22)), is illustrated in a small example in Fig. 4. In this example, the current maximum point corresponds to $l = 1$ and has larger mean value than all other query points ($l \neq 1$), but smaller variance. This is a typical scenario in GP modeling. In this scenario, neglecting the covariance has the undesirable effect of querying the already observed maximum point, causing the active learning to "get stuck". The (full covariance) bivariate EI avoids this, but has the same properties when maximum and query points are independent. For GP models, predictions are typically very dependent when inputs are close to each other. Hence, the bivariate version is not as local as the standard univariate EI. In the following, EI refers to the full bivariate version.

A user's optimal setting is essentially unknown. Therefore, it is not possible to measure how close an optimal setting suggested by IHAPS is to the true optimal setting. However, the average EI over possible queries, $\mathbf{x}_l^*$, is in IHAPS used as a prediction of convergence (convergence measure). Intuitively, when the average EI is zero or close to zero, no further improvement is to be expected from another setting. Thereby, no other setting is under the predictive distribution expected to be preferred over the current optimal setting, $\hat{\mathbf{x}}_{\max}$.

2.3 Graphical User Interface

The graphical user interface (GUI) by which the users interact with IHAPS during experiments is depicted in Fig. 5. An important property of the GUI

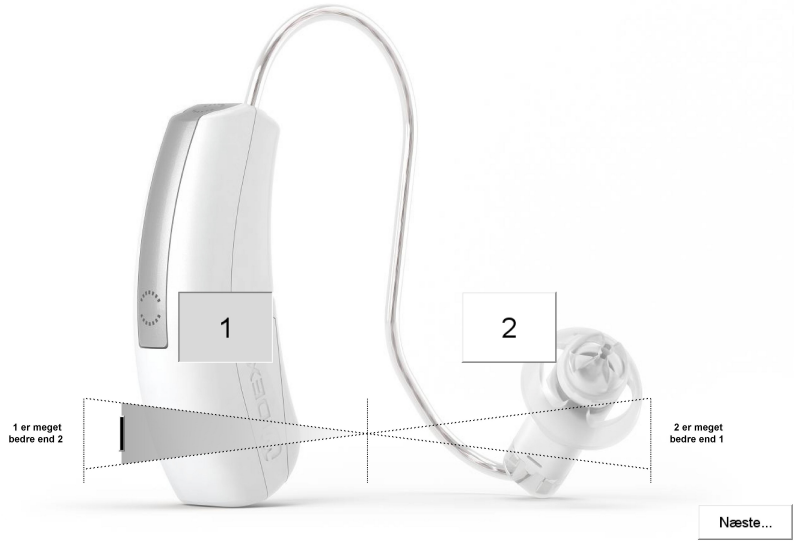


Figure 5: The pairwise graphical user interface (GUI) used for the experiments. A slider is used to capture the *degree-of-preference* for either setting '1' or '2'. The user can listen to setting '1' or '2' by pressing the corresponding button. A gray button indicates that the corresponding setting is selected and thus active in the HAs. When the user is satisfied with the position of the slider, the button in the lower-right corner 'Nste' is pushed to confirm the current assessment. Next IHAPS computes the next comparison with two new settings corresponding to '1' and '2' until a prescribed number of iterations is reached.

is that users shall generally find it intuitive and easy to use. Therefore, the placements of buttons and sliders are arranged to indicate the pairwise nature of the assessments. The slider is designed as a *mirrored volume control* to indicate that the preference is increasing towards the end points of the slider.

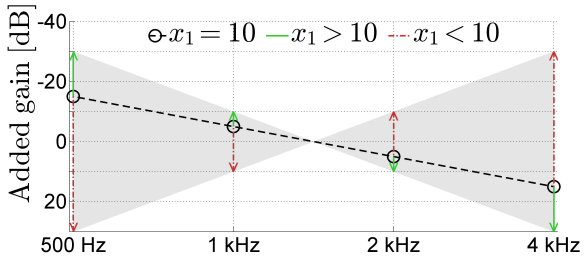
3 Study 1: Two-Dimensional Optimization

In the first study the subjectively best target gains of the HA fitting for the four basic frequency bands—500 Hz, 1 kHz, 2 kHz, and 4 kHz—were learned indirectly by modifying the target gains[5] with two meta-parameters $x_1, x_2 \in \{-20; 20\}$. The meta-parameters were learned while the subject listened to a 32 sec. looping music clip[6]. The parametrization is visualized in Fig. 6. For a specific parametrization, x_1 and x_2, the resulting target gains (in dB) for the four frequency bands are computed as the sum of the two sets of added gains in Fig. 6 (a) and (b) and the measured audiogram[7]. Together x_1 and x_2 define

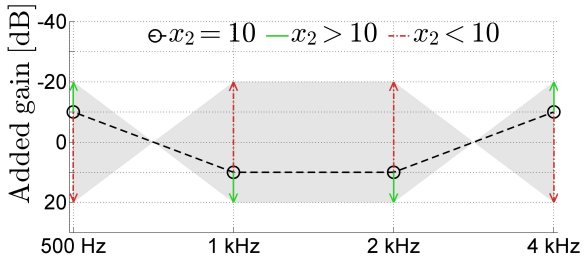
[5] In the WIDEX® fitting software the target gains are set in what is called a SENSOGRAM, see http://www.widex.pro/en/fitting-systems/compass/in-situ-tools/sensogram/.

[6] Teitur, "Sleeping with the Lights on", Poetry & Aeroplanes, 2003. Start at sec. 6. End at sec. 38.

[7] The audiogram is measured binaurally, hence the left and right ears are fitted individually.



(a) Contribution of x_1



(b) Contribution of x_2

Figure 6: Two meta-parameters, x_1 and x_2, are used to modify the target gains in the four basic frequency bands of the HA fitting, as shown in (a) and (b), respectively. Note, that a particular subjects resulting target gains (in dB) for the four frequency bands is the sum of the measured audiogram (binaural) and the added gains specified by the selection of $x_1 \in \{-20; 20\}$ and $x_2 \in \{-20; 20\}$. The grey shaded areas show the added gain limits when varying the meta-parameters in their intervals.

how the audiogram is shaped for the particular piece of music, and IHAPS was used to obtain optimal shapes for the individual subject as quickly as possible.

3.1 Algorithm Details

Although the described framework from Sec. 2 is highly generic, there are still a few properties left to define. The modeling part was set up by defining the covariance function and hyper-prior distributions:

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{x}')^\top \mathbf{P}^{-1}(\mathbf{x} - \mathbf{x}')\right), \tag{23}$$

$$\text{with } \mathbf{P} = \text{diag}([\lambda_1, ..., \lambda_d]^\top), \tag{24}$$

$$\sigma_f \sim p([\boldsymbol{\theta}_\mathcal{C}]_{d+1}) = \delta(\sigma_f = 4), \tag{25}$$

$$\lambda_i \sim p([\boldsymbol{\theta}_\mathcal{C}]_i) \ = \text{half-St}(\lambda_i; 6, 10), \tag{26}$$

$$\sigma \sim p([\boldsymbol{\theta}_\mathcal{L}]_1) = \text{half-St}(\sigma; 6, 10), \tag{27}$$

$$\nu - 2 \sim p([\boldsymbol{\theta}_\mathcal{L}]_2) = \text{half-St}(\nu - 2; 6, 10), \tag{28}$$

where half-St$(z;\xi,s) \propto \left(1+\frac{1}{\xi}\left(\frac{z}{s}\right)^2\right)^{-(\xi+1)/2}$ is the half Student's t-distribution [23, 24, 25] with ξ degrees of freedom and scale s. The above kernel and hyper-prior distribution are common choices, see e.g. [8, 23]. The hyper parameters were learned by optimizing Eq. (15) using a gradient ascend method with initial values $\sigma_f = 4, \lambda_i = 5, \sigma = \exp(1), \nu = 2 + \exp(1)$. The parameters of the hyper-prior distributions and the initialization of the hyper parameters were not tuned to perfection, but were set from a few initial experiments with normal-hearing subjects. It turns out that the framework is not overly sensitive to this tuning.

For the active learning part, the EI from Eq. (21) was not directly maximized. Instead, the EI was calculated for all possible $\mathbf{x}_l^*$ in a grid and collected in $\mathbf{EI}$ such that $[\mathbf{EI}]_l$ contained the EI for $\mathbf{x}_l^*$. The evaluation of the EI for the entire grid was computationally feasible, since $d = 2$ is small. A uniform grid from -20 to 20 with a step size of 1 was used for both x_1 and x_2, hence $\mathcal{X}_* = \{[-20:1:20]^2\}$. Now, the index $\hat{l}$ of the setting $\hat{\mathbf{x}}_l^* \in \mathcal{X}_*$ to add to the next comparison was determined by once drawing a vector $\boldsymbol{\ell}$ of length $o = 41^2$ of binary variables with exactly one nonzero component from a multinomial distribution given by

$$\boldsymbol{\ell}|\mathbf{EI} \sim \text{Mult}\left(\frac{1}{\sum_{l'=1}^{o}[\mathbf{EI}]_{l'}} \cdot \mathbf{EI}\right). \tag{29}$$

The index $\hat{l}$ was thus given by the index of the nonzero component of $\boldsymbol{\ell}$. Compared to maximization of the EI, a little randomness[8] was introduced. Recall, that the bivariate EI compared to the univariate EI avoids querying the current maximum over and over again for the entire test. The extra randomness imposed by the multinomial sampling avoids querying settings *too* close to the current maximum too often towards the end of each test.

