Olli-Pekka Koistinen

Bayesian Classification of fMRI Patterns for Natural Audiovisual Stimuli Using Sparsity Promoting Laplace Priors

School of Electrical Engineering

Thesis submitted for examination for the degree of Master of Science in Technology.

Espoo, 14th May 2012

Thesis supervisor:

Prof. Jouko Lampinen

Thesis instructors:

M.Sc. (Tech.) Pasi Jylänki

D.Sc. (Tech.) Aki Vehtari

Aalto University
School of Electrical
Engineering

AALTO UNIVERSITY
SCHOOL OF ELECTRICAL ENGINEERING

ABSTRACT OF THE MASTER'S THESIS

Author: Olli-Pekka Koistinen

Title: Bayesian Classification of fMRI Patterns for Natural Audiovisual Stimuli Using Sparsity Promoting Laplace Priors

Date: 14^{th} May 2012 Language: English Number of pages: 11+69

Department of Biomedical Engineering and Computational Science

Professorship: Computational Engineering Code: S-114

Supervisor: Prof. Jouko Lampinen

Instructors: M.Sc. (Tech.) Pasi Jylänki, D.Sc. (Tech.) Aki Vehtari

Bayesian linear binary classification models with sparsity promoting Laplace priors were applied to discriminate fMRI patterns related to natural auditory and audiovisual speech and music stimuli. The region of interest comprised the auditory cortex and some surrounding regions related to auditory processing.

Truly sparse posterior mean solutions for the classifier weights were obtained by implementing an automatic relevance determination method using expectation propagation (ARDEP). In ARDEP, the Laplace prior was decomposed into a Gaussian scale mixture, and these scales were optimised by maximising their marginal posterior density. ARDEP was also compared to two other methods, which integrated approximately over the original Laplace prior: LAEP approximated the posterior as well by expectation propagation, whereas MCMC used a Markov chain Monte Carlo simulation method implemented by Gibbs sampling.

The resulting brain maps were consistent with previous studies for simpler stimuli and suggested that the proposed model is also able to reveal additional information about activation patterns related to natural audiovisual stimuli. The predictive performance of the model was significantly above chance level for all approximate inference methods. Regardless of intensive pruning of features, ARDEP was able to describe all of the most discriminative brain regions obtained by LAEP and MCMC. However, ARDEP lost the more specific shape of the regions by representing them as one or more smaller spots, removing also some relevant features.

Keywords: audiovisual, auditory cortex, automatic relevance determination, Bayesian, classification, expectation propagation, fMRI, Laplace prior, music, speech

AALTO-YLIOPISTO
SÄHKÖTEKNIIKAN KORKEAKOULU

DIPLOMITYÖN
TIIVISTELMÄ

Tekijä: Olli-Pekka Koistinen

Työn nimi: Luonnollisiin audiovisuaalisiin ärsykkeisiin liittyvän fMRI-aktivaation bayesilainen luokittelu harvoja ratkaisuja suosivia Laplace-prioreja käyttäen

Päivämäärä: 14.5.2012 Kieli: Englanti Sivumäärä: 11+69

Lääketieteellisen tekniikan ja laskennallisen tieteen laitos

Professuuri: Laskennallinen tekniikka Koodi: S-114

Valvoja: Prof. Jouko Lampinen

Ohjaajat: DI Pasi Jylänki, TkT Aki Vehtari

Bayesilaisia lineaarisia binääriluokittelumalleja ja harvoja ratkaisuja suosivia Laplace-prioreja sovellettiin erottelemaan luonnollisiin auditorisiin ja audiovisuaalisiin puhe- ja musiikkiärsykkeisiin liittyvää fMRI-aktivaatiota kuuloaivokuorella ja sitä ympäröivillä auditoriseen prosessointiin liittyvillä alueilla.

Absoluuttisen harvoja posteriorisia odotusarvoratkaisuja luokittimien painoille saatiin expectation propagation -algoritmin avulla toteutetulla automatic relevance determination -menetelmällä (ARDEP). ARDEP-menetelmässä hyödynnettiin Laplace-priorin gaussista skaalahajotelmaa, jonka skaalaparametrit optimoitiin maksimoimalla niiden marginaalinen posterioritiheys. Menetelmää verrattiin myös kahteen muuhun menetelmään, jotka integroivat approksimatiivisesti alkuperäisen Laplace-priorin yli: LAEP approksimoi posteriorijakaumaa niin ikään expectation propagation -algoritmin avulla, kun taas MCMC käytti Gibbs-poiminnalla toteutettua Markovin ketju Monte Carlo -simulaatiomenetelmää.

Tuloksena saadut aivokartat olivat linjassa aikaisempien, yksinkertaisemmilla ärsykkeillä saatujen tutkimustulosten kanssa, ja niiden perusteella bayesilaisten luokittelumallien avulla on mahdollista saada myös uudenlaista tietoa siitä, miten luonnollisia audiovisuaalisia ärsykkeitä koodataan aivoissa. Mallien ennustuskyky oli kaikilla approksimaatiomenetelmillä merkittävästi sattumanvaraista tasoa korkeampi. Piirteiden voimakkaasta karsinnasta huolimatta ARDEP pystyi kuvaamaan kaikki huomattavimmat LAEP:n ja MCMC:n erottelemat aivoalueet. ARDEP menetti kuitenkin alueiden tarkemman muodon esittämällä ne yhtenä tai useampana pienempänä alueena, poistaen myös osan merkittävistä piirteistä.

Avainsanat: audiovisuaalinen, automatic relevance determination, bayesilainen, expectation propagation, fMRI, kuuloaivokuori, Laplace-priori, luokittelu, musiikki, puhe

Preface

This thesis sums up a laborious but occasionally rewarding year of work at the Department of Biomedical Engineering and Computational Science. I am grateful to Prof. Mikko Sams and Prof. Jouko Lampinen for the opportunity to combine mathematical challenges with the fascinating field of neuroscience. It has not always been straightforward to get the computational methods working desirably in practice, but some weird magnetic effect (maybe a desire to graduate some day) glued me to the office chair and kept me teasing my brain, just to figure out how it works during a more relaxing task.

I would like to express my deepest gratitude to my primary instructor Pasi Jylänki, who is behind most of the methodological ideas used in this work and has been of great help also in the practical implementation of the algorithms. I thank also Dr. Aki Vehtari for more general instruction and comments on the draft of the thesis and Dr. Juha Salmitaival for his help with the neuroscientifical interpretation. My special thanks are addressed to Sasu Mäkelä, who shared his scripts for data import and visualisation. I am grateful also to Enrico Glerean and all other people, who have contributed to the collection and preprocessing of the fMRI data that was used in this work, not forgetting to thank Janne Ojanen for his incisive comments on some statistical principles.

Finally, I would like to thank my fiancée Anu for patiently supporting me through this stressful year. With bad grace, I must thank also the stone that tore apart my Achilles tendon and made it possible to finish this thesis in time.

Otaniemi, 14$^{\text{nd}}$ May 2012

Olli-Pekka Koistinen

Contents

Abstract ii

Abstract (in Finnish) iii

Preface iv

Contents v

Symbols and Abbreviations vii

1 Introduction 1

2 Background 3

2.1 Cortical Auditory Processing 3

2.2 Functional Magnetic Resonance Imaging 5

2.2.1 Magnetic Resonance Imaging 5

2.2.2 Acquisition of T_2^*-weighted Contrast 6

2.2.3 Blood Oxygenation Level Dependent Brain Imaging 7

2.3 Bayesian Classification 9

2.3.1 Basics of Probability Theory 9

2.3.2 Bayesian Inference 11

2.3.3 Classification 12

3 Data 14

3.1 Experimental Design 14

3.2 Data Collection 14

3.3 Preprocessing 14

3.4 Labelling of Observations for Different Classification Settings 15

4 Model 16

4.1 Linear Binary Classifier 16

4.2 Bayesian Inference on the Weights 17

4.3 Laplace Prior Distribution 18

5 Approximate Inference 21

5.1 Automatic Relevance Determination by Expectation Propagation . . 21

5.1.1 Automatic Relevance Determination 21

5.1.2 ARDEP 22

5.1.3 Expectation Propagation 22

5.1.4 Fast Sequential Optimisation of Relevance Hyperparameters . 25

5.1.5 Practical Modifications 26

5.1.6 Alternative Criteria for Selection of the Final Relevance Hyperparameter Vector 27

5.2 Expectation Propagation on Laplace Prior 30

5.3 Markov Chain Monte Carlo 31

5.3.1 Posterior Simulation 31
5.3.2 MCMC 31
5.4 Hyperparameter Selection 34
5.4.1 Cross-validation 34
5.4.2 Alternative Considerations 36

6 Results **39**
6.1 Brain Maps for Piano vs. Speech Setting 39
6.2 Brain Maps for Auditory vs. Audiovisual Setting 42
6.3 Predictive Performance for Piano vs. Speech Setting 43
6.4 Predictive Performance for Auditory vs. Audiovisual Setting 46

7 Discussion **49**

References **53**

A Expectation Propagation for the Probit Model with ARD Prior **58**

B MAP Estimate for a Single Relevance Hyperparameter in ARDEP **66**

Symbols and Abbreviations

Variables

a_i	auxiliary scalar $\mathbf{m}_w^\top t_i \mathbf{x}_i$ (in ARDEP)
a_i^*	auxiliary scalar $(\mathbf{m}_w^*)^\top t_i \mathbf{x}_i$ (in ARDEP)
$a_i^{\setminus i}$	auxiliary scalar $(\mathbf{m}_w^{\setminus i})^\top t_i \mathbf{x}_i$ (in ARDEP)
b_i	auxiliary scalar $t_i \mathbf{x}_i^\top \mathbf{V}_w t_i \mathbf{x}_i$ (in ARDEP)
b_i^*	auxiliary scalar $t_i \mathbf{x}_i^\top \mathbf{V}_w^* t_i \mathbf{x}_i$ (in ARDEP)
$b_i^{\setminus i}$	auxiliary scalar $t_i \mathbf{x}_i^\top \mathbf{V}_w^{\setminus i} t_i \mathbf{x}_i$ (in ARDEP)
$\mathbf{c}_i$	auxiliary vector $\mathbf{V}_w t_i \mathbf{x}_i$ (in ARDEP)
$\mathbf{c}_i^*$	auxiliary vector $\mathbf{V}_w^* t_i \mathbf{x}_i$ (in ARDEP)
$\mathbf{c}_i^{\setminus i}$	auxiliary vector $\mathbf{V}_w^{\setminus i} t_i \mathbf{x}_i$ (in ARDEP)
$\mathrm{CA}_{\mathrm{double-loso}}$	double-cross-validated classification accuracy
$\mathrm{CA}_{\mathrm{loo}}$	leave-one-out classification accuracy
$\widetilde{\mathrm{CA}}_{\mathrm{loo}}$	EP estimate for $\mathrm{CA}_{\mathrm{loo}}$ (in ARDEP)
$\mathrm{CA}_{\mathrm{loso}}$	leave-one-subject-out classification accuracy
$\widetilde{\mathrm{CA}}_{\mathrm{loso}}$	EP estimate for $\mathrm{CA}_{\mathrm{loso}}$ (in ARDEP)
$\mathbf{C}_S$	auxiliary matrix $\mathbf{\Phi}_S \mathbf{V}_w$ (in ARDEP)
D	amount of features
F	set of features included in EP (in ARDEP)
h_j	auxiliary scalar $\boldsymbol{\phi}_j^\top (\mathbf{\Omega}^{\setminus j})^{-1} \boldsymbol{\rho}$ (in ARDEP)
H_j	auxiliary scalar $\boldsymbol{\phi}_j^\top \mathbf{\Omega}^{-1} \boldsymbol{\rho}$ (in ARDEP)
i	observation index
j	feature index
k	subject index
K	amount of subjects
$\mathbf{L_1}$	lower triangular matrix in Cholesky decomposition (in ARDEP)
$\mathbf{L_2}$	lower triangular matrix in Cholesky decomposition (in ARDEP)
m	feature index
$\mathbf{m}_w$	mean vector for approximate posterior distribution of $\mathbf{w}$
$\bar{\mathbf{m}}_w$	sparsified mean vector (in ARDEP)
$\mathbf{m}_w^*$	updated mean vector (in ARDEP)
$\mathbf{m}_w^{\setminus i}$	mean vector for leave-i-out approximation (in ARDEP)
$\mathbf{m}_w^{\setminus S}$	mean vector for leave-S-out approximation (in ARDEP)
$\mathrm{MLPP}_{\mathrm{double-loso}}$	double-cross-validated mean log predictive probability
$\mathrm{MLPP}_{\mathrm{loo}}$	leave-one-out mean log predictive probability
$\widetilde{\mathrm{MLPP}}_{\mathrm{loo}}$	EP estimate for $\mathrm{MLPP}_{\mathrm{loo}}$ (in ARDEP)
$\mathrm{MLPP}_{\mathrm{loso}}$	leave-one-subject-out mean log predictive probability
$\widetilde{\mathrm{MLPP}}_{\mathrm{loso}}$	EP estimate for $\mathrm{MLPP}_{\mathrm{loso}}$ (in ARDEP)
N	amount of observations
N^*	amount of test observations
r_j	auxiliary scalar $\boldsymbol{\phi}_j^\top (\mathbf{\Omega}^{\setminus j})^{-1} \boldsymbol{\phi}_j$ (in ARDEP)
R_j	auxiliary scalar $\boldsymbol{\phi}_j^\top \mathbf{\Omega}^{-1} \boldsymbol{\phi}_j$ (in ARDEP)

s	sample index (in MCMC)
S	set of observations
S_k	set of observations for subject k
$\hat{t}_i^*$	predicted class for feature vector $\mathbf{x}_i^*$
$\mathbf{t} = (t_1, \ldots, t_N)^\mathsf{T}$	training data vector of target classes
$\mathbf{t}^* = (t_1^*, \ldots, t_N^*)^\mathsf{T}$	test data vector of correct classes
$\mathbf{t}_S$	test data vector of correct classes for observations $i \in S$
$\mathbf{t}^{\setminus S}$	training data vector of target classes for observations $i \notin S$
T_1	longitudinal relaxation time
T_2	spin-spin relaxation time
T_2^*	transverse relaxation time
TE	echo time
TR	repetition time
$\mathbf{u} = (u_1, \ldots, u_N)^\mathsf{T}$	vector of latent variables in probit model
$\mathbf{u}^s = (u_1^s, \ldots, u_N^s)^\mathsf{T}$	s^{th} sample of latent variable $\mathbf{u}$ (in MCMC)
v_j^*	maximum a posteriori estimate for v_j (in ARDEP)
$\mathbf{v} = (v_1, \ldots, v_D)^\mathsf{T}$	vector of relevance hyperparameters (in ARDEP)
$\mathbf{v} = (v_1, \ldots, v_D)^\mathsf{T}$	vector of auxiliary scale variables (in MCMC)
$\bar{\mathbf{v}}$	sparsified relevance hyperparameter vector (in ARDEP)
$\hat{v}_j$	auxiliary scalar $-\frac{\lambda^2}{2} - \frac{1}{r_j} + \sqrt{\frac{\lambda^4}{4} + \frac{\lambda^2 h_j^2}{r_j^2}}$ (in ARDEP)
$\mathbf{v}_{\text{MAP}}$	maximum a posteriori estimate for $\mathbf{v}$ (in ARDEP)
$\mathbf{v}^s = (v_1^s, \ldots, v_D^s)^\mathsf{T}$	s^{th} sample of auxiliary scale variable $\mathbf{v}$ (in MCMC)
$\mathbf{V}$	diagonal matrix form of $\mathbf{v}$
$\bar{\mathbf{V}}$	diagonal matrix form of $\bar{\mathbf{v}}$
$\mathbf{V}_G$	covariance matrix for $p(\mathbf{w} \vert \mathbf{u}, \mathbf{v}, \mathbf{X}, \mathbf{t})$ (in MCMC)
$\mathbf{V}_w$	covariance matrix for approximate posterior distribution of $\mathbf{w}$
$\bar{\mathbf{V}}_w$	sparsified covariance matrix (in ARDEP)
$\mathbf{V}_w^*$	updated covariance matrix (in ARDEP)
$\mathbf{V}_w^{\setminus i}$	covariance matrix for leave-i-out approximation (in ARDEP)
$\mathbf{V}_w^{\setminus S}$	covariance matrix for leave-S-out approximation (in ARDEP)
$\mathbf{w} = (w_1, \ldots, w_D)^\mathsf{T}$	vector of feature weights
$\mathbf{w}_{\text{MAP}}$	maximum a posteriori estimate for $\mathbf{w}$
$\mathbf{w}^s$	s^{th} sample of feature vector $\mathbf{w}$ (in MCMC)
$\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)^\mathsf{T}$	training data matrix including feature vectors
$\mathbf{X}^* = (\mathbf{x}_1^*, \ldots, \mathbf{x}_N^*)^\mathsf{T}$	test data matrix including feature vectors
$\mathbf{X}_S$	test data matrix for observations $i \in S$
$\mathbf{X}^{\setminus S}$	training data matrix for observations $i \notin S$
z_i	auxiliary scalar $\frac{a_i^{\setminus i}}{\sqrt{1+b_i^{\setminus i}}}$ (in ARDEP)
$z_i^{\setminus S}$	auxiliary scalar $\frac{t_i(\mathbf{m}_w^{\setminus S})^\mathsf{T}\mathbf{x}_i}{\sqrt{1+\mathbf{x}_i^\mathsf{T}\mathbf{V}_w^{\setminus S}\mathbf{x_i}}}$ (in ARDEP)
Z_i	normalisation constant for $\hat{q}(\mathbf{w})$ (in ARDEP)
Z_P	normalisation constant for posterior distribution of $\mathbf{w}$
$\tilde{Z}_P$	normalisation constant for $\tilde{q}(\mathbf{w})$ (in ARDEP)

α_i	auxiliary scalar $\frac{\mathcal{N}(z_i;0,1)}{\Psi(z_i)\sqrt{1+b_i^{\setminus i}}}$ (in ARDEP)
ϵ_i	noise for latent variable u_i in probit model
η_j	auxiliary scalar $h_j^2 - r_j - \frac{1}{\lambda^2}$ (in ARDEP)
θ_λ	scaled version of λ^2 for logit model (in LAEP)
κ_i^g	site parameter for $\tilde{g}_i(\mathbf{w})$ (in LAEP)
$\boldsymbol{\kappa}_j^f$	parameter matrix for $\tilde{f}_j(\boldsymbol{\omega}_j)$ (in LAEP)
λ	scale hyperparameter in Laplace prior distribution
$\hat{\lambda}$	selected scale hyperparameter
$\mathbf{\Lambda}$	diagonal matrix form of $\boldsymbol{\sigma}$ (in ARDEP)
$\mathbf{\Lambda}_S$	diagonal matrix form of $\boldsymbol{\sigma}_S$ (in ARDEP)
$\boldsymbol{\mu} = (\mu_1, \ldots, \mu_D)^\mathsf{T}$	vector of auxiliary scale variables (in LAEP)
$\boldsymbol{\nu} = (\nu_1, \ldots, \nu_D)^\mathsf{T}$	vector of auxiliary scale variables (in LAEP)
ρ_i^*	updated site parameter for $\tilde{g}_i(\mathbf{w})$ (in ARDEP)
$\boldsymbol{\rho} = (\rho_1, \ldots, \rho_N)^\mathsf{T}$	vector of site parameters for $\tilde{g}_i(\mathbf{w})$, $i = 1, \ldots, N$ (in ARDEP)
$\boldsymbol{\rho}_S$	vector of site parameters for $\tilde{g}_i(\mathbf{w})$, $i \in S$ (in ARDEP)
ϱ_i^g	site parameter for $\tilde{g}_i(\mathbf{w})$ (in LAEP)
$\boldsymbol{\varrho}_j^f$	parameter vector for $\tilde{f}_j(\boldsymbol{\omega}_j)$ (in LAEP)
σ_i^*	updated site parameter for $\tilde{g}_i(\mathbf{w})$ (in ARDEP)
$\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_N)^\mathsf{T}$	vector of site parameters for $\tilde{g}_i(\mathbf{w})$, $i = 1, \ldots, N$ (in ARDEP)
$\boldsymbol{\sigma}_S$	vector of site parameters for $\tilde{g}_i(\mathbf{w})$, $i \in S$ (in ARDEP)
ς_i	site parameter for $\tilde{g}_i(\mathbf{w})$ (in ARDEP)
ς_i^*	updated site parameter for $\tilde{g}_i(\mathbf{w})$ (in ARDEP)
$\mathbf{\Phi} = (\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_D)$	data matrix including $t_i\mathbf{x}_i$, $i = 1, \ldots, N$ (in ARDEP)
$\bar{\mathbf{\Phi}}$	sparsified data matrix (in ARDEP)
$\mathbf{\Phi}_S$	data matrix for observations $i \in S$ (in ARDEP)
$\boldsymbol{\omega}_j = (w_j, \mu_j, \nu_j)^\mathsf{T}$	vector of variables w_j, μ_j, ν_j (in LAEP)
$\mathbf{\Omega}$	auxiliary matrix $\mathbf{\Lambda} + \mathbf{\Phi}\mathbf{V}\mathbf{\Phi}^\mathsf{T}$ (in ARDEP)
$\bar{\mathbf{\Omega}}$	auxiliary matrix $\mathbf{\Lambda} + \bar{\mathbf{\Phi}}\bar{\mathbf{V}}\bar{\mathbf{\Phi}}^\mathsf{T}$ (in ARDEP)
$\mathbf{\Omega}^{\setminus j}$	auxiliary matrix $\mathbf{\Lambda} + \sum_{m \neq j} \boldsymbol{\phi}_m v_m \boldsymbol{\phi}_m^\mathsf{T}$ (in ARDEP)

Functions and Operators

$\mathrm{CA}(\mathbf{t}^*, \mathbf{X}^*\|\mathbf{t}, \mathbf{X}, \lambda)$	classification accuracy for labelled test data set
$\mathrm{Cov}[\cdot, \cdot]$	covariance (between two random variables)
$\mathrm{Cov}[\cdot]$	covariance matrix (for multivariate distribution)
$\mathrm{Cov}_q[\cdot]$	covariance matrix (for multivariate distribution q)
$\mathrm{D_{KL}}(\cdot \parallel \cdot)$	Kullback-Leibler divergence
$\mathrm{E}[\cdot]$	expected value (for scalar random variable)
$\mathrm{E}[\cdot]$	mean vector (for multivariate distribution)
$\mathrm{E}_q[\cdot]$	mean vector (for multivariate distribution q)
$\mathcal{E}(\cdot; \frac{1}{\mu})$	exponential distribution with mean μ
$f(\mathbf{w})$	linear transformation $\mathbf{w}^\mathsf{T}\mathbf{x}$
$f^*(\mathbf{w})$	linear transformation $\mathbf{w}^\mathsf{T}\mathbf{x}^*$
$f_i(\mathbf{w})$	linear transformation $\mathbf{w}^\mathsf{T}t_i\mathbf{x}_i$
$f_j(\boldsymbol{\omega}_j)$	auxiliary variable term for feature j (in LAEP)
$\tilde{f}_j(\boldsymbol{\omega}_j)$	approximation for $f_j(\boldsymbol{\omega}_j)$ (in LAEP)
$g_i(\mathbf{w})$	likelihood of feature weight vector $\mathbf{w}$ given observation i
$\tilde{g}_i(\mathbf{w})$	approximation for $g_i(\mathbf{w})$
$\tilde{g_i}^*(\mathbf{w})$	updated approximation for $g_i(\mathbf{w})$
$\mathcal{H}(\cdot)$	Heaviside step function
$\mathrm{Inv}{-}\Gamma(\cdot; \alpha, \beta)$	inverse gamma distribution with shape α and scale β
$l^{-1}(\cdot)$	logistic activation function
$\mathcal{L}(\mathbf{v})$	logarithm of $p(\mathbf{t}\|\mathbf{X}, \mathbf{v})p(\mathbf{v}\|\lambda)$ (in ARDEP)
$\mathrm{MLPP}(\mathbf{t}^*, \mathbf{X}^*\|\mathbf{t}, \mathbf{X}, \lambda)$	mean log predictive probability for labelled test data set
$\mathcal{N}(\cdot; \mu, s^2)$	Gaussian distribution with mean μ and variance s^2
$\mathcal{N}(\cdot; \boldsymbol{\mu}, \boldsymbol{\Sigma})$	multivariate Gaussian distribution
$\mathrm{P}(A)$	probability of event A
$\mathrm{P}(A\|B)$	conditional probability of event A given event B
$p(\cdot)$	probability mass function (for discrete distribution)
$p(\cdot)$	probability density function (for continuous distribution)
$p(\cdot, \cdot)$	joint probability distribution
$p(\cdot\|x)$	conditional probability distribution given x
$[\mathbf{x}]_m$	m^{th} element of vector x
$\tilde{q}(\mathbf{w})$	approximate posterior distribution of $\mathbf{w}$ (in ARDEP)
$\hat{q}(\mathbf{w})$	target posterior approximation (in ARDEP)
$\tilde{q}^*(\mathbf{w})$	updated posterior approximation (in ARDEP)
$\tilde{q}^{\setminus i}(\mathbf{w})$	leave-i-out posterior approximation (in ARDEP)
$\tilde{q}^{\setminus S}(\mathbf{w})$	leave-S-out posterior approximation (in ARDEP)
$\Psi(\cdot)$	standard Gaussian cumulative distribution function
$\mathrm{Var}[\cdot]$	variance

Abbreviations

AI	primary auditory cortex
ARD	automatic relevance determination
ARDEP	one of the approximate inference algorithms used in this work
AV	audiovisual
BOLD	blood oxygenation level dependent
EP	expectation propagation
EPI	echo-planar imaging
fMRI	functional magnetic resonance imaging
FOV	field of view
FWHM	full width half maximum
GLM	general linear model
HG	Heschl's gyri
HRF	hemodynamic response function
ITG	inferior temporal gyrus
LAEP	one of the approximate inference algorithms used in this work
loo-ARDEP	alternative algorithm for ARDEP
loso-ARDEP	alternative algorithm for ARDEP
MAP	maximum a posteriori
MCMC	one of the approximate inference algorithms used in this work
mlpp-ARDEP	alternative algorithm for ARDEP
MR	magnetic resonance
MRI	magnetic resonance imaging
MTG	middle temporal gyrus
MVPA	multi-voxel pattern analysis
NMR	nuclear magnetic resonance
PT	planum temporale
RF	radiofrequency
SPM	statistical parametric map
STG	superior temporal gyrus
STS	superior temporal sulcus

1 Introduction

The development of non-invasive imaging techniques, such as functional magnetic resonance imaging (fMRI), has been revolutional for neuroscience, enabling measurements of human brain activity without going inside the skull. A conventional statistical treatment on these measurements is based on generative models explaining the measured activity by a given experimental condition. These models are usually mass-univariate in the sense that they treat the voxels in the brain response image independent of each other, before combining the results into a statistical parametric map (SPM). A common choice is to use the general linear model (GLM) as demonstrated by Friston et al. (1994). Even if these methods have revealed many well interpretable results, they have limitations concerning sensitivity to the selection of the generative model and the significance levels used in hypothesis testing. Since brain imaging data typically embodies complex, high-dimensional correlation structures, it would often be more appropriate to use multivariate models.