3.2 Procedure

Every iteration consisted of a comparison between the actively sampled new setting, $\hat{\mathbf{x}}_l^*$, and the current best setting, $\hat{\mathbf{x}}_{\max}$, among the training set, $\mathcal{X}$. To remove a possible bias effect, the two settings were randomly assigned to option 1 and 2. A single test consisted of 30 iterations/comparisons which were the desired maximal number of iterations to achieve an optimal setting. Two tests, *Test 1* and *Test 2*, were conducted to show the reproducibility of the found optima. Prior to the two tests, the subjects rated 10 comparisons between randomly chosen settings. This *training session* was used only to give the subjects an opportunity to learn how to use the setup and how the sound in the HAs varied. Following the two tests, a *significance test* was conducted to investigate if the optimal setting of IHAPS was significantly preferred by the subject over a baseline setting. The significance test used twenty repeated forced-choices between the optimal setting and the baseline setting. In each repetition they were assigned randomly to the two options presented to the subject. Significance was tested with an exact two-tailed binomial test. The optimal setting was taken from *Test 2* unless this test did not converge. In that

[8] Typically, finding the best trade-off between exploitation (utilize the model) and exploration (reduce uncertainty) is the main challenge in active learning.

case the optimal setting was taken from *Test 1.* A natural baseline setting is the setting with target gains equal to the subject's audiogram (i.e., $x_1 = 0$, $x_2 = 0$), since this is the standard setting of the HAs without additional personalization. Settings were automatically uploaded to the HAs using proprietary WIDEX® software.

WIDEX® PASSION440 HAs equipped with RIC 1-Receivers were used in all tests with all subjects. CRET-S soft earmolds (without vent) were constructed individually to each subject to obtain a closed fitting. The HAs were fitted initially using the measured audiogram, with an omni-directional beamformer, noise reduction and speech enhancement turned off and slow-acting less-aggressive feedback cancellation (FBC) particular suitable for music[9].

To avoid placebo effects in the final significance test, subjects were not informed that the aim of the experiments was to optimize the setting of the HAs based on their feedback. Instead, they were informed that they, in a sequence of pairwise comparisons between different settings in the HAs, should judge which settings they preferred and how much. It was emphasized that the judgments should only reflect their subjective opinion. Likewise, subjects were not primed to focus or pay attention to specific things in the music.

3.3 Results

Table 1: Optimal parameter settings (x_1, x_2) for Test 1 and Test and corresponding significance levels.

Subject	Age	Test 1	Test 2	$p_0 <$
#1	55	NC	(−20,−20)	**0.001**
#2	58	(−20,−16)	(−16,−12)	**0.001**
#3	57	(−16,−16)	(−14,−16)	**0.001**
#4	71	(−20,−12)	(−16,−8)	**0.001**
#5	66	(−18,−14)*	(−18,−2)*	**0.001**
#6	77	NC	NC	NC
#7	45	NC	NC	NC
#8	45	(0,−14)	(−4,−12)	**0.001**
#9	35	(0,−18)	(−8,−10)*	**0.001**
#10	53	(−18,−14)	(−20,−6)	**0.001**

NC: Not converged, average EI is clearly non-zero.

*: Average EI not completely zero.

The best settings found in the two consecutive tests and the results of the significance tests are shown in Table 1. NCs indicate tests that did not converge according to the average EI convergence measure. Asterisk symbols denote tests in which the convergence measure did not completely reach zero (see Fig. 7a).

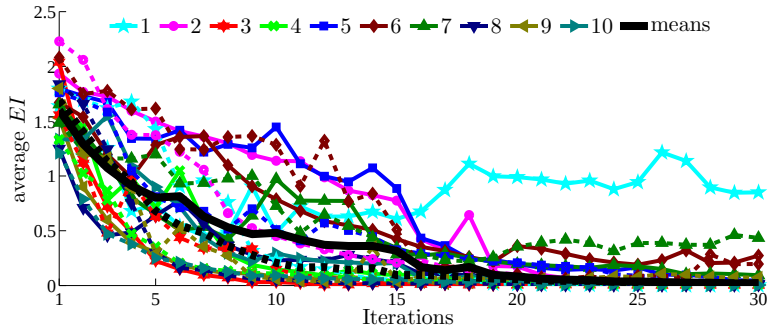
[9]This FBC setting is obtained in the WIDEX® software with the "SuperGain Music" setting of the FBC

The optimal settings (from converged tests) transformed to actual target-gains shapings are shown in 7b.

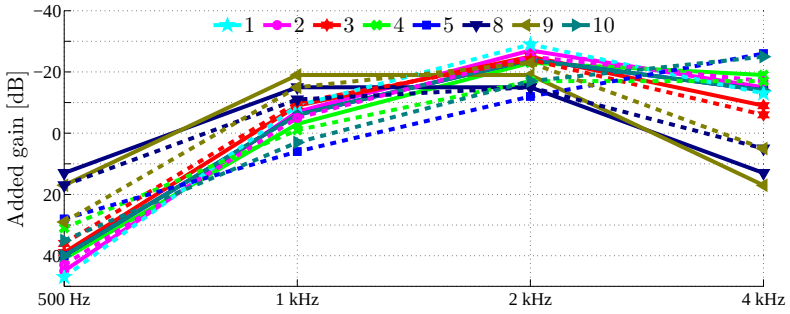
Fig. 8 and Fig. 9 show the IRF predictions and the EI after 30 and 16 iterations, respectively, for subject 4 from Test 1 and Test 2. Additional details are found in [26].

3.4 Discussion

During the tests, some interesting observations were made. Firstly, subject 6 clearly was not able to consistently distinguish between different settings, which is also reflected in the convergence measure (see Fig. 7a). Secondly, subject 7 expressed that he was in conflict with himself during the experiments. Sometimes he preferred a more richer but resounding sound, while other times he preferred a more flat and neutral sound. Unfortunately, he was unable to make up his mind and switched several times between the two types of listening

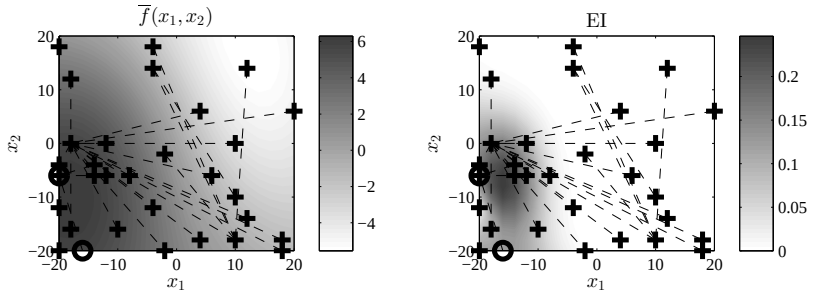


(a) Convergence measure

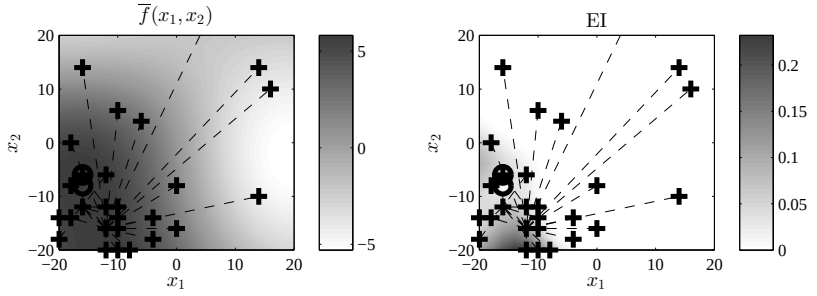


(b) Optimal target-gains shapings

Figure 7: Solid lines correspond to Test 1 and dotted lines correspond to Test 2. (a) The convergence measure for each subject calculated as $\sum_{l'=1}^{o}[\mathbf{EI}]_{l'}$ plotted as a function of iterations and means across subjects that converged (see Tab. 1). (b) Added target-gains shapings given the optimal parameters for the tests that converged (see Table. 1).