During the recent ten years, there has been growing interest in utilisation of pattern recognition methods for analysing brain imaging data (O'Toole et al. 2007; Pereira et al. 2009). These classification methods represent an opposite way of modelling compared to generative methods, by trying to predict the experimental condition from a given activation pattern. In neuroscience literature, this discriminative approach is often referred to as multi-voxel pattern analysis (MVPA). As a multivariate and data-driven approach, MVPA overcomes many of the limitations of mass-univariate generative methods, and thus it may reveal additional information about how different cognitive states are encoded in the human brain. Decoding cognitive states is particularly useful when developing practical applications, such as brain-computer interfaces (Wolpaw et al. 2002) or new clinical markers for distinguishing disease (Klöppel et al. 2008). The possibility to make predictions enables also validation of the model by testing it with new data or by cross-validation, which is essentially important for practical applications.

Along with the benefits, multidimensionality brings also new challenges to the analysis. If the number of parameters is high compared to the number of observations, it becomes difficult to reliably infer the parameters and make relevant conclusions based on the solution. Too high amount of adjustable parameters may also lead to reduced predictive performance due to increased sensitivity to overfitting. In addition, the complexity of computations increases proportionally to the third power of the number of parameters, until it reaches the number of observations. For all these reasons, there is a need for sparsified solutions through feature selection or sparsity promoting priors. (Rasmussen et al. 2012)

The MVPA model used in this work is based on a linear binary classifier that assigns a given activation pattern into the more probable one of two classes according to a linear combination of the voxel activations. Bayesian inference on the voxel weights leads to a multivariate posterior distribution, representing the contribution of different brain locations to the classification and providing uncertainties on both the parameters and the predictions. The prior distribution is chosen from the family of Laplace distributions in order to promote sparsity in the final posterior solution.

Even though using a tightly scaled Laplace-prior favours sparse solutions, a full Bayesian treatment always retains some uncertainty on the parameters. Truly sparse solutions, where the posterior probability mass is concentrated at zero for many of the parameters, would require replacing the full posterior distribution with a point estimate. Using the maximum a posteriori (MAP) estimate for the parameters would be equivalent to L_1-norm regularisation (Tibshirani 1996). In this work, absolutely sparse solutions are obtained by implementing a type II point estimate method, where the Laplace prior is decomposed into a mixture of zero-mean Gaussian priors with separate scale parameters for each weight and these scales are optimised by their approximate marginal MAP estimate. Since many of the scales reduce to zero, also the corresponding voxel weights are forced to be equal to zero and thus pruned out of the model. This approach utilises the idea of automatic relevance determination (ARD), where the Gaussian scales are regarded as relevance hyperparameters to be optimised by maximising their marginal likelihood (MacKay 1994; Neal 1994).

Since exact posterior inference is analytically intractable, approximate methods are needed for summarising the posterior distribution. The implementation of the ARD approach, denoted as ARDEP, is based on the algorithm introduced by Qi et al. (2004), which approximates the posterior distribution as a multivariate Gaussian distribution by using the expectation propagation (EP) procedure (Minka 2001). The original algorithm is carefully rederived, modified for the Laplace prior and implemented in a more efficient computational form. In addition, some practical modifications are applied and alternative criteria for the relevance hyperparameter selection considered to improve the applicability of the method. ARDEP is also compared to two other methods, which integrate approximately over the original Laplace prior. The LAEP solution, obtained by using an algorithm proposed by van Gerven et al. (2010), approximates the posterior distribution as well by a multivariate Gaussian using an EP algorithm. The third approach (MCMC), in turn, is a Markov Chain Monte Carlo simulation method, which generates random samples to simulate the posterior distribution, implemented by using the idea of the Gibbs sampler (Geman and Geman 1984).

The objective of this work is to study, whether the proposed MVPA model is suitable for analysing fMRI activation patterns related to perceiving natural audiovisual stimuli and what kind of results are obtained by the three different approximate inference methods. The example data includes fMRI activation patterns measured from the auditory cortex and some surrounding regions during audiovisual and merely auditory perception of spoken and piano-played versions of popular songs. In the first classification setting, the observations are labelled into piano and speech classes in order to train a model that is able to predict whether a given activation pattern is more probably related to musical or spoken stimuli. The second setting labels the piano observations into auditory and audiovisual classes, aiming at revealing activation patterns related to audiovisual input. The models obtained by different approximate inference methods are compared with respect to both the double-cross-validated predictive performance and the neuroscientifical interpretability of the obtained parameter distributions. The results are also reflected to the previous findings on auditory and audiovisual processing of simpler corresponding stimuli.

2 Background

This chapter reviews some theoretical background that is necessary for understanding this work. The first section serves as a neuroscientifical introduction to the two example classification settings. The second section presents the basic principles of magnetic resonance imaging and explains how functional brain images are acquired. In the last section, I introduce the general idea of Bayesian classification as a basis for the model used in this work.

2.1 Cortical Auditory Processing

Sound is essentially a mechanical wave of pressure propagating through a compressible medium. The human auditory system is capable of perceiving sounds with a frequency between about 20 Hz and 20 kHz. The processing of the original sound begins already in the auricle, which selectively directs sounds coming from different directions into the auditory canal. At the end of the canal, the sound wave vibrates the tympanic membrane, which in turn affects the auditory ossicles transforming the oscillations into a fluid wave in the cochlea. The cochlea is a spiral-shaped cavity including a basilar membrane connected to sensory hair cells that convert the membrane oscillations into neural signals. Since the natural frequency of the basilar membrane depends on spatial location along the cavity, the sound frequency is spatially encoded in neurons. This tonotopy is reserved, when the signals proceed along the ascending pathways through the brainstem towards the auditory cortex. The actual perception of the sound occurs in the cortex, where the neural signals are interpreted by associating them with memories and information from other modalities, such as vision. (Nicholls et al. 2001, pp. 366–376)

The human auditory cortex is located in the superior temporal gyrys (STG), just below the Sylvian fissure (see figure 1 on the following page). The earliest cortical region involved in auditory processing is the primary auditory cortex (AI), which is located in the mediolaterally oriented Heschl's gyri (HG). According to electrophysiological studies in non-human primates, AI and the surrounding secondary regions include several tonotopic maps representing the cochlear frequency encoding (Kaas et al. 2000), with directions orthogonal to the frequency gradients encoding other properties, such as the amplitude of the sound (Read et al. 2002). Promising results have been obtained also by some human studies, suggesting corresponding organisation with frequency gradients along the mediolateral direction of Heschl's gyri (Talavage et al. 2004).

Interpretation of a natural acoustic environment requires much more complicated processing than simply detecting frequencies. This involves both hierarchical and parallel connections to various brain regions. Cortical auditory processing has been illustrated by separating parallel pathways for different computational tasks, originating from AI and proceeding hierarchically through the secondary auditory regions. The ventral pathways, proceeding through the inferior auditory regions towards the superior temporal sulcus (STS) and medial (MTG) and inferior temporal gyrus (ITG), are suggested to be related to speech processing and non-speech audi-

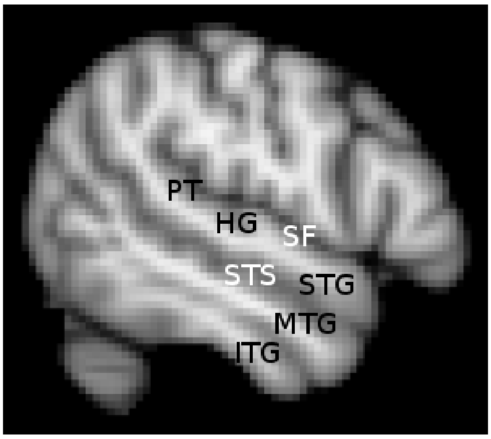

Figure 1: The temporal lobe, separated from the parietal and frontal lobe by the Sylvian fissure (SF), consists of three major gyri: inferior (ITG), medial (MTG) and superior temporal gyrus (STG). The primary auditory cortex is located in the mediolaterally oriented Heschl's gyri (HG). Superior temporal sulcus (STS) between STG and MTG and planum temporale (PT) posterior to HG are suggested to be involved in multi-modal integration.

tory object processing. The dorsal pathways, in turn, proceed through the superior auditory regions towards premotor and prefrontal cortices, and they are suggested to be involved in spatial and audiomotor processing. (Zatorre and Schönwiesner 2011)

This work deals with activation patterns measured from the auditory cortex and some surrounding regions in the superior parts of the temporal lobes during audiovisual and merely auditory perception of spoken and piano-played versions of popular songs. The first classification setting aims at discriminating piano- and speech-related activation patterns. Previous studies, mainly for simpler stimuli, have shown that activated regions during music and speech perception are largely overlapping in STG. However, for example Tervaniemi et al. (2006) have demonstrated more lateral and inferior STG activation for speech sounds compared to music sounds, supporting the special role of ventral pathways and STS in speech processing. Several studies have also observed asymmetry between the hemispheres: right dominance of music-related and especially left dominance of speech-related processing. These effects have been explained by regions specialised for temporal and spectral resolution. (Zatorre and Schönwiesner 2011; DeWitt and Rauschecker 2012)

The second classification setting deals only with the piano observations, trying to discriminate activation patterns related to perception of audiovisual (AV) and merely auditory piano-playing. Previous studies have found many multisensory cortical and sub-cortical convergence zones and even direct connections between primary sensory cortices (Driver and Noesselt 2008; Koelewijn et al. 2010). One possible region to show enhanced activation related to visual perception of hands playing piano is STS, which is commonly regarded to be involved, e.g., in biological motion processing and audiovisual integration (Hein and Knight 2008). Another region suggested to be involved in multi-modal integration is the planum temporale (PT), located posterior to HG. PT is usually larger in the left hemisphere, where it has been found to be activated even during silent lipreading (Calvert et al. 1997). Similar activation has been reported also for silent piano-playing, suggesting that PT is related to learned sensory-motor associations (Hasegawa et al. 2004; Baumann et al. 2005).

2.2 Functional Magnetic Resonance Imaging

2.2.1 Magnetic Resonance Imaging

Magnetic resonance imaging (MRI) is a medical imaging technique based on a physical phenomenon called nuclear magnetic resonance (NMR), which emerges for nuclei with non-zero spin. Since the human body is mostly water, the particle responsible for the phenomenon in practice is the nucleus of the dominant hydrogen isotope, simply consisting of one proton without any neutrons. When a proton gets into a magnetic field, its spin is quantised into two possible states with energy difference directly proportional to the strength of the external magnetic field. Thus, a photon with a frequency corresponding to this energy difference can be absorbed or re-emitted by the proton, causing a swap between the spin states. This nuclear magnetic resonance phenomenon is utilised in magnetic resonance imaging to control the net magnetisation of a group of protons caused by their spins.

Even if spin is a quantum mechanical property, unexplained by classical physics, it is easier to understand the principles of MRI by imaging spin as intrinsic angular momentum of a charged particle, described by a spin vector pointing to a random direction in free space. A spinning charge gives rise to magnetic moment, as well described by a vector directed parallel to the spin vector. Thus, when placed in a magnetic field, the particle tends to precess around the direction of the external magnetic field. The two spin quantum states of a proton can now be pictured as states of precession at a certain angle away from the direction of the external field (parallel state) and from the opposite direction (antiparallel state), with a frequency equal to the resonance frequency of NMR, known as the Larmor frequency.

When a group of protons is in a magnetic field, their spins are distributed between the two states according to probabilities depending on the strength of the external field and the temperature. At equilibrium, the components of spin vectors and magnetic moments perpendicular to the external field cancel each other out. Consequently, since the parallel state is more probable than the antiparallel state, the sum of the magnetic moments can be described as a net magnetisation vector directed parallel to the external field. This net magnetisation can be measured, but it is virtually impossible in the direction of the external field, usually denoted as z-axis. Thus, MRI measures instead the magnetic field in the transverse plane, achieved by a 90° radiofrequency pulse (RF-pulse). By generating a magnetic field rotating at the Larmor frequency in the plane perpendicular to the static external field, also the net magnetisation is enforced to precess, or nutate, towards this plane. By switching this rotating field off at a correct moment, the net magnetisation has been tilted 90 degrees to rotate in the transverse plane around its original direction. An important thing to notice is also, that the spins responsible for the net magnetisation are now all in phase with respect to z-axis. (Levitt 2008, pp. 5–38)

After the RF-pulse, the net magnetisation starts to precess back to the equilibrium, i.e., towards the direction of the static external field. It is this relaxation process, what is particularly interesting with respect to imaging, because the characteristics of the process depend on the properties of the tissue where it occurs. The time constant that determines how fast the z-component of the net magnetisation recovers, while the protons lose their energy, is denoted as the longitudinal relaxation time T_1. Transverse relaxation, in turn, is caused by the loss of coherence between the precession phases of individual spins, due to small differences in the magnetic field they experience and thus in their Larmor frequencies. These differences arise from both the spin-spin interactions (spin-spin relaxation time T_2) and the inhomogenities in the external magnetic field. The actual transverse relaxation time takes both of these effects into account, and it is denoted as T_2^*. (Huettel et al. 2004, pp. 70–73)

In MRI, the structure of interest is placed in a device consisting of three main components. The largest of the components is a powerful magnet, which produces a static magnetic field. Inside the magnet, there are gradient coils, which are used to adjust the strength of the static magnetic field as a function of some spatial dimension, and RF-coils, which produce the RF-pulses and receive the MR-signal emitted back by the spins. (McRobbie et al. 2007, pp. 167–191) By using an appropriate sequence of gradient pulses and RF-pulses, an image representing the desired relaxation time as a function of spatial coordinates can be resolved from the acquired MR-signals. The contrasts in the fMRI data used in this work are based on different T_2^* relaxation times of oxygen-rich and oxygen-poor blood, exposed by a fast echo-planar imaging (EPI) sequence.

2.2.2 Acquisition of T_2^*-weighted Contrast

In a typical MRI setting, a three-dimensional image is obtained slice by slice, by exciting only a thin volume of the structure during each MR-signal acquisition. The slice selection is controlled by applying a one-dimensional gradient pulse simultaneously with the RF-pulse. Since the Larmor frequency depends on the strength of the external magnetic field, the RF-pulse is able to excite only the spins lying inside a restricted range in the direction of the gradient. The gradient pulse causes also some differences in the precession phase between the spins near the edges of this region. This dephasing is compensated by applying an opposite gradient pulse right after the RF-pulse.

The remaining two dimensions are typically encoded by using sequences, where the phase of the received signal depends on another one of the dimensions and the frequency on the other one. The phase encoding is achieved by applying a one-dimensional gradient pulse between the RF-pulse and the signal acquisition into each sequence. During this short pulse, the precession frequencies of spins with different coordinates along the gradient dimension differ from each other, leading to dephasing. The frequency encoding, in turn, is achieved by applying a gradient pulse along the remaining dimension during the signal acquisition. The signal samples received during the sequences are collected into a two-dimensional grid, and the final two-dimensional image is then obtained by using the Fourier transform. (McRobbie et al. 2007, pp. 108–136)

When trying to expose T_2- or T_2^*-weighted contrasts, the time between the RF-pulse and the signal acquisition has to be long enough for the spins to dephase. On the other hand, it must be short enough to still retain some of the transverse magnetisation. The two most common MRI sequence types, gradient-echo and spin-echo sequences, provide a solution for this problem by generating an echo of the original signal and using this echo for the signal acquisition. The time between the 90° RF-pulse and the center of the echo is called the echo time (TE), which is an essential parameter to adjust when exposing different contrasts. Another important parameter is the repetition time (TR) of subsequent 90° RF-pulses, which has to be long enough for the full recovery of the longitudinal magnetisation, in order to minimise the effect of differences in T_1 on the following sequences. The difference between spin-echo and gradient-echo is, that spin-echo sequences invert the net magnetisation by a 180° RF-pulse, whereas gradient-echo is produced by using a negative gradient pulse before the signal acquisition. Since the additional RF-pulse eliminates the effect of field inhomogenities, gradient-echo is the choice for T_2^*-weighted imaging and spin-echo for T_2-contrast. (Huettel et al. 2004, pp. 99–110)

When imaging rapid changes in the structure of interest, such as brain activity, the speed requirements of the image acquisition become crucial. Echo-planar imaging (EPI) is a fast technique, especially suitable for T_2^*-weighted imaging, allowing an entire two-dimensional image to be acquired by a single RF-pulse. The technique speeds up the gradient-echo approach by rapidly applying subsequent negative and positive gradient pulses and applying a perpendicular phase-encoding pulse between each of them. By collecting MR-signal throughout the sequence, the whole acquisition grid becomes filled. To prevent additional artifacts, this raw signal must also be sorted and realigned, before the reconstruction of the final image through the Fourier transform. As a price for the high acquisition speed, the spatial resolution and signal-to-noise ratio are significantly reduced. (Huettel et al. 2004, pp. 120–123) For this reason, functional experiments are usually preceded by acquisition of high-resolution structural images for better identification of the structural components in the EPI images.

2.2.3 Blood Oxygenation Level Dependent Brain Imaging

Hemoglobin is an iron-containing metalloprotein, which is responsible of oxygen-transportation in red blood cells. As its oxygenated form, hemoglobin is diamagnetic, i.e., essentially non-magnetic, whereas deoxygenated hemoglobin is paramagnetic. The greater magnetic susceptibility of deoxygenated blood increases local field inhomogenities and thus leads to a shorter T_2^* relaxation time. This effect, first demonstrated by Thulborn et al. (1982), provides the theoretical basis for functional brain imaging by MRI.

Only a small increase in local neuronal activity is required to significantly increase the energy demand and oxygen consumption in a brain region. Thus, right after the activation, the consentration of deoxygenated hemoglobin increases in the local veins. However, by a lag of a few seconds, the increased need of oxygen is overcompensated by a disproportionate increase in blood perfusion, leading instead to decreased venous consentration of deoxygenated hemoglobin and thus to an increased signal intensity in T_2^*-weighted images.

This blood oxygenation level dependent (BOLD) signal is utilised in functional MRI (fMRI) to study the activation of different brain regions during a certain stimulus or task. A typical experimental design consists of several blocks separated by rest periods to restore and determine the reference level of activation. The activations due to the stimuli are assumed to elicit a signal that follows a hemodynamic response function (HRF), which models the physiological lags occuring before the blood perfusion responds to the change in neuronal activation. After the initial lag, the signal increases rapidly to its maximum and then decreases on a stable level, which is held as long as the stimulus continues. After the presentation of the stimulus ends, there is again a lag of a few seconds in the signal before descending on the rest level. At both ends of the stimulus, the signal should actually fall below the rest level for a moment. The initial dip before the increase of the blood perfusion is not usually observed, but the undershoot after the stimulus is instead visible even with standard EPI techniques. (Huettel et al. 2004, pp. 159–184)

At low field strengths, the contrasts caused by the oxygenation level differences are quite subtle. In addition to a strong magnet, separating the effect of stimulus-correlated neural activation from other effects requires further processing of the acquired data. In order to make generalised conclusions, the same experiment must also be repeated for several subjects. Even if the brain volumes of different subjects are co-registered anatomically, they may still show significant functional differences, which brings challenges to the interpretation of the results.

The conventional way to analyse fMRI data is to take the activation time-series of one voxel at a time from each subject and model them, e.g., as a linear combination of the assumed hemodynamic response functions caused by different stimuli or stimulus features. The parameter estimates for different conditions, or their contrasts, are then combined into a statistical parametric map (SPM) to represent the desired activation patterns (Friston et al. 1994). In this work, the conventional general linear model (GLM) is used as a reference method to validate the results obtained by an inversely directed classification model, which instead predicts the stimulus class from an individual activation pattern by dealing with all voxels at the same time. In neuroscience literature, this discriminative approach is often referred to as multi-voxel pattern analysis (MVPA). As a multivariate and data-driven approach, MVPA may reveal additional information about the complex correlation structure of the activation patterns representing different cognitive states without being restricted by the choice of the generative model.

2.3 Bayesian Classification

2.3.1 Basics of Probability Theory

This subsection briefly presents some basic concepts of probability theory encountered in this work. For more thorough definitions, I refer to the textbook by Shiryaev (1996). Consider X as a random variable following a discrete probability distribution with a countable amount of possible values x. The probability distribution of X is determined by a probability mass function $p(x)$, which defines the probability for X to have a value x:

$$\mathrm{P}(X = x) = p(x). \tag{1}$$

Thus, the probability for X to have a realisation from a closed interval between x_1 and x_2, where $x_1 \leq x_2$, is defined by the sum

$$\mathrm{P}(X \in [x_1, x_2]) = \sum_{x \in [x_1, x_2]} p(x). \tag{2}$$

If X is instead a continuous random variable, its mass function would be zero for all values. In this case, the probability distribution has to be determined by a probability density function, denoted here as well by $p(x)$. The probability for X to have a realisation between x_1 and x_2 is now defined by the integral

$$\mathrm{P}(X \in [x_1, x_2]) = \int_{x_1}^{x_2} p(x)\, \mathrm{d}x. \tag{3}$$

Note, that throughout this work, $p(x)$ is generally called a probability distribution, regardless of whether it refers to a discrete random variable or a continuous one. Even if this may seem confusing, it does not cause problems, as long as the different natures of discrete and continuous distribution functions are acknowledged.

The expected value or the mean of a discrete random variable X is defined by the sum over all possible values x weighted by the probabilities $\mathrm{P}(X = x)$:

$$\mathrm{E}[X] = \sum_{x} (p(x)x). \tag{4}$$

For a continuous random variable, the sum is again replaced by an integral:

$$\mathrm{E}[X] = \int_{x} p(x)x\, \mathrm{d}x. \tag{5}$$

The variance of X, in turn, is defined as the expected squared difference between X and its expected value:

$$\mathrm{Var}[X] = \mathrm{E}[(X - \mathrm{E}(X))^2] = \mathrm{E}[X^2] - (\mathrm{E}[X])^2. \tag{6}$$

A more intuitively scaled measure, standard deviation, is obtained by taking the square root of variance. (Ross 2000, pp. 23–46)

Suppose now, that there are two random variables X and Y. The joint distribution of X and Y is determined by a two-dimensional distribution function $p(x, y)$, which defines the probability of a realisation $(X = x, Y = y)$ or the corresponding probability density in the continuous case. This joint distribution determines also the marginal distributions $p(x)$ and $p(y)$. For example, the marginal distribution $p(x)$ is obtained by summing

$$p(x) = \sum_{y} p(x, y) \tag{7}$$

or integrating

$$p(x) = \int_{y} p(x, y) \, \mathrm{d}y \tag{8}$$

over all possible values y. The conditional distribution of X given $Y = y$, in turn, is defined as

$$p(x|y) = \frac{p(x, y)}{p(y)}. \tag{9}$$

Rewriting this for both $p(x|y)$ and $p(y|x)$ gives the product rule

$$p(x, y) = p(x|y)p(y) = p(y|x)p(x), \tag{10}$$

which, in turn, leads to the following expression for the conditional distribution:

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)}. \tag{11}$$

This equation, known as the Bayes' theorem, describes an important relationship between the conditional distributions $p(x|y)$ and $p(y|x)$. It is also an essential tool for Bayesian inference, which is introduced in the following subsection.

The rules above generalise as well for multiple random variables, which are often gathered together in vectors. For multidimensional probability distributions, the expected value and variance are replaced by a mean vector including the expected values of the marginal distributions and a covariance matrix, where the covariance between two random variables X and Y is defined as

$$\mathrm{Cov}[X, Y] = \mathrm{E}[(X - \mathrm{E}[X])(Y - \mathrm{E}[Y])], \tag{12}$$

expressing their linear dependence. (Bishop 2006, pp. 12–20)

The Gaussian distribution is a commonly used continuous probability distribution, parametrised by its mean and variance. The density function of the Gaussian distribution is

$$p(x) = \mathcal{N}(x; \mu, s^2) = \frac{1}{s\sqrt{2\pi}} \mathrm{e}^{\frac{(x-\mu)^2}{2s^2}}, \tag{13}$$

where μ is the expected value and s the standard deviation of the distribution. For a D-dimensional random vector $\mathbf{x}$, the density function is given by

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = (2\pi)^{-\frac{D}{2}} |\boldsymbol{\Sigma}|^{-\frac{1}{2}} \mathrm{e}^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathsf{T}} \boldsymbol{\Sigma}^{-1} (\mathbf{x}-\boldsymbol{\mu})}, \tag{14}$$

where $\boldsymbol{\mu}$ is the mean vector and $\boldsymbol{\Sigma}$ the covariance matrix of the distribution. (Bishop 2006, p. 78)

2.3.2 Bayesian Inference

The concept of probability has several different interpretations. Frequentists define the probability of an event strictly as the limit of the relative frequency of its occurrence, when repeating a random and well-defined experiment. The Bayesian interpretation provides a broader point of view by regarding probability as a degree of belief, which can be updated based on evidence.

Consider a statistical model that assumes a random vector $\mathbf{y}$ to follow a probability distribution $p(\mathbf{y}|\boldsymbol{\theta})$ parametrised by an unobservable parameter vector $\boldsymbol{\theta}$. With respect to modelling, the essential difference between Bayesian inference and the traditional frequentistic statistical inference lies in the way they treat the model parameters. Whereas frequentists try to determine fixed values, i.e., a point estimate, for the parameters, perhaps with some confidence intervals to describe the uncertainty, Bayesians model also the uncertainty on the parameters with a probability distribution.