(a) Test 1

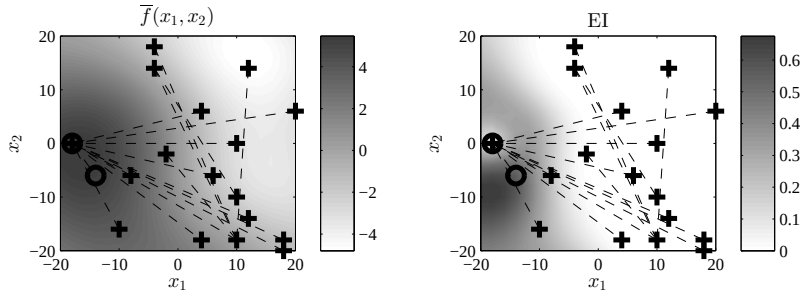


(b) Test 2

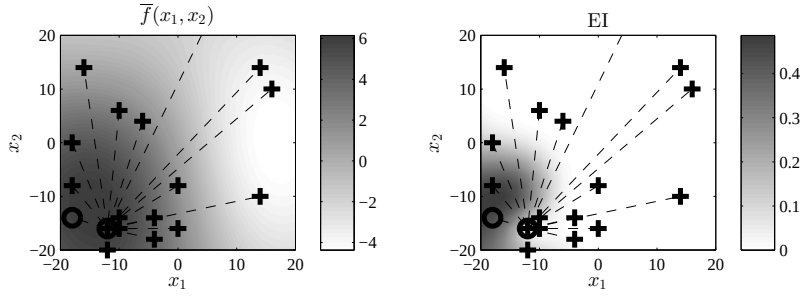
Figure 8: **Reproducibility:** Predictions of the IRF (left figures) and the EI (right figures) for subject 4 after the 30'th (final) iteration from (a): test 1 and from: (b) test 2. Crosses indicate observations, dotted lines indicate comparisons and circles show the suggested next comparison (although the test stops at this point).

strategies. Consequently, IHAPS found him to be inconsistent and did not converge. Thirdly, due to a numerical issue, the active criterion was not working properly in the last part of the second test with subject 9. Therefore, this test did not completely convergence to zero. This is somewhat misleading, because effectively, the last part of the examples presented to subject 9 in the second test was chosen randomly due to the numerical issue. Indirectly, IHAPS thus refrained from *optimization* and performed *generalization* instead. Without the numerical issue, the second test with subject 9 probably would have converged completely based on her behavior from the first test (see Fig. 7a).

Generally, IHAPS was able to obtain a personalized setting of the two meta-HA parameters for eight of the ten hearing-impaired subjects. Obviously, if a user does not have a consistent preference or is unable to distinguish any settings from each other, IHAPS cannot and shall not obtain an optimal setting. Ideally, IHAPS should be able to identify when it has not obtained an optimal setting for the user, i.e., that a session does not converge. Since, each user's true optimum is unknown, only well-founded speculations can be made. Nevertheless, for all tests that converged according to the average EI, IHAPS suggested a setting that the subject preferred significantly over the prescribed setting. Furthermore,



(a) Test 1



(b) Test 2

Figure 9: **Convergence:** Predictions of the IRF (left figures) and the EI (right figures) for subject 4 after the 16'th iteration from (a): test 1 and from: (b) test 2. Crosses indicate observations, dotted lines indicate comparisons and circles show the suggested next comparison.

,

the session of subject 7 did not converge and he did not prefer the suggested setting over the baseline. The comments given after the test by subject 7 explain why the system was unable to obtain an optimal setting for subject 7, as the system cannot deal with subjects that change their opinion during a test. It is speculated that if subject 7 had indicated that two (or even a couple) of settings were equally good, the test would have been successful. This might have been achieved with more thorough instructions about what the test was actually about. For instance, subject 7 could have been instructed to intentionally stick with only one objective at the time, instead of switching between them during a single test. However, this would have biased the results. Nevertheless, it is desirable that IHAPS by the average EI seems to indicate if a test successfully obtains a (near) optimal setting, even though the average EI appears to be somewhat conservative (indicated by asterixes in Fig. 7a).

The reproducibility is actually better than what can be concluded from inspection of the suggested optimal settings only, indicated by studying the close resemblance between the predicted IRF of the two tests (shown for subject 4 in Fig. 8a and Fig. 8b). Furthermore, by comparing Fig. 9 and Fig. 8 it is observed that the IRF for subject 4 was already quite well captured halfway through both

tests. Generally, this was common to all successful tests and illustrates that the average EI is somewhat conservative for predicting of convergence.

4 Study 2: Four-Dimensional Optimization

The setup for the second study was similar to the first study described in Sec. 3. However, instead of modifying the four target gains by two meta parameters, all four target gains were in both hearing aids defined directly ranging from 0 to 80 dB HTL (hearing threshold level) in 5 dB HTL steps, hence $\mathbf{x} = [x_1, ..., x_4]^\top$ with $x_i \in \{0, 5, 10, ..., 80\}$. Note, that this setup did not account for any differences between a subject's two ears, but all subjects had a similar hearing loss on both ears. The main purpose of this study was to compare the performance of IHAPS in a four-dimensional scenario with that of the two-dimensional scenario in terms of reproducibility and convergence.

4.1 Algorithm details

The model was defined similar to the model used in Sec. 3, except that the scale of the half Student's t-distribution for the length-scale parameters, λ_i, was changed since the range of each dimension x_i was different. The model was defined as

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{x}')^\top \mathbf{P}^{-1}(\mathbf{x} - \mathbf{x}')\right), \tag{30}$$

$$\text{with } \mathbf{P} = \text{diag}([\lambda_1, ..., \lambda_d]^\top), \tag{31}$$

$$\sigma_f \sim p([\boldsymbol{\theta}_\mathcal{C}]_{d+1}) = \delta(\sigma_f = 4), \tag{32}$$

$$\lambda_i \sim p([\boldsymbol{\theta}_\mathcal{C}]_i) \ = \text{half-St}(\lambda_i; 6, 100), \tag{33}$$

$$\sigma \sim p([\boldsymbol{\theta}_\mathcal{L}]_1) = \text{half-St}(\sigma; 6, 10), \tag{34}$$

$$\nu - 2 \sim p([\boldsymbol{\theta}_\mathcal{L}]_2) = \text{half-St}(\nu - 2; 6, 10). \tag{35}$$

Again, the hyper parameters were learned by optimizing Eq. (15) using a gradient ascent method with initial values $\sigma_f = 4, \lambda_i = 30, \sigma = \exp(1), \nu = 2{+}\exp(1)$.

For the active learning part, evaluating the EI for all possible input values was computational intractable in this four-dimensional scenario. Instead, the setting, $\hat{\mathbf{x}}^*$, to constitute the next comparison was found directly by maximizing the EI with respect to the input, $\mathbf{x}_l^*$, with a BFGS gradient ascent method [27]. Five uniformly-random starts of the initial value of $\mathbf{x}_l^*$ were used for the gradient ascent method. With only five random starts, the global maximum of the EI is generally not discovered. This creates a similar effect as in Sec. 3, although it was achieved differently. Likewise, the average of the EI could not be computed in a reasonable four-dimensional grid. Instead, the average EI along the path of the gradient ascend method was used as an estimate of the true average. A single estimate can be very different from the true average. If for instance the initialization of the gradient ascent method is close to the maximum, the estimate is much larger than the true average. To remove some of this variance, a 4-block running average was used to smooth the convergence.

4.2 Procedure

The procedure was identical to the two-dimensional study described in Sec. 3.2, except that the baseline setting was directly the measured audiogram.

Due to practical circumstances not all ten subjects from the first study were able to participate in the second study. Therefore, four new test subjects participated. Only subject 6 was deliberately not considered for the second study, since she was clearly unable to distinguish between different settings in the first study. It was considered not to include subject 7 either, due to the results in the first study. However, apparently subject 7 did not have difficulties distinguishing settings, but was only in doubt of what he preferred. Hence as such, subject 7 constitutes an interesting case.

4.3 Results

The found best settings in the two consecutive tests and the significance-test results are shown in Fig 10. The convergence is shown in Fig. 11. Generally, nine of ten subjects obtained a setting that was significantly preferred over the baseline setting given by the user's audiogram. Subject 7 neither preferred the obtained setting nor the baseline setting significantly. The setting resulting from Test 1 - instead of Test 2 - was used for the significance tests for subject 11 and 13. The reason is that the two obtained settings were found to be very different from each other. Furthermore, the two subjects reported, on their own initiative, that the settings presented to them in the second test were in general noticeably worse than the settings from the first test —even at the end of the session, where at least one of the settings should have been good.

From Fig. 11, two runs—test 1 with subject 7 and test 2 with subject 11—are seen not to have converged by the 30th iteration. Overall, the estimation of average EI is more noisy, which makes it more unclear if particular test converged.

In Fig. 12, the long-term power spectra of the SPL at the eardrum generated by the (left) HA are shown for the three different settings (Test 1, Test 2, Audiogram) for five subjects (see sub-figure caption). The measurements were made on a KEMAR through a GRAS IEC711 coupler.

4.4 Discussion

Generally, it is satisfying that the only subject (subject 7), that did not have a significant preference for the setting obtained with IHAPS, actually obtained two settings which are almost identical to the baseline—both as regards the parameters (see Fig. 10) and the output (see Fig. 12c). Remember, that this is the subject that could not decide what type of sound he preferred in the first study (Sec. 3). Before the experiment, this subject actually remembered that he was not able to decide between two types of HA sound in the first experiment, and ensured that he would not behave similarly in the second experiment. This bias is a plausible explanation of why this subject suddenly was able to obtain a similar HA sound in the two tests.

The reproducibility of the found settings is not perfect. However, the processing in HAs for very different target-gains is not necessarily very different; it depends entirely on the fitting rationale. This is the case for the two settings

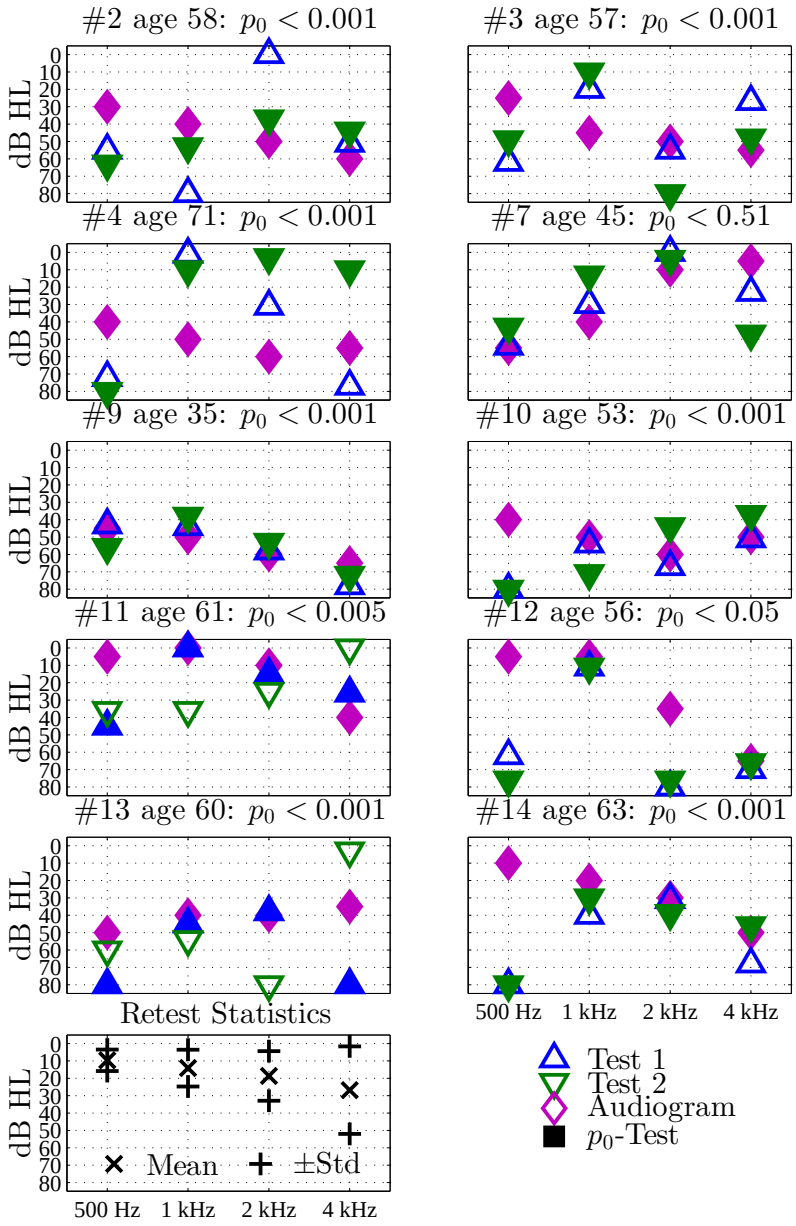


Figure 10: Optimal target-gains settings found in Test 1 (△) and Test 2 (▽) together with the measured audiogram (◇). Filled markers indicate the settings used in the significance tests (■). The bottom left plot shows the mean (×) and standard deviation (+) of the found parameter difference between test 2 and test 1.

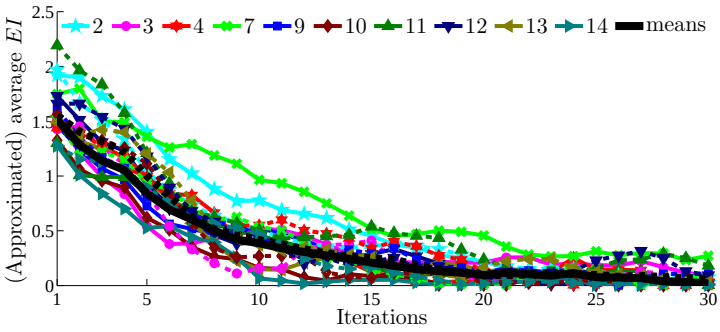
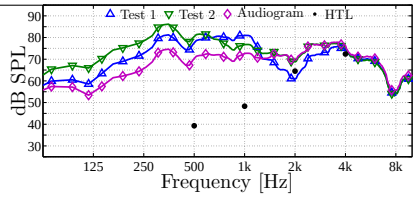


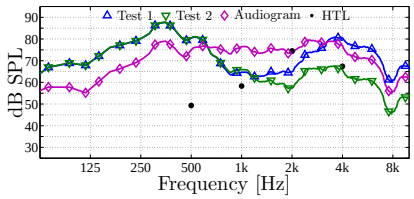
Figure 11: Estimated convergence measure for each subject and the mean convergence over subjects using only the tests that converged (i.e., excluding Test 1 with subject 7 and Test 2 with subject 11).

obtained for subject 2 and 3. To realize this, compare the actual HA output in Fig. 12a with the difference in the obtained settings for subject 2 in Fig. 10. The settings at 1 kHz suggests a gain difference of nearly 30 dB between Test 1 and Test 2. The difference, however, results in less than 5 dB (long-term) SPL difference at 1 kHz. Similarly, at 2 kHz the difference in the obtained setting is around 35 dB, but results in around 7-9 dB difference in the output at 2 kHz. Other effects also occur between the two settings due to the presence of dynamic compression in the HAs, but the long-term power spectra show that the HA outputs for the two target gains obtained for subject 2 were not too different after all. Nevertheless, at least the HA-output difference of 7-9 dB around 2 kHz must have been perceptually distinguishable. The reason why IHAPS apparently failed to obtain a similar parameter setting at 2 kHz in the two tests for subject 2, may be because a parameter change at 2 kHz given the other parameters results only in subtle changes of the HA output around the hearing threshold level (HTL) of subject 2 as seen in Fig. 12a. This demonstrate that internal dependencies among parameters in the HAs obviously need to be included in the analysis of the reproducibility. Some subjects apparently had an IRF with large regions with nearly identical responses as a results of their HTL in combination with the HA processing for different parameter settings. This was actually observed in the two dimensional case for several subjects including the example shown in Fig. 8. Furthermore, one might speculate that the HTL and the HA processing might have imposed multiple optima of the IRF with equal responses, such that the corresponding settings would have equally been preferred. With all the above in mind, it is fair to conclude that the reproducibility is acceptable overall, with subjects 4, 11 and 13 being the exceptions.

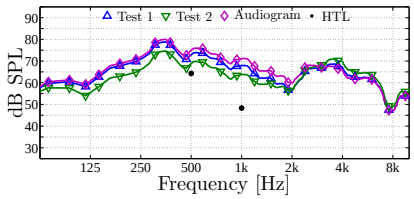
An interesting effect is observed from the bottom left figure of Fig. 10. The mean and standard deviation of the difference in the obtained optimal settings between test 1 and test 2 show a linear increasing tendency as a function of frequency. Three possible explanations (and likely a combination) are: First, after the first test the majority of subjects explained that they primarily preferred a full-bodied and soft sound as opposed to a thin and metallic sound. This indicates that at least in the beginning - i.e., a large part of test 1 - subjects



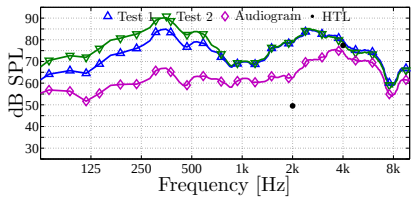
(a) Subject 2



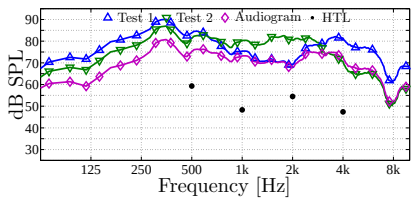
(b) Subject 4



(c) Subject 7



(d) Subject 12



(e) Subject 13

Figure 12: Power spectra of the SPL at the eardrum generated by the (left) HA with the obtained optimal setting from Test 1 (△) and Test 2 (▽) together with the audiogram (◊) for (a) subject 2, (b) subject 4, (c) subject 7, (d) subject 12, and (e) subject 13. The HTL of each subject at the four basis frequencies—500 Hz, 1 kHz, 2 kHz and 4 kHz—are indicated by black dots (if above 25 dB SPL). The A-weighted SPL at the location of the KEMAR/subject was measured to be 69.4 dB SPL. The peaks around 300 Hz are due to a Helmholtz resonance caused by a little leakage in the earplug of the KEMAR.