In the Bayesian framework, $p(\boldsymbol{\theta})$ is a prior distribution assigned on $\boldsymbol{\theta}$, reflecting a priori beliefs on the parameter vector. After receiving an observation on $\mathbf{y}$, the distribution over $\boldsymbol{\theta}$ is updated according to the Bayes' theorem (equation 11):

$$p(\boldsymbol{\theta}|\mathbf{y}) = \frac{p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{y})}. \tag{15}$$

The updated distribution $p(\boldsymbol{\theta}|\mathbf{y})$ is called the posterior distribution of $\boldsymbol{\theta}$, reflecting a posteriori beliefs on the parameter vector. The conditional probability (density) $p(\mathbf{y}|\boldsymbol{\theta})$ as a function of $\boldsymbol{\theta}$ is called the likelihood function. Since $p(\mathbf{y})$ depends only on the fixed $\mathbf{y}$, the posterior is directly proportional to the product of the likelihood and the prior:

$$p(\boldsymbol{\theta}|\mathbf{y}) \propto p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta}). \tag{16}$$

The remaining term $p(\mathbf{y})$ can be simply thought of as a normalisation constant for the posterior distribution. Another interpretation is to regard it as the marginal likelihood for the whole model, obtained by integrating over all possible $\boldsymbol{\theta}$:

$$p(\mathbf{y}) = \int_{\boldsymbol{\theta}} p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})\,\mathrm{d}\boldsymbol{\theta}. \tag{17}$$

The posterior distribution is used also, when making predictions on future observations. Consider $\mathbf{y}^*$ as an observation vector that has not yet been observed. Marginalising $\boldsymbol{\theta}$ out from the joint posterior distribution of $\mathbf{y}^*$ and $\boldsymbol{\theta}$ leads to

$$p(\mathbf{y}^*|\mathbf{y}) = \int_{\boldsymbol{\theta}} p(\mathbf{y}^*, \boldsymbol{\theta}|\mathbf{y})\,\mathrm{d}\boldsymbol{\theta} = \int_{\boldsymbol{\theta}} p(\mathbf{y}^*|\boldsymbol{\theta}, \mathbf{y})p(\boldsymbol{\theta}|\mathbf{y})\,\mathrm{d}\boldsymbol{\theta} = \int_{\boldsymbol{\theta}} p(\mathbf{y}^*|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{y})\,\mathrm{d}\boldsymbol{\theta}, \tag{18}$$

where $p(\mathbf{y}^*|\boldsymbol{\theta})$ is the observation model for $\mathbf{y}^*$ and $p(\boldsymbol{\theta}|\mathbf{y})$ is the posterior distribution of $\boldsymbol{\theta}$. The distribution $p(\mathbf{y}^*|\mathbf{y})$ determines the predictive distribution of $\mathbf{y}^*$ conditional on the observed $\mathbf{y}$, and it is thus called the posterior predictive distribution. (Gelman et al. 2004, pp. 3–14)

2.3.3 Classification

In statistics and machine learning, classification refers to assigning a given observation into one of a countable set of discrete classes, based on its characteristic features described as an input vector $\mathbf{x}$. As a distinction to unsupervised clustering, which divides a group of feature vectors into subgroups using only the similarities between them, classification is a form of supervised learning based on a training set of labelled observations with both the feature vector $\mathbf{x}_i$ and the corresponding target class t_i known. The learning process produces a discriminant function, which defines decision boundaries between different classes, enabling classification of any given feature vector.

In Bayesian classification, inference and decision stages are separated by first determining the posterior predictive probability distribution for the class t^* of a given feature vector $\mathbf{x}^*$ and then using this distribution to make decisions. This way also the uncertainties on the decisions become modelled. Consider a training data set with N labelled feature vectors gathered in a matrix $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)^\mathsf{T}$ and the correspondent target classes gathered in a vector $\mathbf{t} = (t_1, \ldots, t_N)^\mathsf{T}$. The discriminative approach models directly the conditional dependence of class t on feature vector $\mathbf{x}$ by a probability distribution $p(t|\mathbf{x}, \boldsymbol{\theta}_d)$ parametrised by $\boldsymbol{\theta}_d$. Applying a prior distribution on $\boldsymbol{\theta}_d$ leads to a posterior predictive distribution

$$p(t^*|\mathbf{x}^*, \mathbf{t}, \mathbf{X}) = \int_{\boldsymbol{\theta}_d} p(t^*|\mathbf{x}^*, \boldsymbol{\theta}_d) p(\boldsymbol{\theta}_d|\mathbf{t}, \mathbf{X})\, \mathrm{d}\boldsymbol{\theta}_d, \tag{19}$$

where $p(\boldsymbol{\theta}_d|\mathbf{t}, \mathbf{X}) \propto p(\mathbf{t}|\mathbf{X}, \boldsymbol{\theta}_d)p(\boldsymbol{\theta}_d)$ is the posterior distribution of $\boldsymbol{\theta}_d$.

Another alternative would be the generative approach, which models instead the dependence of feature vector $\mathbf{x}$ on class t by a class-conditioned distribution $p(\mathbf{x}|t, \boldsymbol{\theta}_g)$ with a parameter vector denoted by $\boldsymbol{\theta}_g$. Applying a prior distribution on $\boldsymbol{\theta}_g$ leads to a generative posterior distribution for $\mathbf{x}^*$ conditional on t^*:

$$p(\mathbf{x}^*|t^*, \mathbf{t}, \mathbf{X}) = \int_{\boldsymbol{\theta}_g} p(\mathbf{x}^*|t^*, \boldsymbol{\theta}_g) p(\boldsymbol{\theta}_g|\mathbf{t}, \mathbf{X})\, \mathrm{d}\boldsymbol{\theta}_g. \tag{20}$$

The posterior predictive distribution for t^* conditional on $\mathbf{x}^*$ is obtained by

$$p(t^*|\mathbf{x}^*, \mathbf{t}, \mathbf{X}) \propto p(\mathbf{x}^*|t^*, \mathbf{t}, \mathbf{X}) p(t^*|\mathbf{t}), \tag{21}$$

where

$$p(t^*|\mathbf{t}) = \int_{\boldsymbol{\theta}_t} p(t^*|\boldsymbol{\theta}_t) p(\boldsymbol{\theta}_t|\mathbf{t})\, \mathrm{d}\boldsymbol{\theta}_t \tag{22}$$

is the posterior predictive distribution for t^* prior to observing $\mathbf{x}^*$, resulting from modelling the prior distribution of t by $p(t|\boldsymbol{\theta}_t)$ with a separate parametrisation and applying a prior distribution on $\boldsymbol{\theta}_t$. (Bishop 2006, pp. 179–220)

In principle, both of these two alternatives are correct Bayesian approaches for a classification problem. Since the discriminative approach models directly what is desired with respect to classification, it typically has less parameters to deal with and thus becomes more feasible in practice. Furthermore, determining the class-conditional densities may be a difficult problem, especially when there is a large amount of features in $\mathbf{x}$ to model, and incorrect assumptions on the distributions often result in reduced predictive performance. However, in the case of inadequate training data with missing values or outliers, it may be useful to have also access to the distribution of $\mathbf{x}$, provided by the generative approach. One more significant argument for the choice may be obtained by considering which one of the approaches is better supported by the available prior information. (Rasmussen and Williams 2006, p. 35) In this work, the more straightforward discriminative Bayesian approach is used to classify a given brain activation pattern $\mathbf{x}$, including fMRI activation values from several hundreds of brain locations, into one of two stimulus classes.

3 Data

The fMRI data used in this work was collected as a part of a practical course on non-invasive brain imaging (Aalto University course Tfy-99.3760), and it is used also in other projects. All the experiments were carried out at the Advanced Imaging Centre of Aalto University.

3.1 Experimental Design

16 subjects (6 women, age 21–40 years, average age 27.5 years) were listening to spoken, sung and piano-played versions of three popular songs: *Kesäyö* (*Summertime*), *Kulkuset* (*Jingle Bells*) and *Oi niitä aikoja* (*Those Were the Days*). Both audiovisual and merely auditory versions of all these nine combinations were presented in a counterbalanced order, separated by rest periods of five secods. In the auditory versions, the visual stimulus of speaking or singing head or piano-playing hands was replaced by a fixation cross, which was shown also during the rest periods and for a period of ten seconds in the beginning of the experiment. The experiment included also mixtures of the auditory singing and piano versions, but the measurements related to them were not included in the data set used in this work.

3.2 Data Collection

Brain activation during the experiment was measured using a 3.0 T MRI scanner with an eight-channel head coil. Blood oxygenation level dependent (BOLD) fMRI signal was acquired using an echo-planar imaging (EPI) sequence with a repetition time of $TR = 2.0$ s and an echo time of $TE = 32$ ms. 34 near-horizontal slices were collected, with the position slightly inclined to be parallel to the plane penetrating cerebellum and prefrontal cortex. An acquisition matrix of 64 x 64 pixels and a field of view (FOV) of 22 cm x 22 cm in slice directions, with a slice thickness of 4.0 mm, give a spatial resolution of 3.4 mm x 3.4 mm x 4.0 mm. A total of 1160 three-dimensional fMRI samples was acquired from each subject during the experiment.

3.3 Preprocessing

The raw fMRI data was preprocessed following the steps proposed for the analysis tool FEAT in FMRIB Software Library (Smith et al. 2004). To ensure the stability of magnetisation, five samples from each subject were removed from the beginning of the experiment. Motion correction between different samples of the same subject was carried out by ridig-body transformations, allowing head movement of less than 1 mm in any direction. After motion correction, the brain was extracted by removing other tissues from the images. To remove low frequency artefacts, highpass temporal filtering was applied. For noise reduction, every individual sample was also smoothed spatially with a full width half maximum (FWHM) of 10 mm. Finally, a two-step registration process was carried out to align the images of different subjects with each other and a standard brain. Before the functional experiment, high-resolution structural images were recorded from each subject. Transformations from these structural images to an ICBM–152 standard image (Mazziotta et al. 2001) and from the functional images to the structural images were first determined, and these transformations were then combined to register the functional images to the standard space.

To reduce the amount of data for this work, the region of interest was restricted to comprise only the auditory cortex and some surrounding volumes related to auditory processing. In addition, every second voxel in each spatial dimension was removed from the region of interest, justified by the spatial smoothing. The masked 4D-images were then standardised by setting the mean of each individual time-series to zero and by scaling its standard deviation to be one. After removing the last samples of the singing and piano versions of the song *Kulkuset* to even the amounts of samples with the speech versions, the final data set included the preprocessed and standardised fMRI activation values of $D = 707$ voxels from $K = 16$ subjects recorded at 157 time-points of each of the six stimulus types (auditory piano, auditory speech, auditory singing, audiovisual piano, audiovisual speech and audiovisual singing).

3.4 Labelling of Observations for Different Classification Settings

Subsets of the final preprocessed data set are used as two different classification settings. The first one compares the 707-voxel fMRI activation patterns from all the 5024 speech time-points (both auditory and audiovisual) with the data from the 5024 piano time-points, whereas the other one compares the 2512 auditory piano time-points with the 2512 audiovisual piano time-points. The compared observations are labelled into different target classes, denoted here as $t = -1$ and $t = 1$, resulting in samples $(\mathbf{x}_i, t_i)$ consisting of 707-dimensional feature vectors $\mathbf{x}_i$ and their target classes t_i.

4 Model

The analysis methods used in this work aim at revealing activation patterns related to certain conditions, e.g., listening to speech, by trying to construct a discriminative model that predicts the condition based on an observed fMRI activation pattern. The model parameters represent the contribution of different locations to the predictions, and their posterior probability distribution can also be used to test the classifier with new data. Three different methods are used for the approximate inference, but the underlying Bayesian model is similar for all of them: a linear binary classifier for the selected two conditions and univariate Laplace priors on the weights of the classifier.

4.1 Linear Binary Classifier

A linear binary classifier classifies a given feature vector $\mathbf{x}_i$ into either one of two classes based on a linear combination $\mathbf{w}^\mathsf{T}\mathbf{x}_i$, where $\mathbf{w}$ is a vector of the feature weights. To add uncertainty into the classification, consider t_i to be generated by a noisy latent variable u_i distributed around $\mathbf{w}^\mathsf{T}\mathbf{x}_i$ according to the following model:

$$u_i = \mathbf{w}^\mathsf{T}\mathbf{x}_i + \epsilon_i, \tag{23}$$

$$t_i = \begin{cases} 1, & \text{when } u_i > 0, \\ -1, & \text{when } u_i < 0, \end{cases} \tag{24}$$

where ϵ_i are independent noise terms. Assuming ϵ_i to be Gaussian noise with the unit standard deviation ($\epsilon_i \sim \mathcal{N}(0,1)$) leads to the probit model, which defines the probability for a feature vector $\mathbf{x}_i$ to belong to class $t_i = 1$ as

$$\mathrm{P}(t_i = 1|\mathbf{w}, \mathbf{x}_i) = \Psi(\mathbf{w}^\mathsf{T}\mathbf{x}_i), \tag{25}$$

where Ψ is the standard Gaussian cumulative distribution function transforming the real-valued linear combination $\mathbf{w}^\mathsf{T}\mathbf{x}_i$ into probability range $(0,1)$. Since the class probabilities must sum to one, the probability for $\mathbf{x}_i$ to belong to class $t_i = -1$, is given by

$$\mathrm{P}(t_i = -1|\mathbf{w}, \mathbf{x}_i) = 1 - \Psi(\mathbf{w}^\mathsf{T}\mathbf{x}_i) = \Psi(-\mathbf{w}^\mathsf{T}\mathbf{x}_i). \tag{26}$$

The probit model is used as a default choice in this work, because of its computational convenience. One of the approximate inference methods (see 5.2 Expectation Propagation on Laplace Prior), though, uses instead the logit model by replacing the probit activation function Ψ by the logistic one. After rescaling the horizontal axis these two functions become closely similar, differing mainly by the asymptotical behaviour. In theory, the logistic activation function may be considered more robust with respect to outliers, but in practice they usually produce quite similar results (Nickisch and Rasmussen 2008).

4.2 Bayesian Inference on the Weights

Given a training data set $\{(\mathbf{x}_1, t_1), \ldots, (\mathbf{x}_N, t_N)\}$, where the samples are assumed to be independent and identically distributed, the likelihood of a weight vector $\mathbf{w}$ can be written as

$$p(\mathbf{t}|\mathbf{w}, \mathbf{X}) = \prod_{i=1}^{N} p(t_i|\mathbf{w}, \mathbf{x}_i) = \prod_{i=1}^{N} \Psi(t_i \mathbf{w}^{\mathsf{T}} \mathbf{x}_i), \tag{27}$$

where $\mathbf{t} = (t_1, \ldots, t_N)^{\mathsf{T}}$ and $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)^{\mathsf{T}}$. Applying a prior distribution $p(\mathbf{w}|\lambda)$ with a constant hyperparameter λ and using the Bayes' theorem, the posterior distribution over $\mathbf{w}$ is obtained by

$$p(\mathbf{w}|\mathbf{t}, \mathbf{X}, \lambda) = \frac{p(\mathbf{t}|\mathbf{w}, \mathbf{X}) p(\mathbf{w}|\lambda)}{p(\mathbf{t}|\mathbf{X}, \lambda)} \propto \prod_{i=1}^{N} \Psi(t_i \mathbf{w}^{\mathsf{T}} \mathbf{x}_i) p(\mathbf{w}|\lambda), \tag{28}$$

where the normalisation constant $p(\mathbf{t}|\mathbf{X}, \lambda) = \int_{\mathbf{w}} p(\mathbf{t}|\mathbf{w}, \mathbf{X}) p(\mathbf{w}|\lambda) \, \mathrm{d}\mathbf{w}$ is the marginal likelihood for the selected hyperparameter value λ.

Exact inference on the posterior distribution is often intractable, but it can be approximated by different computational methods. The three approximate inference methods used in this work are described in the following chapter. Mapping the obtained posterior distribution into the brain, for example by presenting the marginal probabilities for w_j to be positive or negative according to some colour scale, may reveal information about the activation patterns more related to either of the two stimulus classes.

To appropriately test the model and measure its predictive performance, some new data is needed. Given a test sample $\mathbf{x}^*$, the posterior predictive distribution over t^* is

$$p(t^*|\mathbf{x}^*, \mathbf{t}, \mathbf{X}, \lambda) = \int_{\mathbf{w}} p(t^*|\mathbf{w}, \mathbf{x}^*) p(\mathbf{w}|\mathbf{t}, \mathbf{X}, \lambda) \, \mathrm{d}\mathbf{w}. \tag{29}$$

If the posterior over $\mathbf{w}$ has been approximated as a multivariate Gaussian distribution $p(\mathbf{w}|\mathbf{t}, \mathbf{X}, \lambda) = \mathcal{N}(\mathbf{m}_w, \mathbf{V}_w)$, the distribution over its linear transformation $f^*(\mathbf{w}) = \mathbf{w}^{\mathsf{T}} \mathbf{x}^*$ becomes also Gaussian with parameters $\mathrm{E}[f^*(\mathbf{w})] = \mathbf{m}_w^{\mathsf{T}} \mathbf{x}^*$ and $\mathrm{Var}[f^*(\mathbf{w})] = (\mathbf{x}^*)^{\mathsf{T}} \mathbf{V}_w \mathbf{x}^*$. Consequently, according to Rasmussen and Williams (2006, p. 74), the predictive distribution can be written in a simple form:

$$\begin{aligned} p(t^*|\mathbf{x}^*, \mathbf{t}, \mathbf{X}, \lambda) &= \int_{-\infty}^{\infty} p(t^*|f^*(\mathbf{w})) p(f^*(\mathbf{w})|\mathbf{t}, \mathbf{X}, \lambda) \, \mathrm{d}f^*(\mathbf{w}) \\ &= \int_{-\infty}^{\infty} \Psi(t^* f^*(\mathbf{w})) \mathcal{N}(\mathrm{E}[f^*(\mathbf{w})], \mathrm{Var}[f^*(\mathbf{w})]) \, \mathrm{d}f^*(\mathbf{w}) \\ &= \Psi\left(\frac{t^* \mathbf{m}_w^{\mathsf{T}} \mathbf{x}^*}{\sqrt{1 + (\mathbf{x}^*)^{\mathsf{T}} \mathbf{V}_w \mathbf{x}^*}} \right). \end{aligned} \tag{30}$$

To classify the test sample $\mathbf{x}^*$, the more probable one of the two classes is chosen for the prediction $\hat{t}^*$. Given a labelled test data set $\{(\mathbf{x}_1^*, t_1^*), \ldots, (\mathbf{x}_{N^*}^*, t_{N^*}^*)\}$ with the correct classes t_i^* known, the proportion of correct predictions, where $\hat{t}_i^* = t_i^*$, can be calculated to describe the predictive classification accuracy (CA). Since the more probable class $\hat{t_i^*}$ depends here only on the sign of $\frac{\mathbf{m}_w^\mathsf{T}\mathbf{x}_i^*}{\sqrt{1+(\mathbf{x}_i^*)^\mathsf{T}\mathbf{V}_w\mathbf{x}_i^*}}$, $\mathrm{CA}(\mathbf{t}^*, \mathbf{X}^*|\mathbf{t}, \mathbf{X}, \lambda)$ can be written as

$$\begin{aligned} \mathrm{CA}(\mathbf{t}^*, \mathbf{X}^*|\mathbf{t}, \mathbf{X}, \lambda) &= \frac{1}{N^*}\sum_{i=1}^{N^*} \mathcal{H}\left(\frac{t_i^*\mathbf{m}_w^\mathsf{T}\mathbf{x}_i^*}{\sqrt{1+(\mathbf{x}_i^*)^\mathsf{T}\mathbf{V}_w\mathbf{x}_i^*}}\right) \\ &= \frac{1}{N^*}\sum_{i=1}^{N^*} \mathcal{H}\big(t_i^*\mathbf{m}_w^\mathsf{T}\mathbf{x}_i^*\big), \end{aligned} \tag{31}$$

where $\mathcal{H}$ is the Heaviside step function, defined as follows:

$$\mathcal{H}(x) = \begin{cases} 0, & \text{if } x < 0, \\ \frac{1}{2}, & \text{if } x = 0, \\ 1, & \text{if } x > 0. \end{cases} \tag{32}$$

Another measure for the goodness of the model, taking also the uncertainty on the predictions into account, can be obtained by calculating the mean log predictive probability (MLPP) for the correct classes t_i^*:

$$\begin{aligned} \mathrm{MLPP}(\mathbf{t}^*, \mathbf{X}^*|\mathbf{t}, \mathbf{X}, \lambda) &= \frac{1}{N^*}\sum_{i=1}^{N^*} \ln p(t_i^*|\mathbf{x}_i^*, \mathbf{t}, \mathbf{X}, \lambda) \\ &= \frac{1}{N^*}\sum_{i=1}^{N^*} \ln \Psi\left(\frac{t_i^*\mathbf{m}_w^\mathsf{T}\mathbf{x}_i^*}{\sqrt{1+(\mathbf{x}_i^*)^\mathsf{T}\mathbf{V}_w\mathbf{x}_i^*}}\right). \end{aligned} \tag{33}$$

4.3 Laplace Prior Distribution

Through the selection of the prior distribution $p(\mathbf{w}|\lambda)$, a priori beliefs about the feature weights $\mathbf{w}$ can be included into the model. In this work, the shape of the prior is chosen to promote sparsity in the final posterior distribution. Each individual weight w_j is given here a univariate Laplace prior distribution

$$p(w_j|\lambda) = \frac{1}{2\lambda}\mathrm{e}^{\frac{-|w_j|}{\lambda}}, \tag{34}$$

where $\lambda > 0$ is a constant scale hyperparameter limiting the magnitude of the weights and at the same time the amount of weights of a relevant magnitude. Without any a priori assumptions about the dependencies between the individual weights, the full prior distribution $p(\mathbf{w}|\lambda)$ is the product of the individual univariate priors. Thus, the posterior distribution becomes

$$p(\mathbf{w}|\mathbf{t}, \mathbf{X}, \lambda) \propto p(\mathbf{t}|\mathbf{w}, \mathbf{X}) \prod_{j=1}^{D} p(w_j|\lambda). \tag{35}$$

Figure 2 illustrates the sparsity promoting effect of the Laplace prior compared to the Gaussian prior in a toy example of two features. As noticed, the Laplace prior enforces the mode of the posterior distribution onto another one of the weight axes.

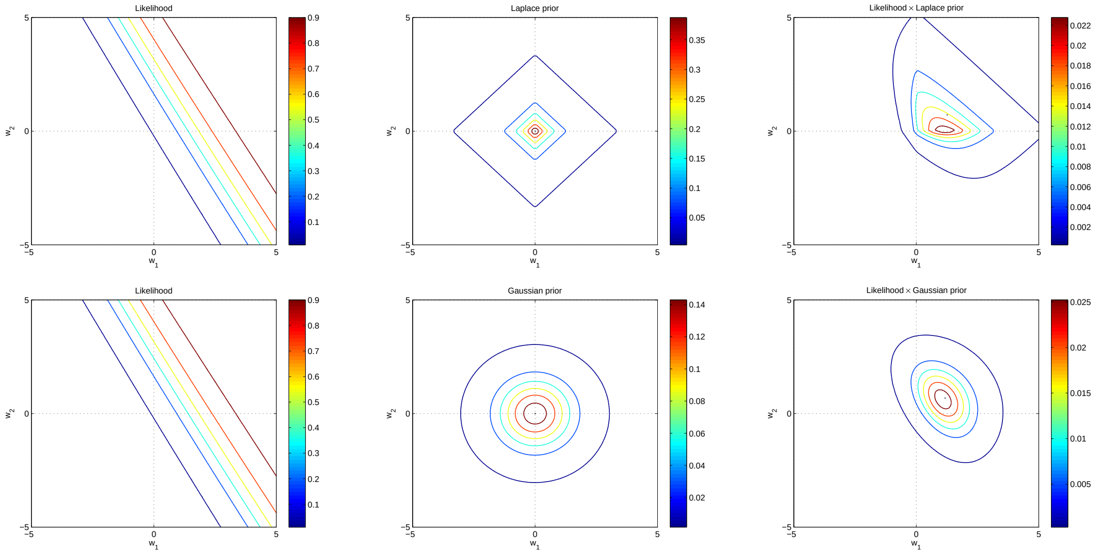

Figure 2: A toy example of the effect of the Laplace prior to the posterior distribution with two features and six toy observations. The upper contour plots describe the likelihood function $p(\mathbf{t}|\mathbf{w}, \mathbf{X})$, the Laplace prior $p(\mathbf{w}|\lambda)$ with $2\lambda^2 = 1$ and their product, which directly proportional to the resulting posterior distribution $p(\mathbf{w}|\mathbf{t}, \mathbf{X}, \lambda)$. The lower contour plots describe the standard Gaussian prior and the resulting unnormalised posterior using the same likelihood function as above.

Besides the sparsity promoting shape, another useful property of the univariate Laplace distribution is the possibility to present it as an infinite mixture of zero-mean Gaussian distributions with variances v_j distributed according to an exponential distribution $\mathcal{E}\left(v_j; \frac{1}{2\lambda^2}\right) = \frac{1}{2\lambda^2}\mathrm{e}^{-\frac{v_j}{2\lambda^2}}$, where $v_j > 0$ (Andrews and Mallows 1974):

$$p(w_j|\lambda) = \int_0^\infty \mathcal{N}\left(w_j; 0, v_j\right) \mathcal{E}\left(v_j; \frac{1}{2\lambda^2}\right) \mathrm{d}v_j. \tag{36}$$

Noticing that the exponential distribution is equal to the χ^2 distribution with two degrees of freedom, i.e., the distribution of the sum of squares of two independent standard Gaussian random variables (van Gerven et al. 2010), the decomposition can be written in an alternative form:

$$p(w_j|\lambda) = \int_{-\infty}^\infty \int_{-\infty}^\infty \mathcal{N}(w_j; 0, \mu_j^2 + \nu_j^2)\mathcal{N}(\mu_j; 0, \lambda^2)\mathcal{N}(\nu_j; 0, \lambda^2)\, \mathrm{d}\mu_j\, \mathrm{d}\nu_j. \tag{37}$$

These decompositions make it easier to computationally approximate the posterior distribution, and either one of the forms is utilised by each of the three approximate inference methods used in this work. One of the methods (see 5.1 Automatic Relevance Determination by Expectation Propagation), based on automatic relevance determination (ARD), actually takes the use of equation 36 even further by regarding the Gaussian auxiliary variances v_j as model hyperparameters representing the relevance of the corresponding feature in the model. By optimising these relevance hyperparameters, the ARD method enforces many of the weights to be zero and thus includes only the most relevant features in the final model.

Sparse solutions are favourable in neuroscience and multi-voxel pattern analysis, because too large amount of adjustable parameters compared to the amount of available observations may reduce the predictive performance and especially the neuroscientifical interpretability of the model (Rasmussen et al. 2012). Using a sparsity promoting prior, such as the Laplace prior with a small enough λ, may alleviate this problem by reducing the effect of irrelevant input features. However, even if the Laplace prior favours sparse solutions, a full Bayesian treatment always retains some uncertainty on the parameters and leads to a truly sparse posterior distribution only with an infinite amount of training data. Solutions with part of the weights exactly reducing to zero would require using a point estimate on the weight vector $\mathbf{w}$, such as the maximum a posteriori (MAP) estimate

$$\mathbf{w}_{\mathrm{MAP}} = \arg\max_{\mathbf{w}} \{p(\mathbf{w}|\mathbf{t}, \mathbf{X}, \lambda)\}, \tag{38}$$

instead of inferring the full posterior distribution. When using the Laplace prior, MAP estimation is equivalent to L_1-norm regularisation (Tibshirani 1996). The previously mentioned ARD approach can be regarded as a type II point estimate method, because it uses a marginal MAP estimate for the relevance hyperparameter vector $\mathbf{v} = (v_1, \dots, v_D)^{\mathsf{T}}$, leading as well to a truly sparse solution with many zero weights. The final posterior solution concerning the remaining non-zero weights, however, includes also the uncertainties on them, like with both of the other approximate inference methods used in this work.