tended to focus on the low-mid frequency regions, but might have been less aware of the subtle details at higher frequencies. Apparently, subjects were not aware of these details until later and possibly not until the second test. This suggests a training effect. Secondly, several subjects appeared to get tired and thus distracted during the second test, whereby they might not have noticed the subtle details at higher frequencies. From the results, the latter seems to be the case for subject 11 and 13. A third and indeed possible explanation is that the majority of the subjects had high-frequency sloping hearing losses. As a consequence, the SPL at higher-frequency was below the HTL for the majority of the settings. As a result, IHAPS might have learned that the high-frequency parameters had no influence on the IRF. IHAPS should eventually be able to learn that actually a limited range of these parameters (the settings above the HTL) imposes a perceptual differences. However, this requires that the active learning criterion queries settings within this limited range. This is not its main priority in the beginning with no assessments indicating that these parameters are important. The effect may explain the output difference around 2 kHz for subject 2 (see Fig. 12a). Obviously, this emphasizes the importance of restricting the parameter range of the HA devices to a reasonable range, where settings are perceptually different.

5 Conclusion and Future Directions

An interactive hearing-aid personalization system (IHAPS) based on a flexible non-parametric Gaussian process model and on an efficient sequential design is proposed. For ten HI subjects it was demonstrated that the system obtained a successful individual setting of a set of HAs controlled by either two or four parameters within ten to twenty user assessment—equivalent to a 5-10 min. session length. The subjects significantly preferred their individual setting provided that they could distinguish between the different settings. An obvious pitfall occurs if no perceptual difference exists for a large range of settings. Furthermore, listener fatigue and training effects appeared to noticeably influence the consistency of subjects and should be investigated more systematically.

In time, IHAPS may potentially be applicable in clinics to help both the hearing-care professional and the client to fine-tune hearing aids more efficiently and precisely to the client's preferences. To get there, the reproducibility of an individual setting should be further studied. Furthermore, the stimulus (music) was kept constant during the experiments; hence, the obtained settings may not generalize to other similar stimuli (music pieces). In a more realistic scenario, the stimulus used in each assessments could be randomly chosen from a library of music, speech and other sound types. Thereby, additional uncertainty is introduced, but an individual setting obtained with IHAPS in this manner has a better chance to generalize to for instance the music context in general.

Acknowledgment

This work was conducted in collaboration between the Technical University of Denmark and Widex A/S through the industrial PhD program (case number 10-096365) and supported in part by The Danish Agency for Science, Technology,

and Innovation.

References

[1] National Acoustic Laboratories, "NAL-NL2," http://www.nal.gov.au/nal-software_tab_nal-nl-2.shtml, accessed: 28-07-2014.

[2] National Centre for Audiology, "DSL v5.0a," http://www.dslio.com/page/en/pubs_downloads.html, accessed: 28-07-2014.

[3] William A. Yost, *Fundamentals of Hearing - An Introduction*, Academic Press, 3rd edition, 1994.

[4] Brian C. J. Moore, *Cochlear Hearing Loss*, John Wiley & Sons, Ltd, 2nd edition, 2007.

[5] H. Dillon, *Hearing Aids*, Boomerang Press, 2nd edition, 2012.

[6] E. Convery, G. Keidser, H. Dillon, and L Hartley, "A self-fitting hearing aid: Need and concept," *Trends in Amplification*, vol. 15, no. (4), pp. 157–166, 2011.

[7] H. Dillon, J.A. Zakis, H McDermott, G. Keidser, W. Dreschler, and E. Convery, "The trainable hearing aid: What will it do for clients and clinicians?," *The Hearing Journal*, vol. 59, no. 4, 2006.

[8] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, 2006.

[9] B. S. Jensen, J. B. Nielsen, and J. Larsen, "Efficient Preference Learning with Pairwise Continuous Observations and Gaussian Processes," *IEEE Workshop MLSP, Beijing*, September 2011.

[10] J. B. Nielsen, B. S. Jensen, J. Nielsen, and J. Larsen, "Hearing Aid Personalization," in *NIPS 2013 Workshop on Personalization*, Lake Tahoe, USA, 2013, pp. 2–5.

[11] J. B. Nielsen and Jakob Nielsen, "Efficient individualization of hearing aid processed sound," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013.

[12] F. K. Kuk and N. M. C. Pape, "The reliability of a modified simplex procedure in hearing aid frequency-response selection," *J Speech Hear Res*, vol. 35, no. 2, pp. 418–429, 1992.

[13] D. Baskent, C. L. Eiler, and B. Edwards, "Using genetic algorithms with subjective input from human subjects: implications for fitting hearing aids and cochlear implants.," *Ear and hearing*, vol. 28, no. 3, pp. 370–80, June 2007.

[14] E. A. Durant, G. H. Wakefield, D. J. Van Tasell, and M. E. Rickert, "Efficient perceptual tuning of hearing aids with genetic algorithms," *Speech and Audio Processing, IEEE Transactions on*, vol. 12, no. 2, pp. 144–155, 2004.

[15] T. Heskes and B. de Vries, "Incremental utility elicitation for adaptive personalization," in *Proceedings of the 17th Belgium-Netherlands Conference on Artificial Intelligence, Brussels*. 2005, pp. 127–134, Citeseer.

[16] L. L. Thurstone, "A Law of Comparative Judgement," *Psychological Review*, vol. 34, 1927.

[17] R. D. Bock and J. V. Jones, *The Measurement and Prediction of Judgment and Choice*, Holden-day, 1968.

[18] W. Chu and Z. Ghahramani, "Preference Learning with Gaussian Processes," *Proceedings of the 22nd International Conference on Machine Learning (ICML)*, pp. 137–144, 2005.

[19] Adriana Birlutiu, Perry Groot, and Tom Heskes, "Multi-task preference learning with Gaussian processes," in *Proceedings of the 17th European Symposium on Artificial Neural Networks (ESANN)*, 2009, pp. 123–128.

[20] B. S. Jensen and J. B. Nielsen, "Pairwise Judgements and Absolute Ratings with Gaussian Process Priors," Tech. Rep., DTU, IMM, November 2011. http://www2.compute.dtu.dk/pubdb/p.php?6151.

[21] N. Houlsby, F. Huszár, Z. Ghahramani, and M. Lengyel, "Bayesian Active Learning for Classification and Preference Learning," *ArXiv e-prints*, Dec. 2011.

[22] D. R. Jones, "A Taxonomy of Global Optimization Methods Based on Response Surfaces," *Journal of Global Optimization*, vol. 21, no. 4, pp. 345–383, 2001.

[23] A. Gelman, "Prior distributions for variance parameters in hierarchical models," *Bayesian Analysis*, vol. 1, no. 3, pp. 515–533, 2006.

[24] J. Vanhatalo and A. Vehtari, "Sparse log gaussian processes via mcmc for spatial epidemiology," in *JMLR Workshop and Conference Proceedings*, 2007, vol. 1, pp. 73–89.

[25] J. Vanhatalo and A. Vehtari, "Speeding up the binary Gaussian process classification," in *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*, 2010.

[26] J. B. Nielsen, "Supplementary material - perception based personalization of hearing aid using gaussian processes and active learning," Jan 2014. http://www2.compute.dtu.dk/pubdb/p.php?6726.

[27] Hans Bruun Nielsen, "A matlab toolbox for optimization and data fitting," 2010. http://www2.compute.dtu.dk/~hbni/immoptibox/Manual.pdf.

Appendix I

Mathematical Derivations

I.1 Gradient descent for bEI

Since the bEI is analytical tractable, it is possible to maximize it using gradient ascent. Firstly, differentiation of the bEI with respect to $\mathbf{x}_l^*$ is performed in I.1.1. Secondly, the required terms of the GP predictive distribution are differentiated trough by $\mathbf{x}_l^*$ in I.1.2.

I.1.1 Gradient of EI

The gradient wrt. the query $\mathbf{x}_l^*$ is given by

$$\frac{\partial EI(\mathbf{x}_l^*)}{\partial \mathbf{x}_l^*} = \frac{\partial \sigma_{I_j}}{\partial \mathbf{x}_l^*} \mathcal{N}\left(\frac{\mu_{I_j}}{\sigma_{I_j}}\right) + \sigma_{I_j} \frac{\partial \mathcal{N}\left(\frac{\mu_{I_j}}{\sigma_{I_j}}\right)}{\partial \mathbf{x}_l^*} + \frac{\partial \mu_{I_j}}{\partial \mathbf{x}_l^*} \phi\left(\frac{\mu_{I_j}}{\sigma_{I_j}}\right) + \mu_{I_j} \frac{\partial \phi\left(\frac{\mu_{I_j}}{\sigma_{I_j}}\right)}{\partial \mathbf{x}_l^*} \tag{I.1}$$

$$= \frac{\partial \sigma_{I_j}}{\partial \mathbf{x}_l^*} \mathcal{N}\left(\frac{\mu_{I_j}}{\sigma_{I_j}}\right) - \mu_{I_j} \mathcal{N}\left(\frac{\mu_{I_j}}{\sigma_{I_j}}\right) \frac{\partial \frac{\mu_{I_j}}{\sigma_{I_j}}}{\partial \mathbf{x}_l^*} \tag{I.2}$$

$$+ \frac{\partial \mu_{I_j}}{\partial \mathbf{x}_l^*} \phi\left(\frac{\mu_{I_j}}{\sigma_{I_j}}\right) + \mu_{I_j} \mathcal{N}\left(\frac{\mu_{I_j}}{\sigma_{I_j}}\right) \frac{\partial \frac{\mu_{I_j}}{\sigma_{I_j}}}{\partial \mathbf{x}_l^*} \tag{I.3}$$

$$= \frac{\partial \sigma_{I_j}}{\partial \mathbf{x}_l^*} \mathcal{N}\left(\frac{\mu_{I_j}}{\sigma_{I_j}}\right) + \frac{\partial \mu_{I_j}}{\partial \mathbf{x}_l^*} \phi\left(\frac{\mu_{I_j}}{\sigma_{I_j}}\right) \tag{I.4}$$

Differentiating Eq. 3.4 and Eq. 3.5 yield

$$\frac{\partial \mu_{I_j}}{\partial \mathbf{x}_l^*} = \frac{\partial \left\{\mu_l^* - \mu_{\hat{i}}^*\right\}}{\partial \mathbf{x}_l^*} = \frac{\partial \mu_l^*}{\partial \mathbf{x}_l^*} \tag{I.5}$$

$$\frac{\partial \sigma_{I_j}}{\partial \mathbf{x}_l^*} = \frac{\partial \sqrt{\Sigma_{\hat{i},\hat{i}}^* + \Sigma_{\hat{i},\hat{i}}^* - 2\Sigma_{\hat{i},l}^*}}{\partial \mathbf{x}_l^*} \tag{I.6}$$

$$= \frac{1}{2\sqrt{\Sigma_{\hat{i},\hat{i}}^* + \Sigma_{l,l}^* - 2\Sigma_{\hat{i},l}^*}} \left(\frac{\partial \Sigma_{l,l}^*}{\partial \mathbf{x}_l^*} - 2\frac{\partial \Sigma_{\hat{i},l}^*}{\partial \mathbf{x}_l^*}\right) \tag{I.7}$$

$$= \frac{\frac{1}{2}\frac{\partial \Sigma_{l,l}^*}{\partial \mathbf{x}_l^*} - \frac{\partial \Sigma_{\hat{i},l}^*}{\partial \mathbf{x}_l^*}}{\sqrt{\Sigma_{\hat{i},\hat{i}}^* + \Sigma_{l,l}^* - 2\Sigma_{\hat{i},l}^*}} \tag{I.8}$$

$$\tag{I.9}$$

I.1.2 Gradient of the GP

Gradients of the relevant terms in the predictive distribution of the GP wrt. the query, $\mathbf{x}_l^*$, are

$$\begin{aligned}
\frac{\partial \mu_l^*}{\partial \mathbf{x}_l^*} &= \frac{\partial}{\partial \mathbf{x}_l^*}\left[k(\mathcal{X}, \mathbf{x}_l^*)^\top \mathbf{B}\right] \\
&= \frac{\partial k(\mathcal{X}, \mathbf{x}_l^*)^\top}{\partial \mathbf{x}_l^*}\mathbf{B} && \text{(I.10)} \\
\frac{\partial \Sigma_{l,l}^*}{[\mathbf{x}_l^*]_d} &= \frac{\partial}{[\mathbf{x}_l^*]_d}\left[k(\mathbf{x}_l^*, \mathbf{x}_l^*) - k(\mathcal{X}, \mathbf{x}_l^*)^\top \mathbf{A} k(\mathcal{X}, \mathbf{x}_l^*)\right] \\
&= \frac{\partial k(\mathbf{x}_l^*, \mathbf{x}_l^*)}{[\mathbf{x}_j]_d} - \operatorname{Tr}\left(\left[(\mathbf{A} + \mathbf{A}^\top) k(\mathcal{X}, \mathbf{x}_l^*)\right]^\top \frac{\partial k(\mathcal{X}, \mathbf{x}_l^*)}{[\mathbf{x}_j]_d}\right) \\
&= \frac{\partial k(\mathbf{x}_l^*, \mathbf{x}_l^*)}{[\mathbf{x}_j]_d} - 2k(\mathcal{X}, \mathbf{x}_l^*)^\top \mathbf{A} \frac{\partial k(\mathcal{X}, \mathbf{x}_l^*)}{[\mathbf{x}_j]_d} \Rightarrow \\
\frac{\partial \Sigma_{l,l}^*}{\mathbf{x}_l^*} &= \frac{\partial k(\mathbf{x}_l^*, \mathbf{x}_l^*)}{\mathbf{x}_l^*} - \left(2k(\mathcal{X}, \mathbf{x}_l^*)^\top \mathbf{A} \frac{\partial k(\mathcal{X}, \mathbf{x}_l^*)}{\mathbf{x}_j^\top}\right)^\top \\
&= \frac{\partial k(\mathbf{x}_l^*, \mathbf{x}_l^*)}{\mathbf{x}_l^*} - 2\frac{\partial k(\mathcal{X}, \mathbf{x}_l^*)^\top}{\mathbf{x}_l} \mathbf{A} k(\mathcal{X}, \mathbf{x}_l^*) && \text{(I.11)} \\
\frac{\partial \Sigma_{i,l}^*}{\partial \mathbf{x}_l^*} &= \frac{\partial}{\partial \mathbf{x}_l^*}\left[k(\mathbf{x}_l^*, \mathbf{x}_{\hat{i}}) - k(\mathcal{X}, \mathbf{x}_l^*)^\top \mathbf{A} k(\mathcal{X}, \mathbf{x}_{\hat{i}})\right] \\
&= \frac{\partial k(\mathbf{x}_l^*, \mathbf{x}_{\hat{i}})}{\partial \mathbf{x}_l^*} - \frac{\partial k(\mathcal{X}, \mathbf{x}_l^*)^\top}{\partial \mathbf{x}_l^*} \mathbf{A} k(\mathcal{X}, \mathbf{x}_{\hat{i}}), && \text{(I.12)}
\end{aligned}$$

where $\mathbf{A}$ and $\mathbf{B}$ are matrices in the predictive distribution of $\mathbf{f}^*$, that depend only on the training data—either directly or through the approximate inference method—and thus, do not depend on the new query point $\mathbf{x}_l^*$, but are different for different kinds of GPs. For standard GP regression $\mathbf{A} = \left(\mathbf{K} + \sigma^2\mathbf{I}\right)^{-1}$ and $\mathbf{B} = \left(\mathbf{K} + \sigma^2\mathbf{I}\right)^{-1}\mathbf{y}$. Note also, that for stationary kernels $\frac{\partial k(\mathbf{x}_l^*, \mathbf{x}_l^*)}{\mathbf{x}_l^*} = 0$. The notation $k(\mathcal{X}, \mathbf{x})$ (or $k(\mathbf{x}, \mathcal{X})$) denotes a column (or row) vector of kernel evaluations between the set of training examples $\mathcal{X}$ and the input $\mathbf{x}$.

Bibliography

Peter Auer. Using Confidence Bounds for Exploitation-Exploration Trade-offs. *Journal of Machine Learning Research*, 3:397–422, 2002. URL http://jmlr.org/papers/volume3/auer02a/auer02a.pdf.

Bart Bakker and Tom Heskes. Task Clustering and Gating for Bayesian Multitask Learning. *Journal of Machine Learning Research*, 4(1):83–99, January 2004. ISSN 1532-4435. doi: 10.1162/153244304322765658. URL http://www.crossref.org/jmlr_DOI.html.

Deniz Baskent, Cheryl L Eiler, and Brent Edwards. Using genetic algorithms with subjective input from human subjects: implications for fitting hearing aids and cochlear implants. *Ear and hearing*, 28(3):370–80, June 2007. ISSN 0196-0202. doi: 10.1097/AUD.0b013e318047935e. URL http://www.ncbi.nlm.nih.gov/pubmed/17485986.