5 Approximate Inference

Three different methods are used to carry out the approximate inference on the posterior distribution over the weights of the classifier: automatic relevance determination by expectation propagation (ARDEP), expectation propagation on the original Laplace prior (LAEP) and a Markov chain Monte Carlo method using the Gibbs sampler (MCMC). In the following, the abbreviations ARDEP, LAEP and MCMC stand for the particular algorithms used in this work, as a distinction from general concepts and other implementations. Both ARDEP and LAEP use an expectation propagation (EP) algorithm to approximate the posterior as a multivariate Gaussian distribution. The difference between these two methods is, that ARDEP decomposes the Laplace prior into a Gaussian scale mixture and optimises these scales by maximising their marginal posterior density. As a result, the solution produced by ARDEP will be truly sparse, with many of the features pruned out of the model. The smoother LAEP approximation, integrating over the original Laplace prior, becomes actually closer to the MCMC solution, which, as a widely acknowledged golden standard solution, should approach the accurate posterior of the original model, if enough samples are drawn from the posterior, but consumes more time than the EP methods. A proper value of the hyperparameter λ is determined separately for all the three methods by testing the models with different λ in a cross-validation scheme, where one subject at a time is removed from the training data.

5.1 Automatic Relevance Determination by Expectation Propagation

5.1.1 Automatic Relevance Determination

Automatic relevance determination is a commonly used Bayesian method for feature selection and sparse learning (MacKay 1994; Neal 1994). The key idea in ARD is to give the feature weights w_j, or more generally groups of them, independent zero-mean Gaussian priors

$$p(w_j|v_j) = \mathcal{N}(w_j; 0, v_j), \tag{39}$$

where the variances v_j are hyperparameters representing the relevance of the particular feature. This prior, often called the ARD prior, restricts the weights of irrelevant features from getting far from zero by controlling their variances. In the conventional ARD, the relevance hyperparameter vector $\mathbf{v} = (v_1, \ldots, v_D)^\mathsf{T}$ is optimised by maximising its marginal likelihood

$$p(\mathbf{t}|\mathbf{X}, \mathbf{v}) = \int_{\mathbf{w}} p(\mathbf{t}, \mathbf{w}|\mathbf{X}, \mathbf{v})\, \mathrm{d}\mathbf{w}, \tag{40}$$

obtained by integrating out the weight vector $\mathbf{w}$. This optimisation leads to a sparse $\mathbf{v}$ with many of the prior variances v_j reducing to zero, meaning that also the corresponding feature weights w_j are forced to be equal to zero. Ideally, this means that irrelevant features are automatically pruned out of the model, leading to a truly sparse posterior distribution.

5.1.2 ARDEP

The ARDEP algorithm implemented for this work is based on the algorithm introduced by Qi et al. (2004). To approximate the integral appearing in the expression of the marginal likelihood for a given relevance hyperparameter vector $\mathbf{v}$ (equation 40), they modify an algorithm from the expectation propagation family developed by Thomas Minka (2001). An EP run produces an approximation for the posterior distribution over the weights, given $\mathbf{v}$, and as a side product, offers also an approximation for the marginal likelihood. To find the optimal hyperparameter vector that maximises the marginal likelihood, they use a fast sequential updating scheme based on the analysis by Faul and Tipping (2002).

Maximising the marginal likelihood is equivalent to finding a maximum a posteriori estimate for $\mathbf{v}$ with a uniform prior over each hyperparameter v_j. My implementation is as well trying to find a MAP estimate for $\mathbf{v}$, but the uniform hyperprior is replaced by an exponential hyperprior according to

$$p(v_j|\lambda) = \mathcal{E}\left(v_j; \frac{1}{2\lambda^2}\right), \tag{41}$$

where λ is a constant scale parameter common for each v_j. The optimal $\mathbf{v}$ is now obtained by maximising the product of the marginal likelihood and the new hyperprior:

$$\mathbf{v}_{\mathrm{MAP}} = \arg\max_{\mathbf{v}}\{p(\mathbf{v}|\mathbf{t}, \mathbf{X}, \lambda)\} = \arg\max_{\mathbf{v}}\{p(\mathbf{t}|\mathbf{X}, \mathbf{v})p(\mathbf{v}|\lambda)\}. \tag{42}$$

By adjusting λ, the complexity of the model can be controlled to reduce overfitting to the training data. Selecting a small enough value for the hyperparameter λ favours small values of v_j, and thus limits the amount of features considered relevant in the model, which also significantly lightens the computation during the algorithm. Notice also, that when the exponential hyperprior is combined with the ARD prior, a scale mixture presentation of the Laplace prior (equation 36) is obtained. Consequently, ARDEP can be interpreted as one kind of an approximate solution for the model introduced in the previous chapter.

In the following two subsections, I briefly describe the implementation of the ARDEP algorithm without further compromises. At first, Qi's presentation of EP for the probit model with ARD prior is reformed in a computationally more efficient way. To confirm the validity of this EP application, I carefully derive it through in appendix A. After that, I renew the optimisation rules for the relevance hyperparameters, taking the additional hyperprior into account. These are derived in detail in appendix B, correcting also several misprints in the original paper by Qi et al. (2004). Finally, I present some practical modifications for ARDEP improving the applicability of the method to the fMRI data used in this work. In the last subsection, I also discuss alternative ways to select $\mathbf{v}$ from the configurations visited during the iterations.

5.1.3 Expectation Propagation

When looking at the equation 27 (p. 17), it is noticed that the likelihood $p(\mathbf{t}|\mathbf{w}, \mathbf{X})$ is a product of N simple terms $g_i(\mathbf{w}) = \Psi(t_i\mathbf{w}^{\mathsf{T}}\mathbf{x}_i)$ representing the effect of each observation. The expectation propagation algorithm used in ARDEP approximates these terms by unnormalised Gaussians

$$\tilde{g}_i(\mathbf{w}) = \varsigma_i \mathrm{e}^{-\frac{1}{2\sigma_i}(t_i \mathbf{w}^\mathsf{T} \mathbf{x}_i - \rho_i)^2}. \tag{43}$$

Since the posterior disribution $p(\mathbf{w}|\mathbf{t}, \mathbf{X}, \mathbf{v})$ is proportional to the product of the likelihood and the Gaussian ARD prior, the approximate posterior $\tilde{q}(\mathbf{w})$ becomes also Gaussian

$$p(\mathbf{w}|\mathbf{t}, \mathbf{X}, \mathbf{v}) \approx \tilde{q}(\mathbf{w}) = \mathcal{N}(\mathbf{w}; \mathbf{m}_w, \mathbf{V}_w) \tag{44}$$

with a covariance matrix

$$\mathbf{V}_w = (\mathbf{\Phi}^\mathsf{T} \mathbf{\Lambda}^{-1} \mathbf{\Phi} + \mathbf{V}^{-1})^{-1} \tag{45}$$

and a mean vector

$$\mathbf{m}_w = \mathbf{V}_w \mathbf{\Phi}^\mathsf{T} \mathbf{\Lambda}^{-1} \boldsymbol{\rho}, \tag{46}$$

where I denote $\mathbf{V} = \mathrm{diag}(\mathbf{v})$, $\mathbf{\Phi} = (t_1\mathbf{x}_1, \ldots, t_N\mathbf{x}_N)^\mathsf{T}$, $\boldsymbol{\rho} = (\rho_1, \ldots, \rho_N)^\mathsf{T}$ and $\mathbf{\Lambda} = \mathrm{diag}(\sigma_1, \ldots, \sigma_N)$. The parameters for the approximate likelihood terms $\tilde{g}_i(\mathbf{w})$ are efficiently computed by the following iterative procedure, derived in detail in appendix A:

EP (for the probit model with ARD prior)

Input: data matrix $\mathbf{\Phi}$, hyperparameter vector $\mathbf{v}$ including the ARD prior variances

Output: approximate likelihood term parameters $(\rho_i, \sigma_i, \varsigma_i)$ for $i = 1, \ldots, N$, approximate posterior parameters $\mathbf{m}_w$ and $\mathbf{V}_w$ for the probit model with ARD prior

```
1. Initialise:
    $\rho_i = 0$, $\sigma_i = \infty$ and $\varsigma_i = 1$ for all $i = 1, \ldots, N$
    $\mathbf{m}_w = 0$ and $\mathbf{V}_w = \mathrm{diag}(\mathbf{v})$
2. Repeat until $(\rho_i, \sigma_i, \varsigma_i)$ for all $i = 1, \ldots, N$ converge:
    For $i = 1, \ldots, N$:
        A. Compute scalars $a_i$ and $b_i$ corresponding to the mean and variance of the marginal distribution over linear transformation $f_i(\mathbf{w}) = \mathbf{w}^\mathsf{T} t_i \mathbf{x}_i$ using the current approximate posterior $\tilde{q}(\mathbf{w})$:
            $a_i = \mathbf{m}_w^\mathsf{T} t_i \mathbf{x}_i$
            $\mathbf{c}_i = \mathbf{V}_w t_i \mathbf{x}_i$
            $b_i = t_i \mathbf{x}_i^\mathsf{T} \mathbf{c}_i$
        B. Remove the approximate term $\tilde{g}_i(\mathbf{w})$ from $\tilde{q}(\mathbf{w})$ to obtain the leave-$i$-out posterior $\tilde{q}^{\setminus i}(\mathbf{w}) \propto \tilde{q}(\mathbf{w}) / \tilde{g}_i(\mathbf{w})$. Compute the parameters $a_i^{\setminus i}$ and $b_i^{\setminus i}$ for the corresponding marginal distribution over $f_i(\mathbf{w})$:
```

$$b_i^{\setminus i} = b_i + \tfrac{b_i^2}{\sigma_i - b_i}$$
$$a_i^{\setminus i} = a_i + \tfrac{b_i^{\setminus i}(a_i - \rho_i)}{\sigma_i}$$
$$\mathbf{c}_i^{\setminus i} = \mathbf{c}_i \left(1 + \tfrac{b_i}{\sigma_i - b_i}\right)$$

C. Replace the removed approximate term $\tilde{g}_i(\mathbf{w})$ with the accurate term $g_i(\mathbf{w})$ to obtain a target posterior approximation $\hat{q}(\mathbf{w}) \propto g_i(\mathbf{w})\tilde{q}^{\setminus i}(\mathbf{w})$, and choose then the new term approximation $\tilde{g}_i{}^*(\mathbf{w})$ to minimise the Kullback-Leibler divergence between $\hat{q}(\mathbf{w})$ and the new posterior approximation $\tilde{q}^*(\mathbf{w}) \propto g_i^*(\mathbf{w})\tilde{q}^{\setminus i}(\mathbf{w})$. For updating $\tilde{q}(\mathbf{w})$ and $\tilde{g}_i(\mathbf{w})$, compute the following auxiliary variables related to the target distribution $\hat{q}(\mathbf{w})$:

$$z_i = \tfrac{a_i^{\setminus i}}{\sqrt{1 + b_i^{\setminus i}}}$$
$$\alpha_i = \tfrac{\mathcal{N}(z_i;0,1)}{\Psi(z_i)\sqrt{1 + b_i^{\setminus i}}}$$
$$a_i^* = a_i^{\setminus i} + \alpha_i b_i^{\setminus i}$$

D. Update parameters for $\tilde{q}(\mathbf{w}) \leftarrow \tilde{q}^*(\mathbf{w})$:

$$\mathbf{V}_w \leftarrow \mathbf{V}_w^* = \mathbf{V}_w + \left(\tfrac{1}{\sigma_i - b_i} - \tfrac{\alpha_i(a_i^* + \alpha_i)}{1 + b_i^{\setminus i}} \left(1 + \tfrac{b_i}{\sigma_i - b_i}\right)^2\right) \mathbf{c}_i \mathbf{c}_i^\mathsf{T}$$
$$\mathbf{m}_w \leftarrow \mathbf{m}_w^* = \mathbf{m}_w + \left(\tfrac{a_i - \rho_i}{\sigma_i} + \alpha_i\right) \mathbf{c}_i^{\setminus i}$$

E. Update parameters for $\tilde{g}_i(\mathbf{w}) \leftarrow \tilde{g}_i{}^*(\mathbf{w})$:

$$\sigma_i \leftarrow \sigma_i^* = \tfrac{1 + b_i^{\setminus i}}{\alpha_i(a_i^* + \alpha_i)} - b_i^{\setminus i}$$
$$\rho_i \leftarrow \rho_i^* = a_i^{\setminus i} + \alpha_i b_i^{\setminus i} + \alpha_i \sigma_i^* = a_i^* + \alpha_i \sigma_i^*$$
$$\varsigma_i \leftarrow \varsigma_i^* = \Psi(z_i)\sqrt{1 + \tfrac{b_i^{\setminus i}}{\sigma_i^*}}\mathrm{e}^{\frac{1}{2}\alpha_i \frac{1 + b_i^{\setminus i}}{a_i^* + \alpha_i}}$$

After convergence, an approximation for an expression that is directly proportional to the marginal posterior over $\mathbf{v}$ can be written with respect to the obtained parameters by multiplying the approximation of the marginal likelihood $p(\mathbf{t}|\mathbf{X}, \mathbf{v})$, derived in appendix A (equation A37), by the hyperprior $p(\mathbf{v}|\lambda)$:

$$\begin{aligned} p(\mathbf{v}|\mathbf{t}, \mathbf{X}, \lambda) &\propto p(\mathbf{t}|\mathbf{X}, \mathbf{v})p(\mathbf{v}|\lambda) \\ &\approx \left(\prod_{i=1}^{N} \varsigma_i\right) |\mathbf{V}|^{-\frac{1}{2}} \mathrm{e}^{-\frac{1}{2}(\boldsymbol{\rho}^\mathsf{T}\boldsymbol{\Lambda}^{-1}\boldsymbol{\rho} - \mathbf{m}_w^\mathsf{T}\mathbf{V}_w^{-1}\mathbf{m}_w)} |\mathbf{V}_w|^{\frac{1}{2}} \prod_{j=1}^{D} \frac{1}{2\lambda^2} \mathrm{e}^{-\frac{v_j}{2\lambda^2}}. \end{aligned} \tag{47}$$

A numerical value for the above expression can be computed efficiently as

$$\begin{aligned} \ln\left(p(\mathbf{t}|\mathbf{X}, \mathbf{v})p(\mathbf{v}|\lambda)\right) &\approx \sum_{i=1}^{N} \ln \varsigma_i - \frac{1}{2}\sum_{j=1}^{D} \ln v_j - \ln|\mathbf{L}_1| \\ &+ \frac{1}{2}\left((\mathbf{L}_1^\mathsf{T}\mathbf{m}_w)^\mathsf{T}(\mathbf{L}_1^\mathsf{T}\mathbf{m}_w) - \sum_{i=1}^{N} \frac{\rho_i^2}{\sigma_i}\right) - \frac{1}{2\lambda^2}\sum_{j=1}^{D} v_j - D\ln(2\lambda^2), \end{aligned} \tag{48}$$

where $\mathbf{L}_1\mathbf{L}_1^\mathsf{T}$ is the Cholesky decomposition (Press 2002, pp. 99–101) of $\mathbf{V}_w^{-1} = \mathbf{\Phi}^\mathsf{T}\mathbf{\Lambda}^{-1}\mathbf{\Phi} + \mathbf{V}^{-1}$.

5.1.4 Fast Sequential Optimisation of Relevance Hyperparameters

As described in the previous subsection, each EP run produces an approximation for the posterior distribution over the weights $\mathbf{w}$, when the hyperparameter vector $\mathbf{v}$ and the data matrix $\mathbf{\Phi} = (t_1\mathbf{x}_1, \ldots, t_N\mathbf{x}_N)^\mathsf{T}$ have been given as input. Thus, to figure out the final posterior approximation, we only have to find the optimal hyperparameter vector $\mathbf{v}$ according to some criterion. In ARDEP, the optimal $\mathbf{v}$ is defined as the MAP estimate $\mathbf{v}_{\text{MAP}}$, which maximises the posterior density over $\mathbf{v}$. This maximum point is searched for by sequentially maximising the approximate expression of $p(\mathbf{t}|\mathbf{X}, \mathbf{v})p(\mathbf{v}|\lambda)$ produced by EP with respect to each v_j and running a new EP with the obtained $\mathbf{v}$. After v_j for all $j = 1, \ldots, D$ have converged, the result of the last EP run, i.e., the one with the optimal input vector $\mathbf{v}_{\text{MAP}}$, is selected as the final posterior approximation.

The update rules for a single hyperparameter v_j are derived in detail in appendix B by analysing the approximate expression of $p(\mathbf{t}|\mathbf{X}, \mathbf{v})p(\mathbf{v}|\lambda)$ in equation 47 with respect to v_j. Denoting the m^{th} column of the data matrix $\mathbf{\Phi}$ as $\boldsymbol{\phi}_m = (t_1[\mathbf{x}_1]_m, \ldots, t_N[\mathbf{x}_N]_m)^\mathsf{T}$ and separating the terms dependent on v_j from the logarithm $\mathcal{L}(\mathbf{v})$ of the approximate $p(\mathbf{t}|\mathbf{X}, \mathbf{v})p(\mathbf{v}|\lambda)$ lead to

$$\mathcal{L}(\mathbf{v}) = \mathcal{L}(\mathbf{v}^{\setminus j}) - \frac{1}{2}\ln(1 + r_j v_j) + \frac{h_j^2}{2}\frac{v_j}{(1 + r_j v_j)} - \frac{1}{2\lambda^2}v_j, \tag{49}$$

where $r_j = \boldsymbol{\phi}_j^\mathsf{T}(\mathbf{\Omega}^{\setminus j})^{-1}\boldsymbol{\phi}_j$, $h_j = \boldsymbol{\phi}_j^\mathsf{T}(\mathbf{\Omega}^{\setminus j})^{-1}\boldsymbol{\rho}$ and $\mathbf{\Omega}^{\setminus j} = \mathbf{\Lambda} + \sum_{m \neq j}\boldsymbol{\phi}_m v_m \boldsymbol{\phi}_m^\mathsf{T}$. The maximum point v_j^* of the above expression with respect to $v_j \geq 0$ depends on the sign of $\eta_j = h_j^2 - r_j - \frac{1}{\lambda^2}$ as follows:

$$v_j^* = \begin{cases} -\frac{\lambda^2}{2} - \frac{1}{r_j} + \sqrt{\frac{\lambda^4}{4} + \frac{\lambda^2 h_j^2}{r_j^2}}, & \text{if } \eta_j > 0, \\ 0, & \text{if } \eta_j \leq 0. \end{cases} \tag{50}$$

When maximising the posterior density over $\mathbf{v}$, many of the hyperparameters v_j tend to reduce to zero, which is equivalent to a model with the corresponding features j removed. Thus, in practice, each EP is run with sparsified input hyperparameter vector $\bar{\mathbf{v}}$ (or $\bar{\mathbf{V}}$ as a diagonal matrix form) and data matrix $\bar{\mathbf{\Phi}}$ including only features $m \in F$, where $F = \{m : v_m > 0\}$. The existence of the extra features in the original model has to be taken into account only when computing a value for $p(\mathbf{t}|\mathbf{X}, \mathbf{v})p(\mathbf{v}|\lambda)$, since it depends on the total feature amount D through the hyperprior. The reduction of dimension speeds up also the hyperparameter updates, since scalars r_j and h_j can be computed efficiently by using the current value of v_j and scalars R_j and H_j written with respect to the sparsified posterior parameters $\bar{\mathbf{m}}_w$ and $\bar{\mathbf{V}}_w$. Ignoring the practical modifications that are discussed in the following subsection, the ARDEP algorithm can now be summarised according to the following pseudocode:

```
**ARDEP** (without practical modifications)

**Input:** data matrix $\mathbf{\Phi}$, Laplace prior hyperparameter $\lambda$

**Output:** sparsified approximate posterior parameters $\bar{\mathbf{m}}_w$ and $\bar{\mathbf{V}}_w$ for the probit model with Laplace prior

1. Initialize the hyperparameter vector $\mathbf{v}$ to the mean of the hyperprior $p(\mathbf{v}|\lambda)$:
     $v_j = 2\lambda^2$ for all $j = 1, \dots, D$
     $j \in F$ for all $j = 1, \dots, D$
2. Repeat until $v_j$ for all $j = 1, \dots, D$ converge:
     A. Run EP (for the probit model with ARD prior):
          Input: sparsified $\bar{\mathbf{\Phi}}$, $\bar{\mathbf{v}}$ including only features $j \in F$
          Output: $\boldsymbol{\rho}$, $\mathbf{\Lambda}$, $\bar{\mathbf{m}}_w$, $\bar{\mathbf{V}}_w$
     B. For $j = 1, \dots, D$:
          I. Compute:
               $R_j = \boldsymbol{\phi}_j^\mathsf{T} \mathbf{\Lambda}^{-1} \boldsymbol{\phi}_j - \boldsymbol{\phi}_j^\mathsf{T} \mathbf{\Lambda}^{-1} \bar{\mathbf{\Phi}} \bar{\mathbf{V}}_w \bar{\mathbf{\Phi}}^\mathsf{T} \mathbf{\Lambda}^{-1} \boldsymbol{\phi}_j$
               $H_j = \boldsymbol{\phi}_j^\mathsf{T} \mathbf{\Lambda}^{-1} \boldsymbol{\rho} - \boldsymbol{\phi}_j^\mathsf{T} \mathbf{\Lambda}^{-1} \bar{\mathbf{\Phi}} \bar{\mathbf{m}}_w$
          II. Compute:
               $r_j = \frac{R_j}{1 - v_j R_j}$
               $h_j = \frac{H_j}{1 - v_j R_j}$
               $\eta_j = h_j^2 - r_j - \frac{1}{\lambda^2}$
          III. Update $v_j$:
               $v_j \leftarrow v_j^* = \begin{cases} -\frac{\lambda^2}{2} - \frac{1}{r_j} + \sqrt{\frac{\lambda^4}{4} + \frac{\lambda^2 h_j^2}{r_j^2}}, & \text{if } \eta_j > 0 \\ 0, & \text{if } \eta_j \leq 0 \end{cases}$
          IV. Update $F$:
               $j \in F$, if $v_j > 0$
               $j \notin F$, if $v_j = 0$
```

5.1.5 Practical Modifications

In practice, the converging of the sequential optimisation of the hyperparameters may be hopelessly slow for large multidimensional models with many correlated features. When applied to the fMRI data used in this work, the algorithm described above is not able to find the maximum of $p(\mathbf{v}|\mathbf{t}, \mathbf{X}, \lambda)$ with respect to the whole hyperparameter vector $\mathbf{v}$, but insted alternates between feature configurations by setting a group of parameters to zero and taking them back into the model during the following iteration. The reason for this behaviour is, that each hyperparameter v_j is optimised separately by keeping the others constant, without updating the expression of $p(\mathbf{t}|\mathbf{X}, \mathbf{v})p(\mathbf{v}|\lambda)$ according to the change of $\mathbf{v}$ within an iteration. The correct way would be to rerun EP after each individual hyperparameter update, but this would be too slow, since it would cost D EP runs at each iteration of $\mathbf{v}$. In this work, the problem is solved by update rule modifications that damp the change of v_j (in item III in the algorithm description) in half, which significantly aids the convergence and speeds up the algorithm.

The drawback of damping is, that it prevents the hyperparameters from ever becoming exactly zero, thus keeping all the features included in the model. With a large amount of features, running EP gets slow, which is why the model has to be pruned by limiting the amount of features and removing the ones with the smallest v_j. In this work, the amount of features is limited in 300 features out of the total amount $D = 707$. For tight Laplace priors with a small value of the scale hyperparameter λ, the limiting is necessary also in the sense that the optimisation procedure may get rid of all the features, if the algorithm is initialised by a constant v_j for all $j = 1, \ldots, D$. To appropriately choose the initial 300 features, a preliminary EP with a small λ (here $2\lambda^2 = 10^{-12}$) is run and the features are ordered by $\hat{v}_j$, defined in appendix B (equation B13), accepting also negative values.

5.1.6 Alternative Criteria for Selection of the Final Relevance Hyperparameter Vector

According to the original paper by Qi et al. (2004), the ARD framework of maximising the marginal likelihood suffers from a mixture of two kinds of overfitting. Firstly, a large amount of features included in the model may lead to overcomplicated classifiers and thus to worse generalisability and predictive performance. At the same time, some generally relevant features may still be pruned out of the model, in case the model happens to fit the data as well without them. This overfitting typically means, that during the hyperparameter optimisation, the true predictive performance increases in the beginning, but at some point starts to decrease towards the converged level.

In ARDEP, the effect of overfitting is minimised through the hyperprior by selecting a proper value of hyperparameter λ using cross-validation (Stone 1974) before constructing the final model (see 5.4 Hyperparameter Selection). To examine the progress of the true predictive performance during the relevance hyperparameter optimisation with the fMRI data, I apply a similar cross-validation scheme, where one subject at a time is left out of the training data set and the obtained model is tested with the removed subject according to the measures introduced in equations 31 and 33. Denote the leave-S_k-out training data as $\mathbf{t}^{\backslash S_k}$ and $\mathbf{X}^{\backslash S_k}$, where the observations $i \in S_k$ belonging to subject k have been removed and gathered in $\mathbf{t}_{S_k}$ and $\mathbf{X}_{S_k}$. The leave-one-subject-out CA and leave-one-subject-out MLPP are obtained by averaging over test subjects $k = 1, \ldots, K$:

$$\mathrm{CA}_{\mathrm{loso}} = \frac{1}{K} \sum_{k=1}^{K} \mathrm{CA}(\mathbf{t}_{S_k}, \mathbf{X}_{S_k} | \mathbf{t}^{\backslash S_k}, \mathbf{X}^{\backslash S_k}, \lambda), \tag{51}$$

$$\mathrm{MLPP}_{\mathrm{loso}} = \frac{1}{K}\sum_{k=1}^{K}\mathrm{MLPP}(\mathbf{t}_{S_k}, \mathbf{X}_{S_k}|\mathbf{t}^{\setminus S_k}, \mathbf{X}^{\setminus S_k}, \lambda). \tag{52}$$

Figure 3 presents an example of the progress of these average measures during the relevance hyperparameter optimisation as a function of the number of iteration rounds. Even if the figure is obtained by using an optimally selected hyperparameter λ, it indeed seems that the model loses some of its predictive power, while the algorithm converges towards a stable, sparse relevance hyperparameter vector $\mathbf{v}_{\mathrm{MAP}}$. The effect appears larger with smaller and larger values of λ, but apparently the adjustment of the hyperparameter does not fully remove the overfitting.