Sø ren Bech and Nick Zacharov. *Perceptual Audio Evaluation - Theory, Method and Application*. John Wiley \& Sons, Ltd, 2007. ISBN 0470869240. URL http://books.google.dk/books/about/Perceptual_Audio_Evaluation_Theory_Metho.html?id=1WGPJai1gX8C&pgis=1.

Adriana Birlutiu, Perry Groot, and Tom Heskes. Multi-task preference learning with Gaussian processes. In *Proceedings of the 17th European Symposium on Artificial Neural Networks (ESANN)*, pages 123–128, 2009. URL http://www.cs.ru.nl/~perry/publications/2009/ESANN/birlutiu-esann.pdf.

Adriana Birlutiu, Perry Groot, and Tom Heskes. Multi-task preference learning with an application to hearing aid personalization. *Neurocomputing*, 73(7-9):1177–1185, March 2010. ISSN 09252312. doi: 10.1016/j.neucom.2009.11.025. URL http://linkinghub.elsevier.com/retrieve/pii/S0925231210000251.

Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 8th editio edition, 2006. ISBN 0-387-31073-8.

R. D. Bock and J. V. Jones. *The measurement and prediction of judgment and choice*. Holden-day, 1968. URL http://psycnet.apa.org/psycinfo/1968-35024-000.

Edwin V. Bonilla, F.V. Agakov, and Christopher K. I. Williams. Kernel Multi-task Learning using Task-specific Features. In *Proceedings of Artificial Intelligence and Statistics (AISTATS)*, volume 11. Citeseer, 2007. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.100.2081&rep=rep1&type=pdf.

Edwin V. Bonilla, K.M. Chai, and Christopher K. I. Williams. Multi-task Gaussian process prediction. *Advances in Neural Information Processing Systems*, 20:153–160, 2008. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.143.4356&rep=rep1&type=pdf.

Edwin V. Bonilla, Shengbo Guo, and Scott Sanner. Gaussian Process Preference Elicitation. *Advances in Neural Information Processing Systems*, 23:262–270, 2010.

ITU-R BS.1534-1. Method for the subjective assessment of intermediate quality level of coding systems. *The ITU Radiocommunication Assembly*, pages 1–18, 2003. URL http://www.itu.int/dms_pubrec/itu-r/rec/bs/R-REC-BS.1534-1-200301-I!!PDF-E.pdf.

Olivier Chapelle and Lihong Li. An Empirical Evaluation of Thompson Sampling. *Advances in Neural Information Processing Systems*, 24: 2249–2257, 2011. URL http://books.nips.cc/papers/files/nips24/NIPS2011_1232.pdf.

Wei Chu and Zoubin Ghahramani. Preference learning with Gaussian processes. *Proceedings of the 22nd international conference on Machine learning - ICML '05*, pages 137–144, 2005a. doi: 10.1145/1102351.1102369. URL http://portal.acm.org/citation.cfm?doid=1102351.1102369.

Wei Chu and Zoubin Ghahramani. Extensions of gaussian processes for ranking: semi-supervised and active learning. In *NIPS workshop on Learning to Rank*, pages 29–34. Citeseer, 2005b. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.85.7197&rep=rep1&type=pdf#page=33.

Elizabeth Convery, Gitte Keidser, Harvey Dillon, and Lisa Hartley. A self-fitting hearing aid: need and concept. *Trends in amplification*, 15(4):157–66, December 2011. ISSN 1940-5588. doi: 10.1177/1084713811427707. URL http://www.ncbi.nlm.nih.gov/pubmed/22143873.

Harvey Dillon. *Hearing Aids*. Boomerang Press, Turramurra, Australia, 2nd edition, 2012. ISBN 9780957816817. URL http://www.amazon.com/Hearing-Aids-Harvey-Dillon/dp/1604068108.

Harvey Dillon, Justin A. Zakis, Hugh J. McDermott, Gitte Keidser, Wouter Dreschler, and Elizabeth Convery. The trainable hearing aid: What will it do for clients and clinicians? *The Hearing Journal*, 59(4):30, 2006. doi: 10.1097/01.HJ.0000286694.20964.4a. URL http://journals.lww.com/thehearingjournal/Abstract/2006/04000/The_trainable_hearing_aid__What_will_it_do_for.5.aspx.

Eric A. Durant, Gregory H. Wakefield, Dianna J. VanTasell, Martin E. Rickert, and D. J. Van Tasell. Efficient Perceptual Tuning of Hearing Aids With Genetic Algorithms. *IEEE Transactions on Speech and Audio Processing*, 12(2):144–155, March 2004. ISSN 1063-6676. doi: 10.1109/TSA.2003.822640. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1284342http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1284342.

David Duvenaud, Hannes Nickisch, and Carl Edward Rasmussen. Additive Gaussian Processes. *Advances in Neural Information Processing Systems*, 24: 226–234, 2011.

David Duvenaud, James Robert Lloyd, Roger Grosse, Joshua B. Tenenbaum, and Zoubin Ghahramani. Structure Discovery in Nonparametric Regression through Compositional Kernel Search. In *Proceedings of the 30th International Conference on Machine Learning (ICML)*, 2013. URL http://jmlr.org/proceedings/papers/v28/duvenaud13.pdf.

Silvia Ferrari and Francisco Cribari-Neto. Beta Regression for Modelling Rates and Proportions. *Journal of Applied Statistics*, 31(7):799–815, August 2004. ISSN 0266-4763.

Johannes Fürnkranz. Preference Learning: An Introduction. In Eyke Hüllermeier, editor, *Preference Learning*, page 1. Springer, Berlin, Heidelberg, 1st edition, 2010. ISBN 978-3-642-14124-9. doi: 10.1007/978-3-642-14125-6. URL http://books.google.com/books?hl=en&lr=&id=nc3XcH9XSgYC&oi=fnd&pg=PR5&dq=Preference+Learning&ots=v1zNmw_zAe&sig=D8RvzCQVaIx4ODMCZbjRtV2kLQk.

Perry Groot, Adriana Birlutiu, and Tom Heskes. Bayesian Monte Carlo for the Global Optimization of Expensive Functions. In *Frontiers in Artificial Intelligence and Applications*, volume 215, pages 249–154, 2010. URL http://www.cs.ru.nl/~adrianab/groot-ecai2010.pdf.

Perry Groot, Tom Heskes, Tjeerd Dijkstra, and James M Kates. Predicting Preference Judgments of Individual Normal and Hearing-Impaired Listeners

With Gaussian Processes. *IEEE Transactions on Audio, Speech & Language Processing*, 19(4):811–821, 2011. ISSN 1558-7916. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5545403.

Tom Heskes and Bert de Vries. Incremental utility elicitation for adaptive personalization. In *Proceedings of the 17th Belgium-Netherlands Conference on Artificial Intelligence, Brussels, Koninklijke Vlaamse Academie van Belgi\\"e voor Wetenschappen en Kunsten*, pages 127–134. Citeseer, 2005. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.104.8332&rep=rep1&type=pdf.

Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máte Lengyel. Bayesian Active learning for Classification and Preference Learning. *arXiv*, pages 1–17, 2011. URL http://arxiv.org/abs/1112.5745.

Neil Houlsby, Jose Miguel Hernandez-Lobato, Ferenc Huszár, and Zoubin Ghahramani. Collaborative Gaussian Processes for Preference Learning. *Advances in Neural Information Processing Systems*, 25:2105–2113, 2012. URL http://books.nips.cc/papers/files/nips25/NIPS2012_1031.pdf.

Ferenc Huszár. A GP Classification Approach to Preference Learning. In *NIPS workshop on Choice Models and Preference Learning*, pages 1–4, 2011.

Tony Jebara, Risi Kondor, and Andrew Howard. Probability Product Kernels. *Journal of Machine Learning Research*, 5:819–844, 2004.

Bjørn Sand Jensen and Jens Brehm Nielsen. Pairwise Judgements and Absolute Ratings with Gaussian Process Priors. Technical report, DTU Compute, Kgs. Lyngby, 2011. URL http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=6151.

Norman L. Johnson and Samual Kotz. *Continuous Univariate Distribution - 1*. Hougton Mifflin Company, Boston, 1st edition, 1970a.

Norman L. Johnson and Samual Kotz. *Continuous Univariate Distribution - 2*. Hougton Mifflin Company, Boston, 1st edition, 1970b.

D.R. Jones. A taxonomy of global optimization methods based on response surfaces. *Journal of Global Optimization*, 21(4):345–383, 2001. ISSN 0925-5001. URL http://www.springerlink.com/index/KG7634766H12880H.pdf.

Sergei Kochkin, Douglas L. Beck, Laurel A. Christensen, Cynthia Compton-Conley, Brian J. Fligor, Patricia B. Kricos, Jay B. Mcspaden, H. Gustav Mueller, Michael J. Nilsson, Jerry L. Northern, Thomas A. Powers, Robert W. Sweetow, Brian Taylor, and Robert G. Turner. MarkeTrak VIII : The Impact of the Hearing Healthcare Professional on the Hearing Aid User Success. *The Hearing Review*, 17(4):12–34, 2010. URL http://entsolutions.us.com/webdocuments/professional-impact-on-success-ha.pdf.