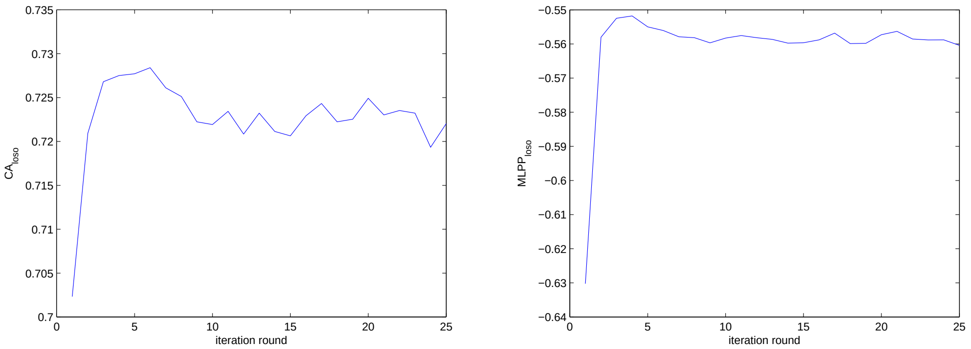

Figure 3: An example of the average progress of $\mathrm{CA}(\mathbf{t}_{S_k}, \mathbf{X}_{S_k}|\mathbf{t}^{\setminus S_k}, \mathbf{X}^{\setminus S_k}, \lambda)$ and $\mathrm{MLPP}(\mathbf{t}_{S_k}, \mathbf{X}_{S_k}|\mathbf{t}^{\setminus S_k}, \mathbf{X}^{\setminus S_k}, \lambda)$ over a cross-validation scheme, where each subject $k = 1, \ldots, K$ at a time is left out of the training data, presented as a function of the number of iteration rounds in ARDEP.

To amend the algorithm due to this unfavourable reduction of predictive power, Qi et al. suggest keeping track of an estimate for predictive performance provided by each EP during the optimisation precedure. In the end, instead of choosing the relevance hyperparameter vector producing the maximum marginal likelihood, they select the one with the highest estimated predictive performance for the final posterior approximation. In particular, they try to estimate the corresponding leave-one-out CA and leave-one-out MLPP, defined by using a cross-validation scheme, where only one observation at a time is left out of the training data set, but without carrying out the actual cross-validation. These estimates for $\mathrm{CA}_{\mathrm{loo}}$ and $\mathrm{MLPP}_{\mathrm{loo}}$ are obtained by using the auxiliary variables z_i obtained during the site updates in EP (see item C in the algorithm description on p. 24), according to equations A39 and A40 in appendix A:

$$\mathrm{\tilde{C}A}_{\mathrm{loo}} = \frac{1}{N}\sum_{i=1}^{N}\mathcal{H}(z_i), \tag{53}$$

$$\mathrm{ML\tilde{P}P}_{\mathrm{loo}} = \frac{1}{N}\sum_{i=1}^{N}\ln\Psi(z_i). \tag{54}$$

The validity of removing only one observation at a time is questionable for data with several observations from the same subject, like the fMRI data used in this work. Because the observations belonging to the same subject may be correlated, the training data still includes information about the removed observation. Thus, a better way to simulate testing with new data would be to leave all the observations of one subject at a time out of the training set, as described above, and use the leave-one-subject-out measures $\mathrm{CA}_{\mathrm{loso}}$ and $\mathrm{MLPP}_{\mathrm{loso}}$. The selection of the hyperparameter λ is carried out by this kind of cross-validation. Also for this reason, it is natural to replace the EP estimates $\widetilde{\mathrm{CA}}_{\mathrm{loo}}$ and $\widetilde{\mathrm{MLPP}}_{\mathrm{loo}}$ with the corresponding leave-one-subject-out estimates:

$$\widetilde{\mathrm{CA}}_{\mathrm{loso}} = \frac{1}{N} \sum_{k=1}^{K} \sum_{i \in S_k} \mathcal{H}(z_i^{\setminus S_k}), \tag{55}$$

$$\widetilde{\mathrm{MLPP}}_{\mathrm{loso}} = \frac{1}{N} \sum_{k=1}^{K} \sum_{i \in S_k} \ln \Psi(z_i^{\setminus S_k}). \tag{56}$$

Unlike for the leave-one-out estimates, the corresponding auxiliary variables $z_i^{\setminus S_k}$ for the leave-one-subject-out estimates require some extra computation as derived in appendix A (equations A41–A45).

Even though using the EP estimates described above sounds tempting, in practice they do not seem to work with the fMRI data. Figure 4 presents an example of the progress of these estimates during the relevance hyperparameter optimisation with all observations included in the training data. If the estimates performed properly, $\widetilde{\mathrm{CA}}_{\mathrm{loso}}$ and $\widetilde{\mathrm{MLPP}}_{\mathrm{loso}}$ should correspond to the cross-validation measures $\mathrm{CA}_{\mathrm{loso}}$ and $\mathrm{MLPP}_{\mathrm{loso}}$ presented in figure 3. However, the EP estimates do not show a clear maximum after a few iteration rounds, but instead keep the maximum levels after reaching it along the maximisation of the posterior of $\mathbf{v}$. The shape of the true predictive performance is followed only, when λ is so small that the hyperprior enforces almost all features to be pruned out, causing that the model cannot fit even the training data. A possible explanation for this kind of behaviour would be, that removing approximate likelihood terms from the posterior approximation

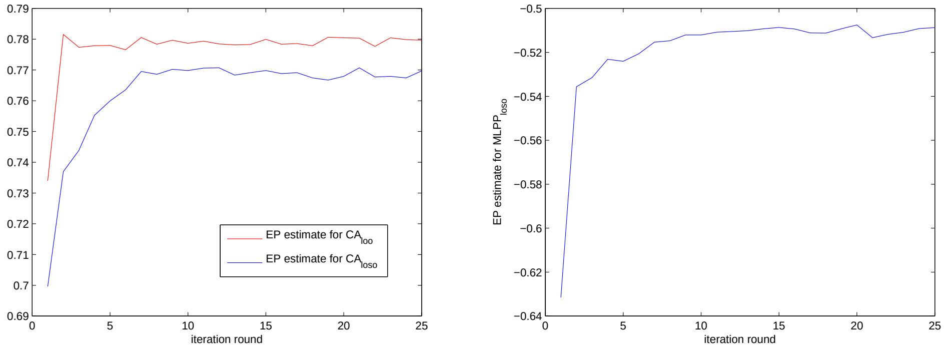

Figure 4: An example of the progress of EP estimates $\widetilde{\mathrm{CA}}_{\mathrm{loo}}$, $\widetilde{\mathrm{CA}}_{\mathrm{loso}}$ and $\widetilde{\mathrm{MLPP}}_{\mathrm{loso}}$ presented as a function of the number of iteration rounds in ARDEP.

does not fully remove the effect of the corresponding observations on the cavity approximations, which would lead to overoptimistic estimates. Since the EP estimates do not follow the true predictive performance, I hold on to the original criterion of choosing the MAP estimate $\mathbf{v}_{\text{MAP}}$ as the final relevance hyperparameter for the primary ARDEP. For reference, I denote these alternative ARDEP algorithms as loo-ARDEP, loso-ARDEP and mlpp-ARDEP.

5.2 Expectation Propagation on Laplace Prior

For the LAEP solution, I use an algorithm implemented in the multivariate module of the FieldTrip toolbox (Oostenveld et al. 2011) by Marcel van Gerven. His algorithm is developed for a similar binary classification task as the one studied in this work, but it accepts also multivariate Laplace priors with desired couplings between the weights (van Gerven et al. 2010). By using a diagonal prior covariance matrix with a constant hyperparameter for all the weights, the model reduces to the one used in this work, with the exception that the activation function Ψ of the probit model is replaced by the logistic activation function l^{-1}:

$$l^{-1}(f_i) = \frac{1}{1 + \mathrm{e}^{-f_i}}. \tag{57}$$

By matching the derivatives at the origin, these two functions become practically similar, leading to an approximation $l^{-1}(f_i) \approx \Psi(\sqrt{\frac{\pi}{8}} f_i)$ (Bishop 2006, p. 219). Since the scale of the linear transformation $f_i = t_i \mathbf{w}^{\mathsf{T}} \mathbf{x}_i$ passed to the activation function is proportional to λ^2, the logit model corresponds approximately to the probit model after multiplying the scale hyperparameter corresponding to λ^2 by a factor $\frac{8}{\pi}$:

$$\theta_\lambda = \frac{8}{\pi}\lambda^2. \tag{58}$$

For posterior inference, LAEP utilises a scale mixture of the Laplace prior in equation 37, which is here presented with respect to the scaled hyperparameter θ_λ:

$$p(w_j|\lambda) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathcal{N}(w_j; 0, \mu_j^2 + \nu_j^2)\mathcal{N}(\mu_j; 0, \theta_\lambda)\mathcal{N}(\nu_j; 0, \theta_\lambda)\, \mathrm{d}\mu_j\, \mathrm{d}\nu_j. \tag{59}$$

Recall from the previous subsection, that the ARDEP algorithm uses a corresponding scale mixture by regarding the scale parameters $v_j = \mu_j^2 + \nu_j^2$ as relevance hyperparameters to be optimised by maximising their marginal posterior density. As a distinction to ARDEP, LAEP does not use point estimates for the scale parameters, but instead approximates the posterior distribution over the whole set of latent variables $(\mathbf{w},\boldsymbol{\mu},\boldsymbol{\nu})$:

$$p(\mathbf{w}, \boldsymbol{\mu}, \boldsymbol{\nu}|\mathbf{X}, \mathbf{t}, \lambda) \propto \left(\prod_{j=1}^{D} \mathcal{N}(w_j; 0, \mu_j^2 + \nu_j^2)\mathcal{N}(\mu_j; 0, \theta_\lambda)\mathcal{N}(\nu_j; 0, \theta_\lambda)\right) \prod_{i=1}^{N} l^{-1}(t_i \mathbf{w}^{\mathsf{T}} \mathbf{x}_i).$$

The posterior distribution over the weights $\mathbf{w}$ is then obtained by marginalising $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ out from the joint posterior approximation. Because of this marginalisation, the result is much smoother than in ARDEP.

For the approximations, LAEP uses an expectation propagation algorithm with a similar idea as in the one used in ARDEP. The likelihood terms $g_i(\mathbf{w}) = l^{-1}(t_i\mathbf{w}^\mathsf{T}\mathbf{x}_i)$ are approximated by

$$\tilde{g}_i(\mathbf{w}) = \mathrm{e}^{\mathbf{w}^\mathsf{T}\mathbf{x}_i\varrho_i^g - \frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{x}_i\kappa_i^g\mathbf{x}_i^\mathsf{T}\mathbf{w}}, \tag{60}$$

where ϱ_i^g and κ_i^g are scalar parameters. Respectively, the auxiliary variable terms $f_j(\boldsymbol{\omega}_j) = \mathcal{N}(w_j; 0, \mu_j^2 + \nu_j^2)$ are approximated by

$$\tilde{f}_j(\boldsymbol{\omega}_j) = \mathrm{e}^{\boldsymbol{\omega}_j^\mathsf{T}\boldsymbol{\varrho}_j^f - \frac{1}{2}\boldsymbol{\omega}_j^\mathsf{T}\boldsymbol{\kappa}_j^f\boldsymbol{\omega}_j}, \tag{61}$$

where $\boldsymbol{\omega}_j = (w_j, \mu_j, \nu_j)^\mathsf{T}$ and $\boldsymbol{\varrho}_j^f$ and $\boldsymbol{\kappa}_j^f$ are the corresponding three-dimensional parameter vector and matrix. Since the prior terms for μ_j and ν_j are already in a proper Gaussian form, the whole posterior can now be approximated as a multivariate Gaussian distribution. As in the EP algorithm used in ARDEP, the term approximations are found by an iterative procedure that updates their parameters term by term to minimise the Kullback-Leibler divergence between a target distribution that uses the accurate term and the new posterior approximation. For a detailed description and derivation, I refer to the original paper by van Gerven et al. (2010).

5.3 Markov Chain Monte Carlo

5.3.1 Posterior Simulation

Monte Carlo simulation methods provide a completely different approach for approximate inference compared to the previous EP methods. Instead of estimating parameters for some tractable parameteric distributions, they generate random samples to represent the posterior distribution. The more samples drawn, the more the distribution of the samples should resemble the accurate solution for the original model. After generating samples from the posterior distribution of the weights, also the predictive distribution can be easily simulated and summarised by using the obtained samples with the new inputs.

When direct simulation from the posterior is not feasible, one solution is to use Markov chain Monte Carlo methods with each draw depending on the previous one. Even if the early samples in the Markov chain are dependent on the initialisation, the chain can be constructed to converge towards the target distribution along the procedure. One of the most widely used Markov chain Monte Carlo methods is the Gibbs sampler (Geman and Geman 1984), which is specifically intended for multivariate distributions. The Gibbs sampler simulates a target distribution of several parameters by sequentially drawing each subset of parameters conditional on the others. Thus, one cycle through all the parameters forms one component in the Markov chain.

5.3.2 MCMC

The MCMC algorithm implemented for this work is a modified version of an algorithm implemented by Mark Schmidt (2006) for sampling from the probit model based on the article by Albert and Chib (1993). It uses the idea of the Gibbs sampler to simulate the posterior distribution over the weights $\mathbf{w}$ and auxiliary variables $\mathbf{u} = (u_1, \ldots, u_N)^\mathsf{T}$, where u_i is the latent variable generating class t_i of observation i according to equations 23 and 24. To apply the Laplace prior to the model, I extend the algorithm to sample also over $\mathbf{v} = (v_1, \ldots, v_D)^\mathsf{T}$, where v_j is the scale parameter of the Laplace prior decomposition in equation 36. Note here, that even though this is the same scale mixture as utilised by ARDEP, v_j is here treated as a latent variable, like the scale parameters μ_j and ν_j in LAEP. Since both MCMC and LAEP preserve the original Laplace prior without further sparsification of features, they produce smoother solutions than ARDEP, which optimises $\mathbf{v}$ by a point estimate.

In MCMC, sampling over the latent variable u_i replaces the likelihood term. Thus, by using the scale mixture for the Laplace prior, the posterior distribution over $\mathbf{u}$, $\mathbf{w}$ and $\mathbf{v}$ can be written as

$$p(\mathbf{u}, \mathbf{w}, \mathbf{v} | \mathbf{X}, \mathbf{t}, \lambda) \propto p(\mathbf{t}|\mathbf{u}) p(\mathbf{u}|\mathbf{w}, \mathbf{X}) p(\mathbf{w}|\mathbf{v}) p(\mathbf{v}|\lambda). \tag{62}$$

Since $\mathbf{u}$ appears only in the first two terms, the posterior distribution over $\mathbf{u}$ conditional on $\mathbf{w}$ and $\mathbf{v}$ becomes

$$p(\mathbf{u}|\mathbf{w}, \mathbf{v}, \mathbf{X}, \mathbf{t}, \lambda) \propto p(\mathbf{t}|\mathbf{u}) p(\mathbf{u}|\mathbf{w}, \mathbf{X}) = \prod_{i=1}^{N} p(t_i|u_i) \mathcal{N}(u_i; \mathbf{w}^\mathsf{T}\mathbf{x}_i, 1). \tag{63}$$

Since $p(t_i|u_i)$ with a fixed t_i simply allows u_i to get only values with the sign t_i, the conditional posterior over u_i is a standard Gaussian distribution centered at $\mathbf{w}^\mathsf{T}\mathbf{x}_i$ and truncated from the origin. Truncated distributions could obviously be simulated by drawing samples from the complete distribution until a sample from the acceptable range is obtained. This is, however, computationally too expensive, when the acceptance probability is low. Thus, to sample u_i, MCMC uses a more efficient accept-reject algorithm introduced by Christian Robert (1995), based on sampling from an optimal envelope distribution.

Also the weight vector $\mathbf{w}$ appears only in two terms in the joint posterior. Thus, the posterior distribution over $\mathbf{w}$ conditional on $\mathbf{u}$ and $\mathbf{v}$ becomes Gaussian according to

$$\begin{aligned} p(\mathbf{w}|\mathbf{u}, \mathbf{v}, \mathbf{X}, \mathbf{t}, \lambda) \propto p(\mathbf{u}|\mathbf{w}, \mathbf{X}) p(\mathbf{w}|\mathbf{v}) &= \left(\prod_{i=1}^{N} \mathcal{N}(u_i; \mathbf{w}^\mathsf{T}\mathbf{x}_i, 1) \right) \prod_{j=1}^{D} \mathcal{N}(w_j; 0, v_j) \\ &\propto \mathcal{N}(\mathbf{w}; \mathbf{V}_G \mathbf{X}^\mathsf{T} \mathbf{u}, \mathbf{V}_G), \end{aligned} \tag{64}$$

where $\mathbf{V}_G = (\mathrm{diag}(\mathbf{v}) + \mathbf{X}^\mathsf{T}\mathbf{X})^{-1}$.

The posterior distribution over $\mathbf{v}$ conditional on $\mathbf{u}$ and $\mathbf{w}$, however, turns out to be of a more inconvenient form:

$$p(\mathbf{v}|\mathbf{u}, \mathbf{w}, \mathbf{X}, \mathbf{t}, \lambda) \propto p(\mathbf{w}|\mathbf{v}) p(\mathbf{v}|\lambda) = \prod_{j=1}^{D} \mathcal{N}(w_j; 0, v_j) \mathcal{E}\left(v_j; \frac{1}{2\lambda^2} \right). \tag{65}$$

For convenience, the sampling is carried out over the inverse of v_j. If v_j follows a gamma distribution, v_j^{-1} follows an inverse gamma distribution with the same parameters. Since the exponential distribution is equal to the gamma distribution with the unit scale, the conditional posterior over v_j^{-1} becomes

$$p(v_j^{-1}|\mathbf{u},\mathbf{w},\mathbf{X},\mathbf{t},\lambda) \propto \mathcal{N}\left(w_j; 0, \frac{1}{v_j^{-1}}\right)\text{Inv}-\Gamma\left(v_j^{-1}; 1, \frac{1}{2\lambda^2}\right)$$
$$\propto (v_j^{-1})^{-\frac{3}{2}}\mathrm{e}^{-\frac{w_j^2}{2}v_j^{-1}-\frac{1}{2\lambda^2}(v_j^{-1})^{-1}}. \qquad (66)$$

To sample from the above distribution, MCMC uses a method called slice sampling (Neal 2003), which is applicable for distributions of almost any form. Slice sampling utilises the fact that sampling from a probability distribution is equivalent to sampling from the area under the curve of its density function $y = f(x)$. As a first step, a vertical value y^* is sampled uniformly from the interval $[0, f(x^*)]$ according to the previous sample x^*. The value of y^* assigns then a horizontal slice including the values of x that meet the requirement of $f(x) \geq y^*$, and the new value of x^* is sampled uniformly from this horizontal slice. The efficient implementation of the slice sampling used by MCMC belongs to the GPstuff toolbox (Vanhatalo et al. 2011).

After sequentially sampling from the posterior distributions over $\mathbf{z}$, $\mathbf{w}$ and $\mathbf{v}$ conditional on the latest samples, it is important to check that the Markov chain has converged to the desired distribution (Gelman et al. 2004, pp. 294–299). To suppress the effect of the initial values of $\mathbf{w}$ and $\mathbf{v}$, the early samples are discarded as a burn-in. Since the remaining chain of samples is still more or less autocorrelated, the effective number of samples is much smaller than the total amount drawn. Thus, the Markov chain can be also thinned by skipping a constant amount of samples after each one that is selected into the final set. In this work, I use a burn-in of 500 and a thinning interval of 10 samples. To sum up the whole MCMC algorithm, I present the sampling procedure as the following pseudocode:

```
MCMC

Input: data matrix $\mathbf{X}$, label vector $\mathbf{t}$, Laplace prior hyperparameter $\lambda$

Output: 1000 samples of $\mathbf{w}$ drawn approximately from the posterior distribution $p(\mathbf{w}|\mathbf{X},\mathbf{t},\lambda)$ according to the probit model with Laplace prior

1. Initialise $\mathbf{w}$ to zero and $\mathbf{v}$ to the mean of $p(\mathbf{v}|\lambda)$:

     $w_j^0 = 0$ and $v_j^0 = 2\lambda^2$ for all $j = 1, \ldots, D$

2. For $s = 1, \ldots, 10500$

     A. Sample $\mathbf{u}^s$ from $p(\mathbf{u}|\mathbf{w}^{s-1}, \mathbf{v}^{s-1}, \mathbf{X}, \mathbf{t}, \lambda)$:
          For $i = 1, \ldots, N$:
               Draw $z > 0$, so that $p(z) \propto \mathcal{N}(z; t_i\mathbf{w}^\mathsf{T}\mathbf{x}_i, 1)$.
               $u_i^s = t_i|z|$
```

```
    B. Sample $\mathbf{w}^s$ from $p(\mathbf{w}|\mathbf{u}^s, \mathbf{v}^{s-1}, \mathbf{X}, \mathbf{t}, \lambda)$:
        $\mathbf{V}_G = (\mathrm{diag}(\mathbf{v}^{s-1}) + \mathbf{X}^\mathsf{T}\mathbf{X})^{-1}$
        Draw $\mathbf{z}$, so that $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \mathbf{V}_G\mathbf{X}^\mathsf{T}\mathbf{u}, \mathbf{V}_G)$.
        $\mathbf{w}^s = \mathbf{z}$
    C. Sample $\mathbf{v}^s$ from $p(\mathbf{v}|\mathbf{u}^s, \mathbf{w}^s, \mathbf{X}, \mathbf{t}, \lambda)$:
        For $j = 1, \ldots, D$:
            Draw $z > 0$, so that $p(z) \propto z^{-\frac{3}{2}} \mathrm{e}^{-\frac{(w_j^s)^2}{2} z - \frac{1}{2\lambda^2} z^{-1}}$.
            $v_j^s = \frac{1}{z}$
3. Select every tenth sample $\mathbf{w}^s$ from iterations $s > 500$ for the output set.
```

5.4 Hyperparameter Selection

5.4.1 Cross-validation

A proper value of the hyperparameter λ is selected separately for each approximate inference method, by using a similar cross-validation (Stone 1974) scheme as already described, when illustrating the overfitting of ARDEP (see 5.1.6 Alternative Criteria for Selection of the Final Relevance Hyperparameter Vector). A group of candidate values for λ is selected, and $\mathrm{CA}_{\mathrm{loso}}$ and $\mathrm{MLPP}_{\mathrm{loso}}$ are computed for each value according to equations 51 and 52, by leaving one subject at a time away from the training set and testing the model with the removed subject. Depending on which one of the measures of predictive performance has been chosen as the hyperparameter selection criterion, the candidate value with the best $\mathrm{CA}_{\mathrm{loso}}$ or with the best $\mathrm{MLPP}_{\mathrm{loso}}$ is selected as the final hyperparameter $\hat{\lambda}$. In this work, I use $\mathrm{MLPP}_{\mathrm{loso}}$ as the selection criterion, since it takes also uncertainties of the predictions into account. For clarity, I present the hyperparameter selection procedure for a given approximate inference method as the following pseudocode:

```
HYPERPARAMETER SELECTION

Input: data matrix $\mathbf{X}$ and label vector $\mathbf{t}$ including observations from $K$ subjects

Output: selected hyperparameter $\hat{\lambda}$ for the Laplace prior

1. For $2\lambda^2 = 10^{-6}, 10^{-5}, \ldots, 10^{0}$:
    A. For $k = 1, \ldots, K$:
        I. Divide $\mathbf{X}$ and $\mathbf{t}$ in two separate parts:
            $\mathbf{X}_{S_k}$, $\mathbf{t}_{S_k}$ including observations $i \in S_k$ belonging to subject $k$
            $\mathbf{X}^{\setminus S_k}$, $\mathbf{t}^{\setminus S_k}$ including the remaining observations $i \notin S_k$
        II. Train a model with $\mathbf{X}^{\setminus S_k}$, $\mathbf{t}^{\setminus S_k}$ and $\lambda$.
        III. Test the model by computing $\mathrm{MLPP}(\mathbf{t}_{S_k}, \mathbf{X}_{S_k} | \mathbf{t}^{\setminus S_k}, \mathbf{X}^{\setminus S_k}, \lambda)$.
```

B. Compute cross-validated predictive performance for λ:

$$\mathrm{MLPP}_{\mathrm{loso}}(\lambda) = \frac{1}{K}\sum_{k=1}^{K} \mathrm{MLPP}(\mathbf{t}_{S_k}, \mathbf{X}_{S_k}|\mathbf{t}^{\setminus S_k}, \mathbf{X}^{\setminus S_k}, \lambda)$$

2. Select λ with the best cross-validated predictive performance as $\hat{\lambda}$:

$$\hat{\lambda} = \arg\max_{\lambda}\{\mathrm{MLPP}_{\mathrm{loso}}(\lambda)\}$$

Figures 5 and 6 present $\mathrm{CA}_{\mathrm{loso}}$ and $\mathrm{MLPP}_{\mathrm{loso}}$ as a function of λ, for the two different classification settings: piano vs. speech and auditory vs. audiovisual. In the latter case, the selected hyperparameter value is $2\hat{\lambda}^2 = 10^{-4}$ for all the approximate inference methods, and the result would be the same, even if $\mathrm{CA}_{\mathrm{loso}}$ was used as a selection criterion. In the case of piano vs. speech setting, the selected value is $2\hat{\lambda}^2 = 10^{-5}$. If $\mathrm{CA}_{\mathrm{loso}}$ was used, the only exception would be MCMC with $2\hat{\lambda}^2 = 10^{-6}$. All in all, the cross-validated predictive performance measures seem to behave quite similarly for all methods and lead to consistent decisions. Both of the measures show the effect of overfitting with large hyperparameter values and on the other hand the effect of underfitting with small values of λ, although the latter effect appears more clear with $\mathrm{CA}_{\mathrm{loso}}$.

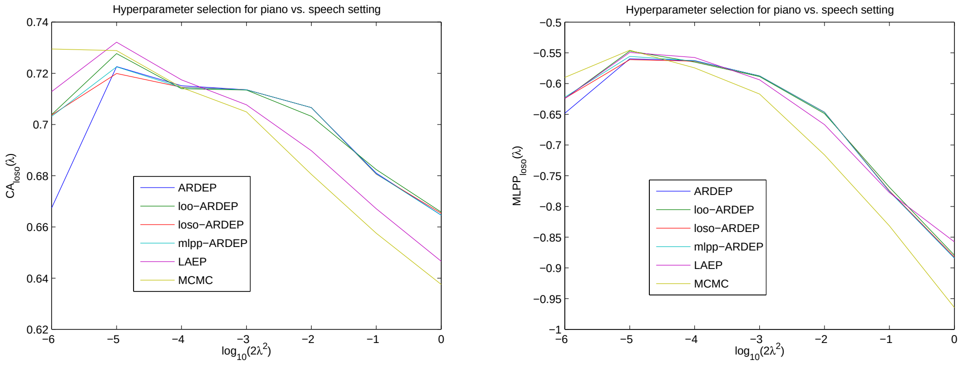

Figure 5: $\mathrm{CA}_{\mathrm{loso}}$ and $\mathrm{MLPP}_{\mathrm{loso}}$ obtained from a cross-validation scheme for piano vs. speech classification setting, presented as a function of λ.

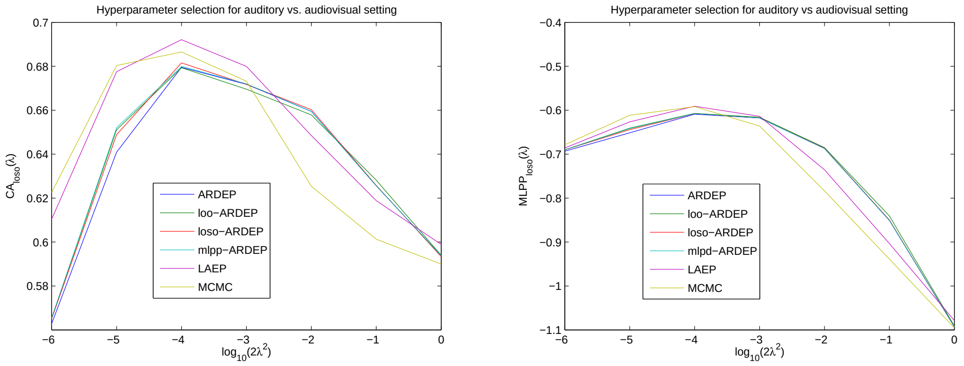

Figure 6: $\mathrm{CA}_{\mathrm{loso}}$ and $\mathrm{MLPP}_{\mathrm{loso}}$ obtained from a cross-validation scheme for auditory vs. audiovisual classification setting, presented as a function of λ.

5.4.2 Alternative Considerations

Cross-validation is a reliable way to find a proper hyperparameter value, but computationally it is quite expensive, since it requires the approximate inference to be run K times for each hyperparameter. Furthermore, using cross-validation for hyperparameter adjustment invalidates the measures of predictive performance produced by the cross-validation. Thus, the comparison of the predictive performance between different approximate inference methods must be carried out by a double-cross-validation, which means even $K(K-1)$ runs for each hyperparameter. For these reasons, it would be useful, if the hyperparameter could be reliably selected during the approximate inference algorithm without the expensive cross-validation.

For ARDEP, a possible solution to consider could be to use the EP estimates for $\mathrm{CA}_{\mathrm{loso}}$, $\mathrm{MLPP}_{\mathrm{loso}}$ or $\ln\left(p(\mathbf{t}|\mathbf{X},\mathbf{v})p(\mathbf{v}|\lambda)\right)$. However, when applying these estimates to the fMRI data, none of them manages to identify a proper hyperparameter value. Figure 7 presents an example of a typical behaviour of the estimates as a function of λ. As noticed, the approximation of $\ln\left(p(\mathbf{t}|\mathbf{X},\mathbf{v})p(\mathbf{v}|\lambda)\right)$ increases with decreasing λ, regardless of the underfitting. When the hyperprior is tightened enough, the marginal likelihood term $p(\mathbf{t}|\mathbf{X},\mathbf{v})$ should be reduced towards a chance level, but since the hyperprior density $p(\mathbf{v}|\lambda)$ keeps peaking higher and higher along the tightening, it dominates the measure. The EP estimates $\tilde{\mathrm{CA}}_{\mathrm{loso}}$ and $\tilde{\mathrm{MLPP}}_{\mathrm{loso}}$, instead, show opposite behaviour by increasing with increasing λ. When comparing the EP estimates to the true cross-validation measures, it is noticed that the looser the hyperprior is, i.e., the more freedom is given to the classifier to fit the data, the more overoptimistic the EP estimates become. This observation supports the conclusion made earlier, that removing approximate likelihood terms from the posterior approximation is not sufficient to remove the effect of the corresponding observations on the cavity approximations, and thus the EP estimates obtained this way cannot properly penalise overfitting. As noticed from the leftmost graph, the leave-one-subject-out estimate $\tilde{\mathrm{CA}}_{\mathrm{loso}}$ actually remains closer to the corresponding leave-one-out estimate $\tilde{\mathrm{CA}}_{\mathrm{loo}}$ (presented by the red curve) instead of following the true cross-validation measure $\mathrm{CA}_{\mathrm{loso}}$.

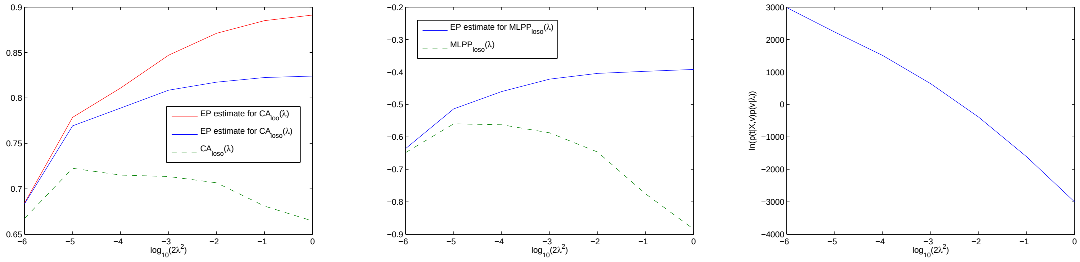

Figure 7: An example of EP estimates $\tilde{\mathrm{CA}}_{\mathrm{loo}}$, $\tilde{\mathrm{CA}}_{\mathrm{loso}}$ and $\tilde{\mathrm{MLPP}}_{\mathrm{loso}}$ and EP approximation for $\ln\left(p(\mathbf{t}|\mathbf{X},\mathbf{v})p(\mathbf{v}|\lambda)\right)$ produced by ARDEP as a function of λ. For reference, the true $\mathrm{CA}_{\mathrm{loso}}$ and $\mathrm{MLPP}_{\mathrm{loso}}$ obtained by cross-validation are presented as dashed lines.

For LAEP, a similar solution would be to use an approximation of the log marginal likelihood $\ln p(\mathbf{t}|\mathbf{X},\lambda)$ for λ, provided by the algorithm. Unfortunately this measure suffers from the same problem as $\tilde{\mathrm{CA}}_{\mathrm{loso}}$ and $\widetilde{\mathrm{MLPP}}_{\mathrm{loso}}$ in ARDEP. As noticed from figure 8, the approximate $\ln p(\mathbf{t}|\mathbf{X}, \lambda)$ keeps increasing with increasing λ, and thus favours overfitting.

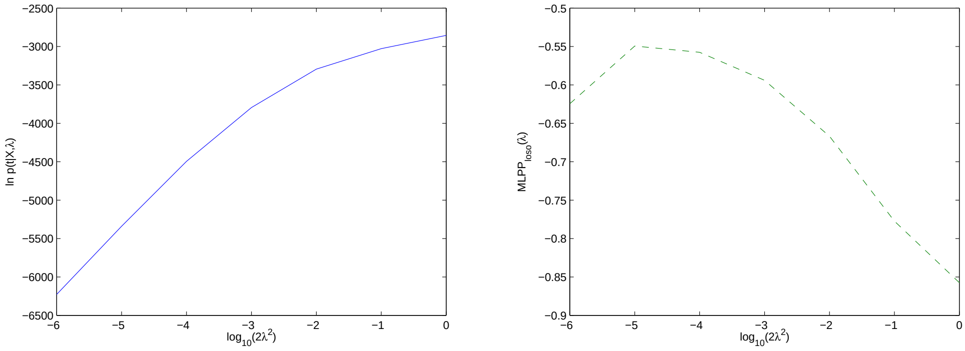

Figure 8: An example of an approximation of the log marginal likelihood $\ln p(\mathbf{t}|\mathbf{X}, \lambda)$ for λ, produced by LAEP, presented as a function of λ. For reference, $\mathrm{MLPP}_{\mathrm{loso}}$ obtained by cross-validation is presented as a dashed line in the contiguous graph.

In the case of MCMC, an alternative for selecting a fixed value for the hyperparameter λ would be to state a hyperprior $p(\lambda)$ and sample λ from the posterior distribution $p(\lambda|\mathbf{u}, \mathbf{w}, \mathbf{v}, \mathbf{X}, \mathbf{t})$ as a part of the Gibbs sampler. When applying this strategy to the fMRI data by using a similar idea of slice sampling as for $\mathbf{w}$, it is noticed that the samples of λ escape towards the upper limit of the sampling range. Since the increase of λ does not seem to be properly constrained by any sensibly formed prior distribution, also this alternative has to be discarded.

For the fMRI data used in this work, cross-validation seems to be the only reliable way to optimise the value of λ, in the case of any of the three approximate inference methods. If, however, there turns out to be no resources for such a heavy procedure, it would be worthwhile to have some hunch of a generally applicable value for the scale hyperparameter. A natural approach arises from a desire to avoid the possibility, that a constant shift in $\mathbf{x}_i$ would be able to change the classification result from almost certain $t = -1$ to almost certain $t = 1$, or vice versa (Gelman et al. 2008). From this perspective, it would be appropriate to scale $\mathbf{w}^{\mathsf{T}}\mathbf{x}_i$ according to the scale of the probit activation function. Since $2\lambda^2$ corresponds to the prior variance of the feature weights w_j, matching the prior variance of $\mathbf{w}^{\mathsf{T}}\mathbf{x}_i$ averaged over observations $i = 1, \ldots, N$ with the unit variance of the standard Gaussian distribution leads to

$$\frac{\sum_{i=1}^{N}\left(\sum_{j=1}^{D}([\mathbf{x}_i]_j)^2 2\lambda^2\right)}{N} = 2\lambda^2 \sum_{j=1}^{D} \frac{\sum_{i=1}^{N}([\mathbf{x}_i]_j)^2}{N} = 1. \tag{67}$$

Since the fMRI data used in this work has been standardised over each set of observations obtained from one voxel of one subject, $\frac{\sum_{i=1}^{N}([\mathbf{x}_i]_j)^2}{N}$ can be approximated as one also for the classification subsets. For the feature amount $D = 707$ this leads to

$$2\lambda^2 = \frac{1}{D} \approx 0.0014 \approx 10^{-2.8}. \tag{68}$$

As noticed from figures 5 and 6, this prior scale does not quite match with the optimal choices for the particular classification settings, but the predictive performance still remains near the maximum. Thus, it may be regarded as a heuristic for a sufficiently safe and uninformative choice of the hyperparameter, which, however, utilises information about the scale and the dimension of the data.

6 Results

The final classification model is the approximate posterior distribution of the feature weights, obtained by carrying out the approximate inference with the whole training data including all subjects. The distribution is visualised by presenting the marginal probabilities for the individual weights to be positive or negative, mapped into the corresponding brain locations. Since the whole data has been used in the inference stage, the predictive performance of this model model cannot be directly tested. Furthermore, because the hyperparameter λ has been adapted using cross-validation, a similar cross-validation is neither appropriate for testing the final model. Thus, the comparison between the predictive performances of the three different approximate inference methods is carried out by a double-cross-validation scheme, where one subject at a time is first left out of the data and then both the hyperparameter selection and the final approximate inference are performed without using the test subject.

6.1 Brain Maps for Piano vs. Speech Setting

To illustrate the final classification model, i.e., the approximate posterior distribution of the feature weight vector $\mathbf{w}$, the marginal distributions of individual weights w_j are mapped back to the brain by presenting the probability $\mathrm{P}(w_j > 0)$ for each feature j at the corresponding brain location. In the brain maps presented in this chapter, high probabilities are represented by bright yellow colour, indicating that high activations in these locations tend to move the classification result towards label $t = 1$. Low probabilities, in turn, are represented by dark red colour, indicating greater sensitivity to the condition labelled as $t = -1$. Neutral locations with $\mathrm{P}(w_j > 0) \approx 0.5$ are presented by orange.

Since the original data was thinned by removing every second voxel in each spatial dimension, filling the exact volumes corresponding to the features would not be too illustrative, as noticed from the left-hand map in figure 9. To produce better visualised maps, the gaps between the remaining voxels are filled by interpolating from the neighbouring voxels, as done in the right-hand map.

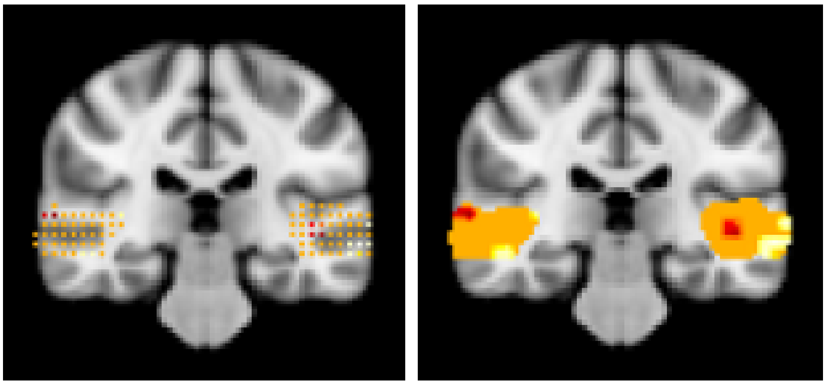

Figure 9: An example of the visual effect of interpolation.

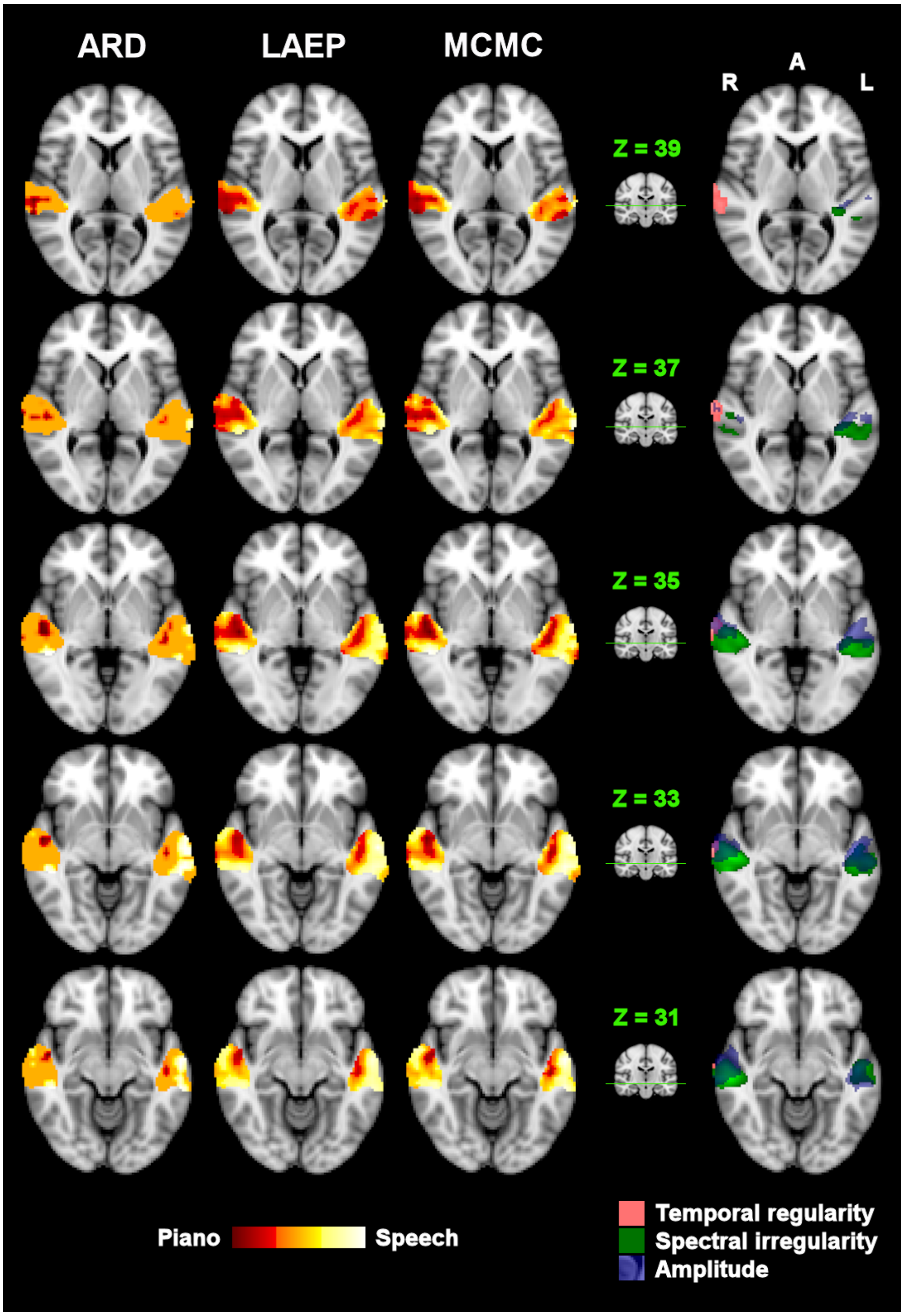

Figure 10: Interpolated brain maps obtained by different approximate inference methods for piano vs. speech setting, where negative and positive feature weights mean that activation in the corresponding voxel tend to move the classification result towards piano and speech, respectively. The colour scale from dark red (0) to bright yellow (1) represents the probability for the corresponding feature weight to be positive. Orange colour indicates neutral probability $\mathrm{P}(w_j > 0) \approx 0.5$. The rightmost slices present the locations, where activation is most related to temporal regularity, spectral irregularity and amplitude of the stimulus.

Figure 10 presents five horizontal slices of the interpolated brain maps for piano vs. speech setting, obtained by each of the three approximate inference methods. The most immediate observation concerns the sparseness of the ARDEP solution compared to the LAEP and MCMC solutions. As predicted earlier, ARDEP prunes most of the features effectively out of the model, resulting in a truly sparse solution with only about one hundred out of $D = 707$ weights being more probably on either side of the origin. As illustrated in the leftmost histogram in figure 11, a great deal of the remaining weights are almost certainly positive or almost certainly negative. The LAEP and MCMC solutions are spatially more continuous with smoother shifts between nearby voxels, and hence $\mathrm{P}(w_j > 0)$ is more often somewhere between the neutral 0.5 and the extremes 0 and 1.

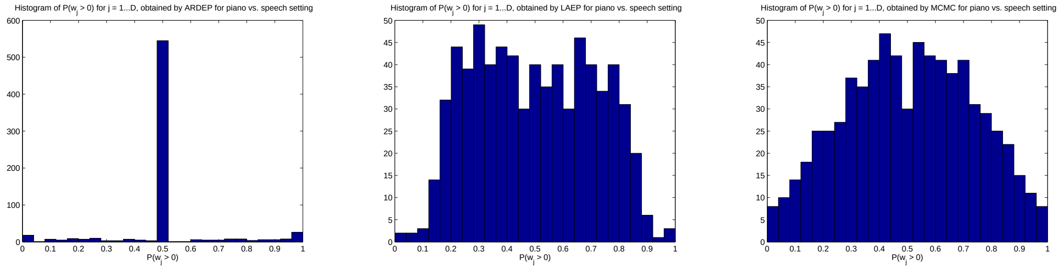

Figure 11: Histograms of $\mathrm{P}(w_j > 0)$ for $j = 1, \ldots, D$, obtained by different approximate inference methods for piano vs. speech setting.

The most discriminative locations between piano and speech in the region of interest are the same with each of the approximate inference methods, even though ARDEP prunes out some of the details. When looking generally, speech-related regions seem to be more inferior and lateral than piano-related regions. The most noticeable piano-related regions are located in the medial parts of superior temporal gyri (STG). The most probable piano-related voxels are in the right hemisphere, where the piano-related regions are also wider than in the left hemisphere, spreading also to the lateral parts of STG. The most probable speech-related voxels, in turn, are in the lateral parts of the left middle temporal gyrus (MTG) and superior temporal sulcus (STS). A corresponding speech-related region is noticed also in the left hemisphere, but it is smaller and restricted on the most posterior parts of the region of interest.

To support the interpretation of the results, the obtained patterns are compared to a separate analysis, where the fMRI signal was modelled as a sum of components related to the characteristics of the stimuli (Salmi et al. 2012). The locations, where activation is most related to temporal regularity (pink), spectral irregularity (green) and amplitude (transparent blue) of the stimulus, are presented in the rightmost slices of figure 10. The larger piano-related region in the right hemisphere seems to be connected with enhanced sensitivity to temporal regularity in the lateral parts of the right STG. The regions most sensitive to spectral irregularity, in turn, seem to match quite well with the most posterior speech-related regions in MTG and STS.

6.2 Brain Maps for Auditory vs. Audiovisual Setting

Figure 12 presents three horizontal slices of the interpolated brain maps for auditory (piano) vs. audiovisual (piano) setting, obtained by each of the three approximate inference methods. The general impression of the characteristic differences between the solutions is the same as in piano vs. speech setting. The sparseness of the ARDEP solution compared to the smoother LAEP and MCMC solutions becomes apparent also by looking at the histograms in figure 13. In this case, the ARDEP solution contains only a few dozen relevant voxels with most of them almost certainly positive or almost certainly negative.

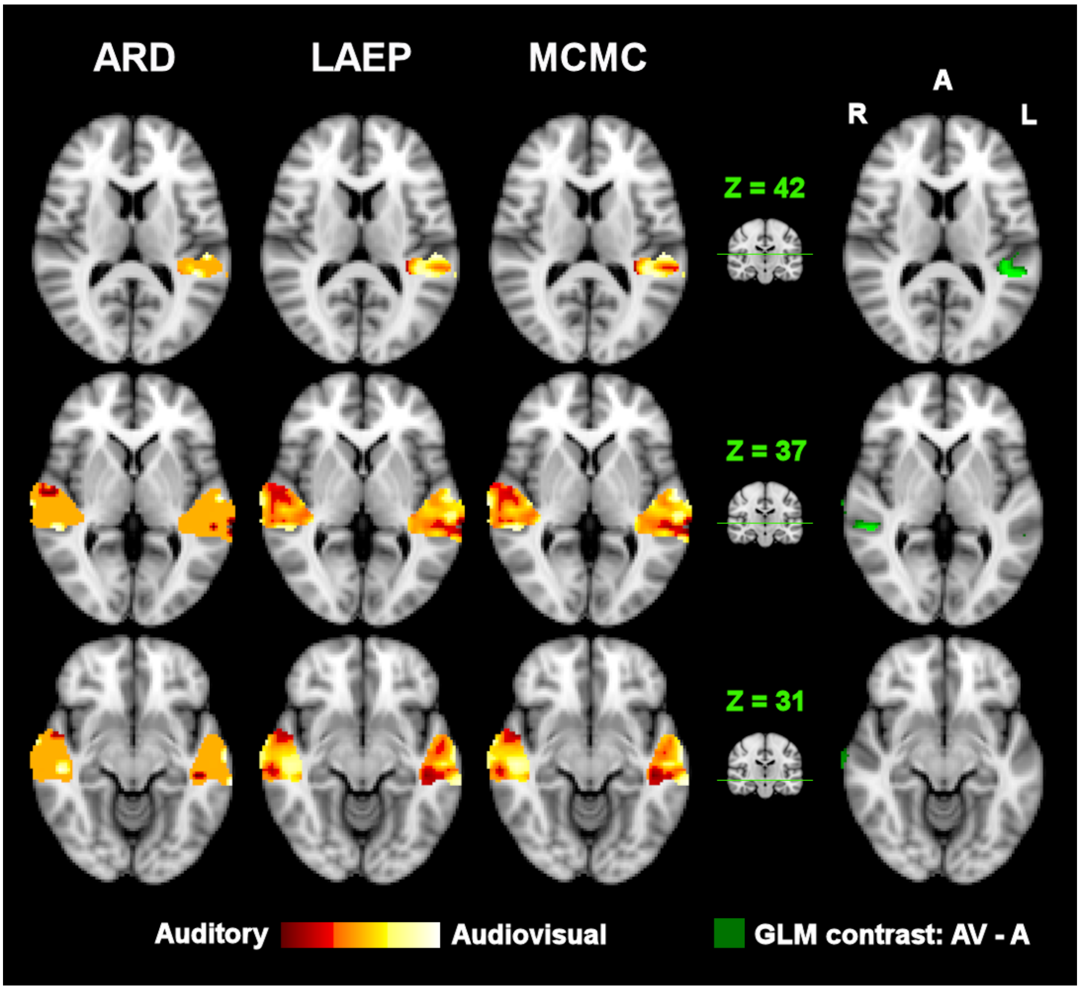

Figure 12: Interpolated brain maps obtained by different approximate inference methods for auditory (piano) vs. audiovisual (piano) setting, where negative and positive feature weights mean that activation in the corresponding voxel tend to move the classification result towards auditory and audiovisual, respectively. The colour scale from dark red (0) to bright yellow (1) represents the probability for the corresponding feature weight to be positive. Orange colour indicates neutral probability $P(w_j > 0) \approx 0.5$. The rightmost slices present the locations, where the contrast of activation during audiovisual (piano) compared to activation during auditory (piano) is most significant according to a separate GLM analysis.

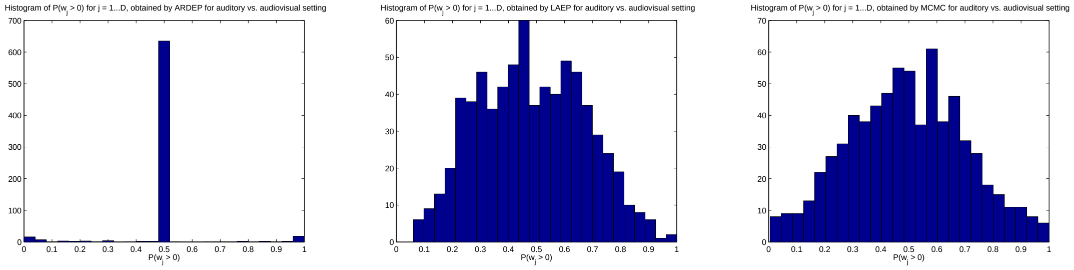

Figure 13: Histograms of $\mathrm{P}(w_j > 0)$ for $j = 1, \ldots, D$, obtained by different approximate inference methods for auditory vs. audiovisual setting.

In spite of even more effective pruning of features, ARDEP maintains all of the most discriminative regions obtained by LAEP and MCMC also for auditory (piano) vs. audiovisual (piano) setting. However, ARDEP loses the more specific shape of the regions by representing them as one or more smaller spots. The most noticeable regions related to audiovisual (AV) input are located in the medial parts of the right MTG and STS and in the left planum temporale (PT), which is the most superior part of the region of interest. Smaller AV-related regions are noticed also in the most posterior parts of the region of interest up in the right STG and down in the left lateral MTG.

The rightmost slices of figure 12 present the locations, where the contrast of activation during audiovisual piano-playing compared to activation during auditory piano-playing is most significant according to a separate multi-level general linear model (GLM) obtained by the analysis tool FEAT (Beckmann et al. 2003) in FMRIB Software Library. The GLM analysis distinguishes the same regions as mentioned in the superior parts of the region of interest, but does not show significant contrast in the more inferior regions.