Francis K. Kuk and Nonalee M. C. Pape. The reliability of a modified simplex procedure in hearing aid frequency-response selection. *Journal of Speech and Hearing Research*, 35:418–429, 1992.

Neil Lawrence, Matthias Seeger, and Ralf Herbrich. Fast sparse Gaussian process methods: The informative vector machine. In *Neural Information Processing Systems (NIPS)*, page 8, 2002. URL http://books.nips.cc/papers/files/nips15/AA16.pdf.

Miguel Lázaro-Gredilla. Bayesian Warped Gaussian Processes. *Advances in Neural Information Processing Systems*, 25:1628–1636, 2012. URL http://eprints.pascal-network.org/archive/00009880/.

Miguel Lazaro-Gredilla and Anibal Figueiras-Vidal. Inter-domain Gaussian Processes for Sparse Inference using Inducing Features. *Advances in Neural Information Processing Systems*, 22:1087–1095, 2009. URL http://books.nips.cc/papers/files/nips22/NIPS2009_0537.pdf.

Gregory R. Lockhead. Absolute Judgments Are Relative: A Reinterpretation of Some Psychophysical Ideas. *Review of General Psychology*, 8(4):265–272, 2004. ISSN 1939-1552. doi: 10.1037/1089-2680.8.4.265. URL http://doi.apa.org/getdoi.cfm?doi=10.1037/1089-2680.8.4.265.

David J. C. Mackay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 7.2 edition, 2003. ISBN 0521642981. URL http://www.inference.phy.cam.ac.uk/itprnn/book.pdf.

Thomas P. Minka. Expectation Propagation for Approximate Bayesian Inference. In *Proceedings of the 17th Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 362–369, 2001. URL http://www.mendeley.com/research/no-title-avail/http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.93.4345&rep=rep1&type=pdfhttp://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.86.1319&rep=rep1&type=pdf.

Arlene C. Neuman, Harry Levitt, Russell Mills, and Teresa Schwander. An evaluation of three adaptive hearing aid selection strategies. *The Journal of the Acoustical Society of America*, 82(6):1967–76, December 1987. ISSN 0001-4966. URL http://www.ncbi.nlm.nih.gov/pubmed/3429734.

Bryan Pardo, David Little, and Darren Gergle. Building a personalized audio equalizer interface with transfer learning and active learning. *Proceedings of the second international ACM workshop on Music information retrieval with user-centered and multimodal strategies - MIRUM '12*, page 13, 2012. doi: 10.1145/2390848.2390852. URL http://dl.acm.org/citation.cfm?doid=2390848.2390852.

Kaare Brandt Petersen and Michael Syskind Pedersen. The Matrix Cookbook. Technical report, Technical University of Denmark, Kgs. Lyngby, 2008. URL http://matrixcookbook.com.

Joaquin Quiñonero Candela and Carl Edward Rasmussen. A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research*, 6:1939–1959, 2005. URL http://portal.acm.org/citation.cfm?id=1194909.

Carl E. Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006. URL http://www.gaussianprocess.org/gpml/chapters/RW.pdf.

Carl Edward Rasmussen and Hannes Nickisch. Gaussian Processes for Machine Learning (GPML) Toolbox. *Journal of Machine Learning Research*, 11: 3011–3015, 2010. URL http://jmlr.org/papers/volume11/rasmussen10a/rasmussen10a.pdf.

D Reed. Capturing perceptual expertise: a sound equalization expert system. *Knowledge-Based Systems*, 14(1-2):111–118, March 2001. ISSN 09507051. doi: 10.1016/S0950-7051(00)00098-8. URL http://linkinghub.elsevier.com/retrieve/pii/S0950705100000988.

Andrew Sabin and Bryan Pardo. Rapid Learning of Subjective Preference in Equalization. In *Audio Engineering Society Convention*, volume 125, 2008.

Anton Schwaighofer, Volker Tresp, and Kai Yu. Learning Gaussian process kernels via hierarchical Bayes. *Advances in Neural Information Processing Systems*, 17, 2005. URL http://www.tresp.org/papers/Learn_Gaussian_priors_2004_final1.pdf.

Michael Smithson and Jay Verkuilen. A Better Lemon Squeezer? Maximum-Likelihood Regression with Beta-Distributed Dependent Variables. *Psychological Methods*, 11(1):54–71, March 2006. ISSN 1082-989X. doi: 10.1037/1082-989X.11.1.54. URL http://www.ncbi.nlm.nih.gov/pubmed/16594767.

Edward Snelson and Zoubin Ghahramani. Sparse Gaussian Processes using Pseudo-inputs. *Advances in Neural Information Processing Systems*, 18:1257–1264, 2006. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.60.2209&rep=rep1&type=pdf.

Edward Snelson and Zoubin Ghahramani. Local and global sparse gaussian process approximations. In *Proceedings of Artificial Intelligence and Statistics (AISTATS)*, volume 11. Citeseer, 2007. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.62.2676&rep=rep1&type=pdfhttp://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.62.2676&rep=rep1&type=pdf.

Edward Snelson, Carl E. Rasmussen, and Zoubin Ghahramani. Warped Gaussian Processes. *Advances in Neural Information Processing Systems*, 16, 2004. ISSN 1049-5258. URL http://books.nips.cc/papers/files/nips16/NIPS2003_AA43.pdf.

Niranjan Srinivas, Andreas Krause, Sham Kakade, and Matthias Seeger. Gaussian Process Optimization in the Bandit Setting: No Regret and Experimental Design. In *Proceeding of the 25th International Conference on Machine Learning (ICML)*, pages 1015–1022, 2010. URL http://las.ethz.ch/files/srinivas10gaussian.pdf.

Xiaoyuan Su and Taghi M. Khoshgoftaar. A Survey of Collaborative Filtering Techniques. *Advances in Artificial Intelligence*, 2009(Section 3):1–19, 2009. ISSN 1687-7470. doi: 10.1155/2009/421425. URL http://www.hindawi.com/journals/aai/2009/421425/.

Hideyuki Takagi and Miho Ohsaki. IEC-based hearing aid fitting. In *IEEE SMC'99 Conference Proceedings. 1999 IEEE International Conference on Systems, Man, and Cybernetics*, volume 3, pages 657–662. Ieee, 1999. ISBN 0-7803-5731-0. doi: 10.1109/ICSMC.1999.823291. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=823291.

William R. Thompson. On the Likelihood that One Unknown Probability Exceeds Another in View of the Evidence of Two Samples. *Biometrika*, 25(34): 285–294, 1933.

LL Thurstone. Psychophysical analysis. *The American journal of psychology*, 38 (3):368–389, 1927. URL http://www.jstor.org/stable/10.2307/1415006.

Jarno Vanhatalo and Aki Vehtari. Modelling local and global phenomena with sparse Gaussian processes. In *Proceedings of the 24th Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 571–578, 2008. URL http://www.lce.hut.fi/~jpvanhat/publications/vanhatalo.pdf.

Francesco Vivarelli and Christopher K. I. Williams. Discovering hidden features with Gaussian processes regression. *Advances in Neural Information Processing Systems*, 11, 1999.

Christian Walder, Kwang In Kim, and Bernhard Schölkopf. Sparse Multiscale Gaussian Process Regression. In *The 25th international conference on Machine Learning*, pages 1112–1119, 2008. URL http://dl.acm.org/citation.cfm?id=1390296.

Ernst H. Weber. On the sense of touch and common sensibility. In B. Haupt (Trans.), R. J. Hernstein, and E. G. Boring, editors, *A sourcebook in the history of psychology*. Harvard University Press (Original work published 1846), Cambridge, 1965.

Christopher K. I. Williams and David Barber. Bayesian Classification With Gaussian Processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1342–1351, 1998.

Andrew Gordon Wilson and Ryan Prescott Adams. Gaussian Process Kernels for Pattern Discovery and Extrapolation. In *Proceedings of the 30th International Conference on Machine Learning (ICML)*, pages 1067–1075, 2013. URL http://jmlr.org/proceedings/papers/v28/wilson13.pdf.

Kai Yu and Wei Chu. Gaussian Process Models for Link Analysis and Transfer Learning. *Advances in Neural Information Processing Systems*, 19:1–8, 2007. URL http://machinelearning.wustl.edu/mlpapers/paper_files/NIPS2007_928.pdf.

Kai Yu, Volker Tresp, and Anton Schwaighofer. Learning Gaussian processes from Multiple Tasks. In *Proceedings of the 22nd International Conference on Machine Learning (ICML)*, volume 17, pages 1012–1019, New York, New York, USA, 2005. ACM Press. ISBN 1595931805. doi: 10.1145/1102351.1102479. URL http://www.tresp.org/papers/Learn_Gaussian_priors_2004_final1.pdf.