6.3 Predictive Performance for Piano vs. Speech Setting

The predictive performance of the final model is evaluated by a double-cross-validation scheme, where the observations of one subject at a time are first left out of the data and then both the hyperparameter selection and the final approximate inference are carried out without using the removed observations. The removed subject is used as a test subject to compute CA and MLPP as defined in equations 31 and 33. The double-cross-validated measures $\mathrm{CA}_{\mathrm{double-loso}}$ and $\mathrm{MLPP}_{\mathrm{double-loso}}$ are obtained by averaging over different test subjects. For clarity, I present the double-cross-validation procedure for a given approximate inference method as the following pseudocode:

DOUBLE-CROSS-VALIDATION

Input: data matrix $\mathbf{X}$ and label vector $\mathbf{t}$ including observations from K subjects

Output: predictive performance measures $\mathrm{CA}_{\mathrm{double-loso}}$ and $\mathrm{MLPP}_{\mathrm{double-loso}}$

```
1. For $k = 1, \dots, K$:
    A. Divide $\mathbf{X}$ and $\mathbf{t}$ in two separate parts:
        $\mathbf{X}_{S_k}$, $\mathbf{t}_{S_k}$ including observations $i \in S_k$ belonging to subject $k$
        $\mathbf{X}^{\setminus S_k}$, $\mathbf{t}^{\setminus S_k}$ including the remaining observations $i \notin S_k$
    B. Run HYPERPARAMETER SELECTION using leave-$S_k$-out data:
        Input: $\mathbf{X}^{\setminus S_k}$ and $\mathbf{t}^{\setminus S_k}$ including observations from $K-1$ subjects
        Output: $\hat{\lambda}^{\setminus S_k}$
    C. Train a model with $\mathbf{X}^{\setminus S_k}$, $\mathbf{t}^{\setminus S_k}$ and $\hat{\lambda}^{\setminus S_k}$.
    D. Test the model by computing $\mathrm{CA}(\mathbf{t}_{S_k}, \mathbf{X}_{S_k} | \mathbf{t}^{\setminus S_k}, \mathbf{X}^{\setminus S_k}, \hat{\lambda}^{\setminus S_k})$ and
    $\mathrm{MLPP}(\mathbf{t}_{S_k}, \mathbf{X}_{S_k} | \mathbf{t}^{\setminus S_k}, \mathbf{X}^{\setminus S_k}, \hat{\lambda}^{\setminus S_k})$.

2. Compute double-cross-validated predictive performance:
    $\mathrm{CA}_{\mathrm{double-loso}} = \frac{1}{K} \sum_{k=1}^{K} \mathrm{CA}(\mathbf{t}_{S_k}, \mathbf{X}_{S_k} | \mathbf{t}^{\setminus S_k}, \mathbf{X}^{\setminus S_k}, \hat{\lambda}^{\setminus S_k})$
    $\mathrm{MLPP}_{\mathrm{double-loso}} = \frac{1}{K} \sum_{k=1}^{K} \mathrm{MLPP}(\mathbf{t}_{S_k}, \mathbf{X}_{S_k} | \mathbf{t}^{\setminus S_k}, \mathbf{X}^{\setminus S_k}, \hat{\lambda}^{\setminus S_k})$
```

Figure 14 presents $\mathrm{CA}_{\mathrm{double-loso}}$ and $\mathrm{MLPP}_{\mathrm{double-loso}}$ obtained by different approximate inference methods for piano vs. speech setting. The round spots represent $\mathrm{CA}_{\mathrm{double-loso}}$ and $\mathrm{MLPP}_{\mathrm{double-loso}}$ and the line segments are 95 % confidence intervals for classification accuracy and mean log predictive performance. The confidence intervals for classification accuracy are obtained analytically by assuming the amount of correct predictions to be binomially distributed and applying a uniform prior distribution for the probability of a single prediction to be correct (Bolstad 2007, pp. 141–143). The confidence intervals for mean log predictive probability are obtained by generating 1000 Bayesian bootstrap replicates of the set of the log predictive probabilities for the correct classes (Rubin 1981).

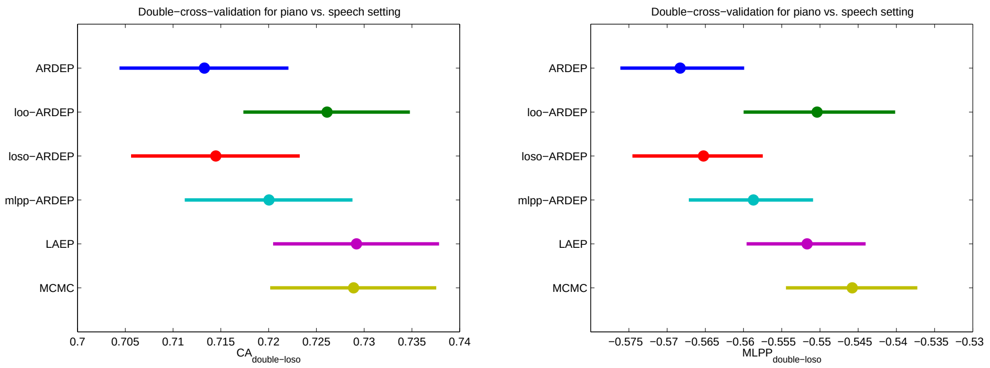

Figure 14: $\mathrm{CA}_{\mathrm{double-loso}}$ and $\mathrm{MLPP}_{\mathrm{double-loso}}$ with 95 % confidence intervals for piano vs. speech setting. The confidence interval for classification accuracy is obtained by assuming the amount of correct predictions to be binomially distributed. The confidence interval for mean log predictive probability is obtained by Bayesian bootstrap.

The chance levels for classification accuracy and mean log predictive probability are 0.5 and $\ln 0.5 \approx -0.69$, respectively. For all approximate inference methods, $\mathrm{CA}_{\mathrm{double-loso}}$ and $\mathrm{MLPP}_{\mathrm{double-loso}}$ are significantly higher: $\mathrm{CA}_{\mathrm{double-loso}}$ is above 0.70 and $\mathrm{MLPP}_{\mathrm{double-loso}}$ above -0.58. In the case of ARDEP, the confidence interval for classification accuracy lies approximately between 0.70 and 0.72, whereas the corresponding intervals for LAEP and MCMC lie between 0.72 and 0.74. A similar difference between LAEP and MCMC compared to ARDEP is observed in mean log predictive probabilities, with the performance of MCMC slightly above LAEP.

For reference, also the predictive performances of the alternative modifications of ARDEP are presented in the same figure. Even if loo-ARDEP and mlpp-ARDEP seem to perform better than the converged ARDEP, it is important to remember that the EP estimates they lean on do not properly follow the true predictive performance, as illustrated in the previous chapter. Due to the overfitting during the ARD framework, choosing any of the earlier iterations after a few initial ones may lead to a better performance on average, no matter which criterion is used for the selection. Note also, that the effect is not as significant with the predictive performance for loso-ARDEP, even if it uses a more properly defined EP estimate than loo-ARDEP. Thus, I argue that the better performance for loo-ARDEP and mlpp-ARDEP in this case is more of an illusion than a result of truly worthwhile modifications.

The individual predictive performance measures $\mathrm{CA}(\mathbf{t}_{S_k}, \mathbf{X}_{S_k} | \mathbf{t}^{\setminus S_k}, \mathbf{X}^{\setminus S_k}, \hat{\lambda}^{\setminus S_k})$ and $\mathrm{MLPP}(\mathbf{t}_{S_k}, \mathbf{X}_{S_k} | \mathbf{t}^{\setminus S_k}, \mathbf{X}^{\setminus S_k}, \hat{\lambda}^{\setminus S_k})$ for each test subject $k = 1, \ldots, 16$ are presented in figure 15. The individual classification accuracies vary from below 0.60 above 0.80, and the mean log predictive performances behave quite similarly. The most noticeable differences between the approximate inference methods occur for test subjects $k = 10$ and $k = 15$, with significantly lower predictive performance for ARDEP compared to LAEP and MCMC. When looking at the selected hyperparameters for each training set, it is noticed that the reduced predictive performance for these test subjects is due to overfitting caused by larger hyperparameters $2(\hat{\lambda}^{\setminus S_{10}})^2 = 2(\hat{\lambda}^{\setminus S_{15}})^2 = 10^{-4}$ selected for ARDEP. The larger hyperparameter is selected also for $\hat{\lambda}^{\setminus S_{13}}$ for both ARDEP and LAEP, which similarly reduces the classification accuracy compared to MCMC.

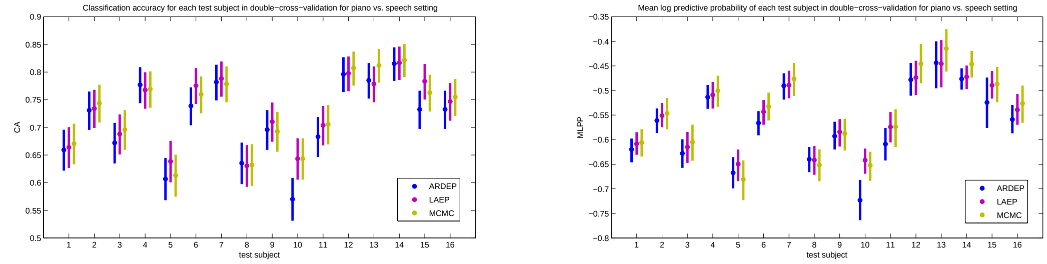

Figure 15: $\mathrm{CA}(\mathbf{t}_{S_k}, \mathbf{X}_{S_k} | \mathbf{t}^{\setminus S_k}, \mathbf{X}^{\setminus S_k}, \hat{\lambda}^{\setminus S_k})$ and $\mathrm{MLPP}(\mathbf{t}_{S_k}, \mathbf{X}_{S_k} | \mathbf{t}^{\setminus S_k}, \mathbf{X}^{\setminus S_k}, \hat{\lambda}^{\setminus S_k})$ for test subjects $k = 1, \ldots, 16$ with 95 % confidence intervals for piano vs. speech setting. The confidence intervals for classification accuracies are obtained by assuming the amount of correct predictions to be binomially distributed. The confidence intervals for mean log predictive probabilities are obtained by Bayesian bootstrap.

Figure 16 presents the predictive probabilities $p(t_i|x_i, \mathbf{t}^{\setminus S_k}, \mathbf{X}^{\setminus S_k}, \hat{\lambda}^{\setminus S_k})$ for the correct classes t_i of the individual test observations $i \in S_k$. The observations of all test subjects $k = 1, \ldots, 16$ are presented in the same graph, where the probabilities for correspondent observations obtained by different approximate inference methods are plotted pairwise against each other. As noticed, the individual probabilities are highly correlated between different methods. In the rightmost graphs, the core of the point cloud forms a gentle S letter, due to the more audacious predictions by MCMC. Most of the points outside the core are overoptimistic predictions for test subjects $k = 10$, $k = 13$ and $k = 15$, due to overfitting caused by the larger selected hyperparameters for the corresponding training data sets.

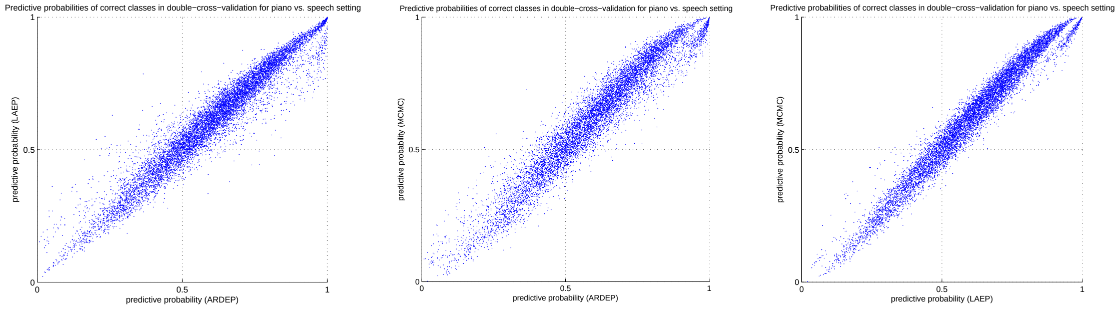

Figure 16: Predictive probabilities $p(t_i|x_i, \mathbf{t}^{\setminus S_k}, \mathbf{X}^{\setminus S_k}, \hat{\lambda}^{\setminus S_k})$ for the correct classes t_i, where $i \in S_k$, in piano vs. speech setting. The probabilities for correspondent observations obtained by different approximate inference methods are plotted pairwise against each other.

By determining 95 % confidence intervals for the mean difference of correspondent predictive probabilities between two approximate inference methods, it is confirmed that the differences between their predictive performances are significant. The confidence interval for the mean difference between LAEP and ARDEP is $[0.0037, 0.0057]$, between MCMC and ARDEP $[0.0169, 0.0191]$ and between MCMC and LAEP $[0.0125, 0.0142]$.

6.4 Predictive Performance for Auditory vs. Audiovisual Setting

Figure 17 presents $\text{CA}_{\text{double-loso}}$ and $\text{MLPP}_{\text{double-loso}}$ obtained by different approximate inference methods for auditory (piano) vs. audiovisual (piano) setting. Both of the predictive performance measures are lower than for piano vs. speech setting, but still significantly above the chance levels. In addition, the uncertainties on the classification accuracies are a little higher than in the first setting, which is natural with less available observations. In the case of ARDEP, the confidence interval for classification accuracy lies approximately between 0.66 and 0.69, whereas the corresponding interval for LAEP lies between 0.68 and 0.71. The classification accuracy for MCMC is slightly lower than for LAEP, even if their mean log predictive probabilities are at the same level. The alternative modifications of ARDEP do not show considerable differences compared to the converged ARDEP.

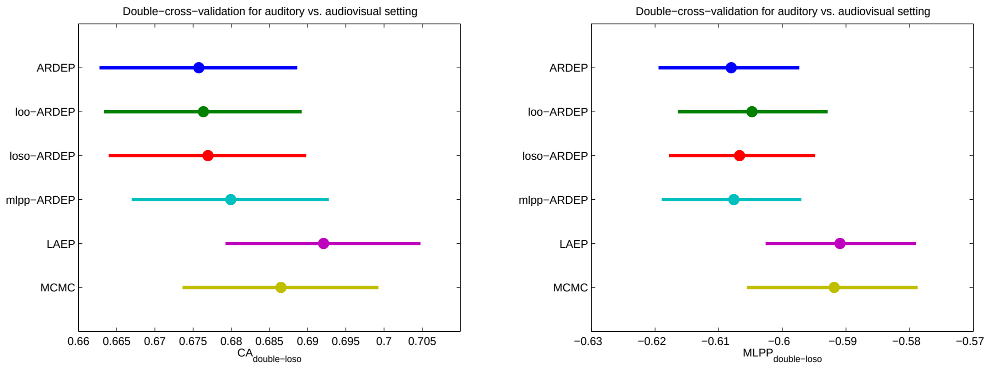

Figure 17: $\mathrm{CA}_{\mathrm{double-loso}}$ and $\mathrm{MLPP}_{\mathrm{double-loso}}$ with 95 % confidence intervals for auditory vs. audiovisual setting. The confidence interval for classification accuracy is obtained by assuming the amount of correct predictions to be binomially distributed. The confidence interval for mean log predictive probability is obtained by Bayesian bootstrap.

The individual predictive performance measures $\mathrm{CA}(\mathbf{t}_{S_k}, \mathbf{X}_{S_k} | \mathbf{t}^{\setminus S_k}, \mathbf{X}^{\setminus S_k}, \hat{\lambda}^{\setminus S_k})$ and $\mathrm{MLPP}(\mathbf{t}_{S_k}, \mathbf{X}_{S_k} | \mathbf{t}^{\setminus S_k}, \mathbf{X}^{\setminus S_k}, \hat{\lambda}^{\setminus S_k})$ for each test subject $k = 1, \ldots, 16$ are presented in figure 18. The variability between the individual measures is at the same level as for piano vs. speech setting. The uncertainties on most of them are, however, are noticeably higher than in the first case. In auditory vs. audiovisual setting, the selected hyperparameters are the same for each training set, with the exception of $2(\hat{\lambda}^{\setminus S_{14}}) = 10^{-3}$ for ARDEP. In this case, the larger hyperparameter does not stand out of the predictive performance measures.

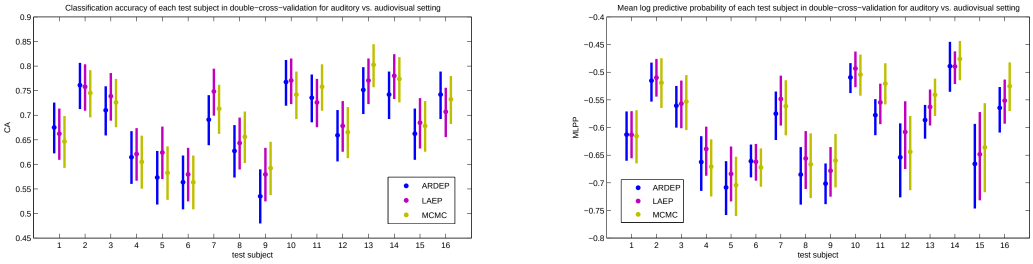

Figure 18: $\mathrm{CA}(\mathbf{t}_{S_k}, \mathbf{X}_{S_k} | \mathbf{t}^{\setminus S_k}, \mathbf{X}^{\setminus S_k}, \hat{\lambda}^{\setminus S_k})$ and $\mathrm{MLPP}(\mathbf{t}_{S_k}, \mathbf{X}_{S_k} | \mathbf{t}^{\setminus S_k}, \mathbf{X}^{\setminus S_k}, \hat{\lambda}^{\setminus S_k})$ for test subjects $k = 1, \ldots, 16$ with 95 % confidence intervals for auditory vs. audiovisual setting. The confidence intervals for classification accuracies are obtained by assuming the amount of correct predictions to be binomially distributed. The confidence intervals for mean log predictive probabilities are obtained by Bayesian bootstrap.

Figure 19 presents the predictive probabilities for the correct classes of the individual test observations, plotted pairwise against different approximate inference methods. The gentle S letter due to the more audacious predictions by MCMC stands out also for auditory vs. audiovisual setting. Otherwise the point clouds are more regularly shaped, since there are no such differences between the methods as caused by the deviant hyperparameters in piano vs. speech setting.

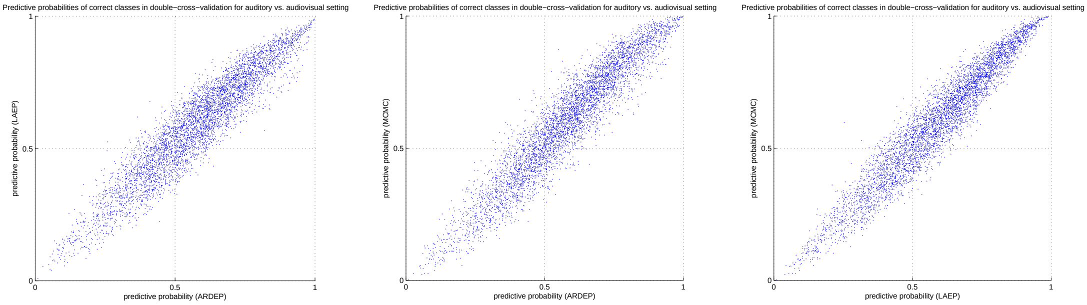

Figure 19: Predictive probabilities $p(t_i|x_i, \mathbf{t}^{\backslash S_k}, \mathbf{X}^{\backslash S_k}, \hat{\lambda}^{\backslash S_k})$ for the correct classes t_i, where $i \in S_k$, in auditory vs. audiovisual setting. The probabilities for correspondent observations obtained by different approximate inference methods are plotted pairwise against each other.

By determining 95 % confidence intervals for the mean difference of correspondent predictive probabilities between two approximate inference methods, a conflicting order is obtained between them, when compared to the predictive performance measures. The confidence interval for the mean difference between LAEP and ARDEP is $[0.0082, 0.0120]$, between MCMC and ARDEP $[0.0170, 0.0207]$ and between MCMC and LAEP $[0.0069, 0.0103]$. These suggest significant differences between the methods in the same order as in piano vs. speech setting, even if LAEP has slightly higher classification accuracy and mean log predictive performance than MCMC for auditory vs. audiovisual setting. Since the mean difference does not penalise the variability of the predictive probabilities, it favours the audacious predictions of MCMC. Even if the amount of false predictions is larger with MCMC than with LAEP, the higher predictive probabilities for the majority of the correct predictions overrides the effect of the false ones.

7 Discussion

In this work, Bayesian linear binary classification models with sparsity promoting Laplace priors were applied to analyse multi-voxel fMRI patterns related to natural audiovisual stimuli. The approach represents an opposite way of modelling compared to conventional generative methods by trying to predict the experimental condition from a given activation pattern. The parameters of the model, i.e., the classifier weights, represent the contribution of different brain locations to the result of the classification. Performing Bayesian inference on the parameters leads to a multivariate posterior distribution, which is assumed to reveal relevant information about the complex activation patterns related to different cognitive states. To carry out the approximate inference, three different methods were used: automatic relevance determination by expectation propagation (ARDEP), expectation propagation on the original Laplace prior (LAEP) and a Markov chain simulation method using the Gibbs sampler (MCMC). An appropriate scale hyperparameter for the Laplace prior controlling the sparsity of the model was adjusted by cross-validation separately for each method. The models obtained by different approximate inference methods were compared with respect to both the double-cross-validated predictive performance and the neuroscientifical interpretability of the obtained parameter distributions.

The analysed data included fMRI activation patterns measured from auditory cortex and some surrounding regions in the superior parts of temporal lobes during audiovisual and merely auditory perception of spoken and piano-played versions of popular songs. In the first classification setting, the observations were labelled into piano and speech classes in order to train a model that is able to predict whether a given activation pattern is more probably related to musical or spoken stimuli. The posterior distribution of the voxel weights was visualised by mapping the marginal probabilities for the individual weights to be positive or negative back into the corresponding brain location. According to these brain maps, speech-related regions seemed to be located generally in more inferior and lateral parts of the region of interest than piano-related regions. This finding is consistent with the results obtained for simpler stimuli by Tervaniemi et al. (2006). Furthermore, the results showed also differences between the hemispheres, suggesting left dominance of speech-related and right dominance of piano-related processing, which has been a common observation in neuroscientifical studies, best explained by regions specialised for temporal and spectral resolution (Zatorre and Schönwiesner 2011; DeWitt and Rauschecker 2012). This perspective was supported by a separate analysis (Salmi et al. 2012), where the fMRI signal was modelled as a sum of components related to the characteristics of the stimuli. According to these results, the larger piano-related region in the right hemisphere seemed to be connected with enhanced sensitivity to temporal regularity in the lateral parts of the right superior temporal gyrus (STG). The regions most sensitive to spectral irregularity, in turn, seemed to match quite well with the most posterior speech-related regions in middle temporal gyrus (MTG) and superior temporal sulcus (STS).

The second setting dealt only with the piano observations, dividing them into auditory and audiovisual classes in order to train a model that is able to discriminate between activation patterns related to perception of audiovisual and merely auditory piano-playing. The most noticeable regions related to audiovisual input were located in the medial parts of the right MTG and STS and in the left planum temporale (PT). STS is a multifunctional region commonly regarded to be involved, e.g., in biological motion processing and audiovisual integration (Hein and Knight 2008). According to the results of piano vs. speech setting with the right dominance of piano-related processing, it feels natural that the effect of audiovisual input is more distinguishable in the right STS. The left PT has also been suggested to be involved in multi-modal integration, and it has been found to be activated even during silent lipreading (Calvert et al. 1997). Similar activation has been reported also for silent piano-playing, suggesting that PT is related to learned sensory-motor associations (Hasegawa et al. 2004; Baumann et al. 2005). In addition to right STS and left PT, smaller AV-related regions were found also in the most posterior parts of the region of interest up in the right STG and down in the left lateral MTG. The brain maps were also compared to the results obtained by a separate multi-level general linear model (GLM). The conventional GLM analysis distinguished the same regions as mentioned in the superior parts of the region of interest, but did not show significant contrast in the more inferior regions.

In conclusion, the effects exposed by the brain maps were promising, suggesting that the proposed model is able to provide additional information about the brain activation patterns related to natural audiovisual stimuli. However, the visualisation of the marginal probabilities regarding the sign of the individual weights is a crude simplification of the multivariate posterior distribution. More sophisticated analyses of the correlation structure could reveal even more profound information about how different cognitive states are encoded in the brain. The predictive performance was significantly above chance level for both of the classification settings, importantly indicating the generalisability of the results. For speech vs. piano setting, the classification accuracy was a few percentage units above and for auditory vs. audiovisual setting a few units below 70 %, which is quite a good result for a joint model, considering that there may be significant differences in the functional and anatomical organisation of the brain between different subjects.

When comparing the models produced by the different approximate inference methods, there were relatively small differences between the predictive performances. For both classification settings, the classification accuracy for ARDEP was less than two percentage units worse than for LAEP and MCMC, which performed quite evenly. A similar behaviour was observed in the mean log predictive probabilities for the correct classes. Also the individual predictive probabilities were highly correlated between the different methods.

The LAEP and MCMC solutions turned out to be almost similar also with respect to the marginal probabilities of the weight parameter signs. Regardless of the Gaussian approximation used by LAEP, it is difficult to distinguish the brain maps produced by these two methods from each other. Thus, at this level of examination, the LAEP approximation seems to be accurate enough to replace the computationally expensive MCMC solution. Whereas LAEP and MCMC produced smooth solutions with natural spatial correlation between neighbouring voxels, ARDEP pruned most of the voxels out of the model, resulting instead in truly sparse solutions with less than one hundred relevant voxels out of the total amount of $D = 707$. This difference occurs, because ARDEP decomposes the Laplace prior into an exponentially distributed scale mixture of individual Gaussian priors on each parameter and regards these scales as relevance hyperparameters to be optimised by maximum a posteriori (MAP) estimation based on approximate marginal likelihood. Due to this optimisation, most of the voxel relevances reduce to zero, forcing also the corresponding weights to be equal to zero.

Sparse solutions are favoured in neuroscience and multi-voxel pattern analysis, because too large amount of adjustable parameters compared to the amount of available observations may reduce the predictive performance and the neuroscientifical interpretability of the resulting model (Rasmussen et al. 2012). Using a tightly scaled Laplace prior for the parameters may alleviate this problem by reducing the effect of irrelevant input features. Even though the Laplace prior promotes sparsity, a full Bayesian treatment always retains some uncertainty on the parameters, keeping all features included in the model. The idea of ARDEP is to automatically select only the relevant features to reduce dimensions and avoid the challenging integration over all uncertainty.

If the objective of the model is only to classify, a truly sparse model with slightly reduced predictive performance may still be desirable due to its frugal form. When it comes to neuroscientifical interpretability, the question is more ambiguous. For both of the classification settings, ARDEP was able to maintain all of the most discriminative regions obtained by LAEP and MCMC. However, ARDEP lost the more specific shape of the regions by representing them as one or more smaller spots. On one hand, the sparse maps may help to distinguish the most relevant regions from the complex pattern, but on the other hand, they may also hide relevant information by oversimplifying the interpretation.

The major problem in ARDEP concerns the treatment of correlated features. Even if one of two correlated voxels happens to explain the training data as well as the two voxels together, pruning another one of them out of the model may still reduce predictive performance for new data, and especially distort the interpretation of the resulting voxel patterns. This effect represents another type of overfitting, which may occur simultaneously with the conventional overfitting or underfitting. This typically means that during the relevance hyperparameter optimisation, the true predictive performance increases in the beginning, but at some point starts to decrease towards the converged level.

Even if the effect of overfitting was minimised by adjusting the hyperprior, it did not fully disappear. Thus, the algorithm produced better predictive performance, if an earlier iteration was selected instead of the converged MAP estimate for the relevance hyperparameters. To detect the optimal iteration, EP estimates for predictive performance were tested as alternative criterions. However, these estimates were not able to properly simulate testing with new data, and thus they did not follow the true predictive performance. One approach for the problem of correlated voxels would be to define additional spatial dependencies or other prior information based on neuroscientifical knowledge. Another solution would be to use frameworks that automatically couple correlated variables to be included in the model or exluded from it together (Zou and Hastie 2005; Qi and Yan 2011).

The example settings used in this work comprised only voxels from a restricted region of interest. However, it has been suggested that one region, e.g., superior temporal sulcus, may support several different functions depending on the activation of a wider network (Hein and Knight 2008). Thus, extending the region of interest to comprise the whole brain may be rewarding for future projects. Increasing the amount of voxels while keeping a constant amount of observations inevitably brings further challenges regarding the generalisability and the interpretability of the resulting models. To obtain the best possible interpretability, the mere predictive performance may not generally be the optimal criterion for the selection of the scale hyperparameter. As demonstrated by Rasmussen et al. (2012), a better solution could possibly be to balance a trade-off between the predictive power and the spatial reproducibility of the model.

References

Albert, J. H. and Chib, S. Bayesian analysis of binary and polychotomous response data. *Journal of the American Statistical Association,* 1993, vol. 88, no. 422, pp. 669–679.

Andrews, D. F. and Mallows, C. L. Scale mixtures of normal distributions. *Journal of the Royal Statistical Society, Series B: Methodological,* 1974, vol. 36, no. 1, pp. 99–102.

Baumann, S., Koeneke, S., Meyer, M., Lutz, K. and Jäncke, L. A network for sensory-motor integration: what happens in the auditory cortex during piano playing without acoustic feedback? *Annals of the New York Academy of Sciences,* 2005, vol. 1060, no. 1, pp. 186–188.

Beckmann, C. F., Jenkinson, M. and Smith, S. M. General multilevel linear modeling for group analysis in fMRI. *NeuroImage,* 2003, vol. 20, no. 2, pp. 1052–1063.

Bishop, C. M. *Pattern Recognition and Machine Learning.* New York, Springer Science+Business Media LLC, 2006.

Bolstad, W. M. *Introduction to Bayesian Statistics,* 2$^{\text{nd}}$ edition. Hoboken, New Jersey, John Wiley & Sons Inc., 2007.

Calvert, G. A., Bullmore, E. T., Brammer, M. J., Campbell, R., Williams, S. C. R., McGuire, P. K., Woodruff, P. W. R., Iversen, S. D. and David, A. S. Activation of auditory cortex during silent lipreading. *Science,* 1997, vol. 276, no. 5312, pp. 593–596.

DeWitt, I. and Rauschecker, J. P. Phoneme and word recognition in the auditory ventral stream. *Proceedings of the National Academy of Sciences of the United States of America,* 2012, vol. 109, no. 8, pp. E505–E514.

Driver, J. and Noesselt, T. Multisensory interplay reveals crossmodal influences on 'sensory-specific' brain regions, neural responses, and judgements. *Neuron,* 2008, vol. 57, no. 1, pp. 11–23.

Faul, A. C. and Tipping, M. E. Analysis of sparse Bayesian learning. Published in: Dietterich, T. G., Becker, Z. and Ghahramani, Z. (eds.) *Advances in Neural Information Processing Systems 14.* Vancouver, 2001. Cambridge, Massachusetts, The MIT Press, 2002, pp. 383–389.

Friston, K. J., Holmes, A. P., Worsley, K. J., Poline, J.-P., Frith, C. D. and Frackowiak, R. S. J. Statistical parametric maps in functional imaging: a general linear approach. *Human Brain Mapping,* 1994, vol. 2, no. 4, pp. 189–210.

Gelman, A., Carlin, J. B., Stern, H. S. and Rubin, D. B. *Bayesian Data Analysis.* 2$^{\text{nd}}$ edition. Boca Raton, Florida, Chapman & Hall/CRC, 2004.

Gelman, A., Jakulin, A., Pittau, M. G. and Su, Y.-S. A weakly informative default prior distribution for logistic and other regression models. *The Annals of Applied Statistics,* 2008, vol. 2, no. 4, pp. 1360–1383.

Geman, S. and Geman, D. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 1984, vol. 6, no. 1, pp. 721–741.

Hasegawa, T., Matsuki, K.-I., Ueno, T., Maeda, Y., Matsue, Y., Konishi, Y. and Sadato, N. Learned audio-visual cross-modal associations in observed piano playing activate the left planum temporale: an fMRI study. *Cognitive Brain Research,* 2004, vol. 20, no. 3, pp. 510–518.

Hein, G. and Knight, R. T. Superior temporal sulcus – it's my area: or is it? *Journal of Cognitive Neuroscience,* 2008, vol. 20, no. 12, pp. 2125–2136.

Huettel, S. A., Song, A. W. and McCarthy, G. *Functional Magnetic Resonance Imaging.* Sunderland, Massachusetts, Sinauer Associates Inc., 2004.

Kaas, J. H. and Hackett, T. A. Subdivisions of auditory cortex and processing streams in primates. *Proceedings of the National Academy of Sciences of the United States of America,* 2000, vol. 97, no. 22, pp. 11793–11799.

Klöppel, S., Stonnington, C. M., Chu, C., Draganski, B., Scahill, R. I., Rohrer, J. D., Fox, N. C., Clifford, R. J., Ashburner, J. and Frackowiak, R. S. J. Automatic classification of MR scans in Alzheimer's disease. *Brain,* 2008, vol. 131, no. 3, pp. 681–689.

Koelewijn, T., Bronkhorst, A. and Theeuwes, J. Attention and the multiple stages of multisensory integration: a review of audiovisual studies. *Acta Psychologica,* 2010, vol. 134, no. 3, pp. 372–384.

Levitt, M. H. *Spin Dynamics.* 2$^{\text{nd}}$ edition. Chichester, United Kingdom, John Wiley & Sons Ltd, 2008.

MacKay, D. J. C. Bayesian non-linear modeling for the prediction competition. Published in: *ASHRAE Transactions, Vol. 100, Pt. 2.* Orlando, 1994. Atlanta, ASHRAE, 1994, pp. 1053–1062.

Mazziotta, J. C., Toga, A. W., Evans, A. C., Fox, P. T., Lancaster, J. L., Zilles, K., Woods, R. P., Paus, T., Simpson, G., Pike, G. B., Holmes, C. J., Collins, D. L., Thompson, P. M., MacDonald, D., Iacoboni, M., Schormann, T., Amunts, K., Palomero-Gallagher, N., Geyer, S., Parsons, L. M., Narr, K. L., Kabani, N. J., Le Goualher, G., Boomsma, D. I., Cannon, T. D., Kawashima, R. and Mazoyer, B. A probabilistic atlas and reference system for the human brain: International Consortium for Brain Mapping (ICBM). *Philosophical Transactions of the Royal Society B: Biological Sciences,* 2001, vol. 356, no. 1412, pp. 1293–1322.

McRobbie, D. W., Moore, E. A., Graves, M. J. and Prince, M. R. *MRI from Picture to Proton.* 2nd edition. Cambridge, United Kingdom, Cambridge University Press, 2007.

Minka, T. P. *A Family of Algorithms for Approximate Bayesian Inference.* Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, 2001.

Neal, R. M. *Bayesian Learning for Neural Networks.* Ph.D. thesis, University of Toronto, Toronto, 1994.

Neal, R. M. Slice sampling. *The Annals of Statistics,* 2003, vol. 31, no. 3, pp. 705–741.

Nicholls, J. G., Martin, A. R., Wallace, B. G. and Fuchs, P. A. *From Neuron to Brain.* 4th edition. Sunderland, Massachusetts, Sinauer Associates Inc., 2001.

Nickisch, H. and Rasmussen, C. E. Approximations for binary Gaussian process classification. *Journal of Machine Learning Research,* 2008, vol. 9, no. 10, pp. 2035–2078.

Oostenveld, R., Fries, P., Maris, E. and Schoffelen, J.-M. FieldTrip: open source software for advanced analysis of MEG, EEG, and invasive electrophysiological data. *Computational Intelligence and Neuroscience,* 2011, vol. 2011, article ID 156869.

O'Toole, A. J., Jiang, F., Abdi, H., Pénard, N., Dunlop, J. P. and Parent, M. A. Theoretical, statistical, and practical perspectives on pattern-based classification approaches to the analysis of functional neuroimaging data. *Journal of Cognitive Neuroscience,* 2007, vol. 19, no. 11, pp. 1735–1752.

Pereira, F., Mitchell, T. and Botvinick, M. Machine learning classifiers and fMRI: a tutorial overview. *NeuroImage,* 2009, vol. 45, supplement 1, pp. S199–S209.

Press, W. H., Teukolsky, S. A., Vetterling, W. T. and Flannery, B. P. *Numerical Recipes in C++.* 2nd edition. New York, Cambridge University Press, 2002.

Qi, Y., Minka, T. P., Picard R. W. and Ghahramani, Z. Predictive automatic relevance determination by expectation propagation. Published in: Brodley, C. E. (ed.) *Proceedings of the* 21st *International Conference on Machine Learning.* Banff, Canada, 2004. New York, Accociation for Computing Machinery Inc., 2004, p. 85.

Qi, Y. and Yan, F. EigenNet: A Bayesian hybrid of generative and conditional models for sparse learning. Published in: Shawe-Taylor, J., Zemel, R. S., Bartlett, P. L., Pereira, F. C. N. and Weinberger, K. Q. (eds.) *Advances in Neural Information Processing Systems 24.* Granada, Spain, 2011. Red Hook, New York, Curran Associates Inc., 2012, pp. 2663–2671.

Rasmussen, C. E. and Williams, C. K. I. *Gaussian Processes for Machine Learning.* Cambridge, Massachusetts, The MIT Press, 2006.

Rasmussen, P. M., Madsen, K. H., Churchill, N. W., Hansen, L. K. and Strother, S. C. Model sparsity and brain pattern interpretation of classification models in neuroimaging. In press. *Pattern Recognition,* 2012, vol. 45, no. 6, pp. 2085-2100.

Read, H. L., Winer, J. A. and Schreiner, C. E. Functional architecture of auditory cortex. *Current Opinion in Neurobiology,* 2002, vol. 12, no. 4, pp. 433–440.

Robert, C. P. Simulation of truncated normal variables. *Statistics and Computing,* 1995, vol. 5, no. 2, pp. 121–125.

Ross, S. M. *Introduction to Probability Models.* 7th edition. San Diego, A Harcourt Science and Technology Company, 2000.

Rubin, D. B. The Bayesian Bootstrap. *The Annals of Statistics,* 1981, vol. 9, no. 1, pp. 130–134.

Salmi, J., Glerean, E., Mäkelä, S., Vehtari, A., Jylänki, P., Kettunen, J., Jääskeläinen, I. P., Nummenmaa, L., Nummi-Kuisma, K., Nummi, I. and Sams, M. Parcellating the neuronal patterns for natural audiovisual speech and music with fMRI. Abstract. Accepted for the 18th Annual Meeting of the Organization on Human Brain Mapping. Beijing, 2012. Poster no. 980MT.

Schmidt, M. *Blogreg.* Web document. 2006. Available at: `http://www.di.ens.fr/~mschmidt/Software/blogreg.html`. Accessed 21st March 2012.

Shiryaev, A. N. *Probability,* 2nd edition. New York, Springer-Verlag New York Inc., 1996.

Smith, S. M., Jenkinson, M., Woolrich, M. W., Beckmann, C. F., Behrens, T. E., Johansen-Berg, H., Bannister, P. R., De Luca, M., Drobnjak, I., Flitney, D. E., Niazy, R. K., Saunders, J., Vickers, J., Zhang, Y., De Stefano, N., Brady, J. M. and Matthews, P. M. Advances in functional and structural MR image analysis and implementation as FSL. *NeuroImage,* 2004, vol. 23, supplement 1, pp. S208–S219.

Stone, M. Cross-validatory choice and assesment of statistical predictions. *Journal of the Royal Statistical Society, Series B: Methodological,* 1974, vol. 36, no. 2, pp. 111–147.

Talavage, T. M., Sereno, M. I., Melcher, J. R., Ledden, P. J., Rosen, B. R. and Dale, A. M. Tonotopic organization in human auditory cortex revealed by progressions of frequency sensitivity. *Journal of Neurophysiology,* 2004, vol. 91, no. 3, pp. 1282–1296.

Tervaniemi, M., Szameitat, A. J., Kruck, S., Schröger, E., Alter, K., De Baene, W. and Friederici, A. D. From air oscillations to music and speech: functional magnetic resonance imaging evidence for fine-tuned neural networks in audition. *The Journal of Neuroscience,* 2006, vol. 26, no. 34, pp. 8647–8652.

Tibshirani, R. Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society, Series B: Methodological,* 1996, vol. 58, no. 1, pp. 267–288.

Thulborn, K. R., Waterton, J. C., Matthews, P. M. and Radda, G. K. Oxygenation dependence of the transverse relaxation time of water protons in whole blood at high field. *Biochimica et Biophysica Acta,* 1982, vol. 714, no. 2, pp. 265-270.

van Gerven, M. A. J., Cseke, B., de Lange, F. P. and Heskes, T. Efficient Bayesian multivariate fMRI analysis using a sparsifying spatio-temporal prior. *NeuroImage,* 2010, vol. 50, no. 1, pp. 150–161.

Vanhatalo, J., Riihimäki, J., Hartikainen, J. and Vehtari, A. Bayesian modeling with Gaussian processes using the MATLAB toolbox GPstuff. Submitted for publication.

Wolpaw, J. R., Birbaumer, N., McFarland, D. J., Pfurtscheller, G. and Vaughan, T. M. Brain-computer interfaces for communication and control. *Clinical Neurophysiology,* 2002, vol. 113, no. 6, pp. 767–791.

Zatorre, R. J. and Schönwiesner, M. Cortical speech and music processes revealed by functional meuroimaging. Published in: Winer, J. A. and Schreiner, C. E. (eds.) *The Auditory Cortex.* New York, Springer Science+Business Media LLC, 2011, pp. 657–677.

Zou, H. and Hastie, T. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society, Series B: Statistical Methodology,* 2005, vol. 67, no. 2, pp. 301–320.

A Expectation Propagation for the Probit Model with ARD Prior

Expectation propagation (EP) is a family of algorithms for approximate Bayesian inference, developed by Thomas Minka (2001). The original introduction demonstrates that the technique can be applied also for the probit model with a spherical Gaussian prior. To be able to utilise EP as a part of an automatic relevance determination (ARD) framework, Qi et al. (2004) present a corresponding EP algorithm with an ARD prior, but without detailed derivation. In this appendix, I derive Qi's presentation of the algorithm in detail and reform it in a computationally efficient way, as it is used in the ARDEP approximate inference method (see 5.1 Automatic Relevance Determination by Expectation Propagation).

Approximations

To begin with, recall equation 27 (p. 17) for the probit model likelihood $p(\mathbf{t}|\mathbf{w},\mathbf{X}) = \prod_{i=1}^{N}\Psi(t_i\mathbf{w}^\mathsf{T}\mathbf{x}_i)$ and equation 39 (p. 21) for the ARD prior $p(\mathbf{w}|\mathbf{v}) = \prod_{j=1}^{D}\mathcal{N}(0,v_j)$. This EP algorithm approximates the likelihood terms $g_i = \Psi(t_i\mathbf{w}^\mathsf{T}\mathbf{x}_i)$ by unnormalised Gaussians

$$\tilde{g}_i(\mathbf{w}) = \varsigma_i \mathrm{e}^{-\frac{1}{2\sigma_i}(t_i\mathbf{w}^\mathsf{T}\mathbf{x}_i-\rho_i)^2}. \tag{A1}$$

Consequently, the approximation $\tilde{q}(\mathbf{w})$ of the posterior distribution $p(\mathbf{w}|\mathbf{t},\mathbf{X},\mathbf{v})$ becomes also Gaussian:

$$\begin{aligned}
p(\mathbf{w}|\mathbf{t},\mathbf{X},\mathbf{v}) &= \frac{1}{Z_P}\left(\prod_{i=1}^{N} g_i(\mathbf{w})\right)p(\mathbf{w}|\mathbf{v}) \approx \tilde{q}(\mathbf{w}) = \frac{1}{\tilde{Z}_P}\left(\prod_{i=1}^{N}\tilde{g}_i(\mathbf{w})\right)\mathcal{N}(\mathbf{w}|0,\mathbf{V})\\
&= \frac{1}{\tilde{Z}_P}\left(\prod_{i=1}^{N}\varsigma_i\right)\mathrm{e}^{-\frac{1}{2}(\boldsymbol{\Phi}\mathbf{w}-\boldsymbol{\rho})^\mathsf{T}\boldsymbol{\Lambda}^{-1}(\boldsymbol{\Phi}\mathbf{w}-\boldsymbol{\rho})}(2\pi)^{-\frac{D}{2}}|\mathbf{V}|^{-\frac{1}{2}}\mathrm{e}^{-\frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{V}^{-1}\mathbf{w}}\\
&= \frac{1}{\tilde{Z}_P}\left(\prod_{i=1}^{N}\varsigma_i\right)(2\pi)^{-\frac{D}{2}}|\mathbf{V}|^{-\frac{1}{2}}\mathrm{e}^{-\frac{1}{2}\left(\mathbf{w}^\mathsf{T}(\boldsymbol{\Phi}^\mathsf{T}\boldsymbol{\Lambda}^{-1}\boldsymbol{\Phi}+\mathbf{V}^{-1})\mathbf{w}-2\boldsymbol{\rho}^\mathsf{T}\boldsymbol{\Lambda}^{-1}\boldsymbol{\Phi}\mathbf{w}+\boldsymbol{\rho}^\mathsf{T}\boldsymbol{\Lambda}^{-1}\boldsymbol{\rho}\right)}\\
&= \frac{1}{\tilde{Z}_P}\left(\prod_{i=1}^{N}\varsigma_i\right)(2\pi)^{-\frac{D}{2}}|\mathbf{V}|^{-\frac{1}{2}}\mathrm{e}^{-\frac{1}{2}\left(\mathbf{w}^\mathsf{T}\mathbf{V}_w^{-1}\mathbf{w}-2\mathbf{m}_w^\mathsf{T}\mathbf{V}_w^{-1}\mathbf{w}+\boldsymbol{\rho}^\mathsf{T}\boldsymbol{\Lambda}^{-1}\boldsymbol{\rho}\right)}\\
&= \frac{1}{\tilde{Z}_P}\left(\prod_{i=1}^{N}\varsigma_i\right)|\mathbf{V}|^{-\frac{1}{2}}\mathrm{e}^{-\frac{1}{2}(\boldsymbol{\rho}^\mathsf{T}\boldsymbol{\Lambda}^{-1}\boldsymbol{\rho}-\mathbf{m}_w^\mathsf{T}\mathbf{V}_w^{-1}\mathbf{m}_w)}(2\pi)^{-\frac{D}{2}}\mathrm{e}^{-\frac{1}{2}\left((\mathbf{w}-\mathbf{m}_w)^\mathsf{T}\mathbf{V}_w^{-1}(\mathbf{w}-\mathbf{m}_w)\right)}\\
&= \frac{1}{\tilde{Z}_P}\left(\prod_{i=1}^{N}\varsigma_i\right)|\mathbf{V}|^{-\frac{1}{2}}\mathrm{e}^{-\frac{1}{2}(\boldsymbol{\rho}^\mathsf{T}\boldsymbol{\Lambda}^{-1}\boldsymbol{\rho}-\mathbf{m}_w^\mathsf{T}\mathbf{V}_w^{-1}\mathbf{m}_w)}|\mathbf{V}_w|^{\frac{1}{2}}\mathcal{N}(\mathbf{w};\mathbf{m}_w,\mathbf{V}_w)\\
&= \mathcal{N}(\mathbf{w};\mathbf{m}_w,\mathbf{V}_w),
\end{aligned} \tag{A2}$$

where

$$
\begin{aligned}
\mathbf{V} &= \mathrm{diag}(\mathbf{v}), && \text{(A3)}\\
\boldsymbol{\Phi} &= (t_1\mathbf{x}_1, \ldots, t_N\mathbf{x}_N)^{\mathsf{T}}, && \text{(A4)}\\
\boldsymbol{\rho} &= (\rho_1, \ldots, \rho_N)^{\mathsf{T}}, && \text{(A5)}\\
\boldsymbol{\sigma} &= (\sigma_1, \ldots, \sigma_N)^{\mathsf{T}}, && \text{(A6)}\\
\boldsymbol{\Lambda} &= \mathrm{diag}(\boldsymbol{\sigma}), && \text{(A7)}\\
\mathbf{V}_w &= (\boldsymbol{\Phi}^{\mathsf{T}}\boldsymbol{\Lambda}^{-1}\boldsymbol{\Phi} + \mathbf{V}^{-1})^{-1}, && \text{(A8)}\\
\mathbf{m}_w &= \mathbf{V}_w\boldsymbol{\Phi}^{\mathsf{T}}\boldsymbol{\Lambda}^{-1}\boldsymbol{\rho}, && \text{(A9)}\\
\tilde{Z}_P &= \left(\prod_{i=1}^{N} \varsigma_i\right) |\mathbf{V}|^{-\frac{1}{2}}, \mathrm{e}^{-\frac{1}{2}(\boldsymbol{\rho}^{\mathsf{T}}\boldsymbol{\Lambda}^{-1}\boldsymbol{\rho} - \mathbf{m}_w^{\mathsf{T}}\mathbf{V}_w^{-1}\mathbf{m}_w)}|\mathbf{V}_w|^{\frac{1}{2}}. && \text{(A10)}
\end{aligned}
$$

Parameter Iteration

As already described in subsection 5.1.3, EP finds the approximations $\tilde{g}_i$ for the likelihood terms g_i by iteratively updating the site parameters $(\rho_i, \sigma_i, \varsigma_i)$ term by term. After initialising the approximate posterior $\tilde{q}$ to converge with the ARD prior, i.e., setting $\mathbf{m}_w = 0$, $\mathbf{V}_w = \mathrm{diag}(\mathbf{v})$ and $\tilde{g}_i = 1$ for all $i = 1, \ldots, N$, an approximate term $\tilde{g}_i$ is removed from the approximate posterior and replaced first by the accurate term g_i to obtain a target posterior approximation $\hat{q}$. The new term approximation $\tilde{g}_i{}^*$ is then chosen to minimise the Kullback-Leibler divergence between $\hat{q}$ and the new posterior approximation $\tilde{q}^*$. The same is done for the other terms, respectively, using the updated posterior approximation $\tilde{q}^*$ as the initial $\tilde{q}$, and the procedure is repeated until the site parameters $(\rho_i, \sigma_i, \varsigma_i)$ for all $i = 1, \ldots, N$ converge.

Cavity Parameters

I begin the derivation of the parameter update rules by denoting the corresponding variables for $\boldsymbol{\Phi}$, $\boldsymbol{\rho}$, $\boldsymbol{\sigma}$ and $\boldsymbol{\Lambda}$ including only the observations $i \in S$ by $\boldsymbol{\Phi}_S$, $\boldsymbol{\rho}_S$, $\boldsymbol{\sigma}_S$ and $\boldsymbol{\Lambda}_S$, respectively. By extracting observations $i \in S$ from equations A8 and A9 and using the Woodbury formula (Press et al. 2002, pp. 78–80), the parameters for the leave-S-out posterior approximation $\tilde{q}^{\backslash S}(\mathbf{w}) \propto \tilde{q}(\mathbf{w}) / \prod_{i \in S} \tilde{g}_i(\mathbf{w})$ are obtained according to the following formulas:

$$
\begin{aligned}
\mathbf{V}_w^{\backslash S} &= (\boldsymbol{\Phi}^{\mathsf{T}}\boldsymbol{\Lambda}^{-1}\boldsymbol{\Phi} - \boldsymbol{\Phi}_S^{\mathsf{T}}\boldsymbol{\Lambda}_S^{-1}\boldsymbol{\Phi}_S + \mathbf{V}^{-1})^{-1} = (\mathbf{V}_w^{-1} - \boldsymbol{\Phi}_S^{\mathsf{T}}\boldsymbol{\Lambda}_S^{-1}\boldsymbol{\Phi}_S)^{-1}\\
&= \mathbf{V}_w + \mathbf{V}_w\boldsymbol{\Phi}_S^{\mathsf{T}}(\boldsymbol{\Lambda}_S - \boldsymbol{\Phi}_S\mathbf{V}_w\boldsymbol{\Phi}_S^{\mathsf{T}})^{-1}\boldsymbol{\Phi}_S\mathbf{V}_w, && \text{(A11)}\\
\mathbf{m}_w^{\backslash S} &= \mathbf{V}_w^{\backslash S}(\boldsymbol{\Phi}^{\mathsf{T}}\boldsymbol{\Lambda}^{-1}\boldsymbol{\rho} - \boldsymbol{\Phi}_S^{\mathsf{T}}\boldsymbol{\Lambda}_S^{-1}\boldsymbol{\rho}_S)\\
&= \mathbf{V}_w^{\backslash S}\left((\mathbf{V}_w^{-1} - \boldsymbol{\Phi}_S^{\mathsf{T}}\boldsymbol{\Lambda}_S^{-1}\boldsymbol{\Phi}_S)\mathbf{V}_w + \boldsymbol{\Phi}_S^{\mathsf{T}}\boldsymbol{\Lambda}_S^{-1}\boldsymbol{\Phi}_S\mathbf{V}_w\right)\boldsymbol{\Phi}^{\mathsf{T}}\boldsymbol{\Lambda}^{-1}\boldsymbol{\rho} - \mathbf{V}_w^{\backslash S}\boldsymbol{\Phi}_S^{\mathsf{T}}\boldsymbol{\Lambda}_S^{-1}\boldsymbol{\rho}_S\\
&= \mathbf{V}_w^{\backslash S}\left((\mathbf{V}_w^{\backslash S})^{-1}\mathbf{V}_w + \boldsymbol{\Phi}_S^{\mathsf{T}}\boldsymbol{\Lambda}_S^{-1}\boldsymbol{\Phi}_S\mathbf{V}_w\right)\boldsymbol{\Phi}^{\mathsf{T}}\boldsymbol{\Lambda}^{-1}\boldsymbol{\rho} - \mathbf{V}_w^{\backslash S}\boldsymbol{\Phi}_S^{\mathsf{T}}\boldsymbol{\Lambda}_S^{-1}\boldsymbol{\rho}_S\\
&= \mathbf{V}_w\boldsymbol{\Phi}^{\mathsf{T}}\boldsymbol{\Lambda}^{-1}\boldsymbol{\rho} + \mathbf{V}_w^{\backslash S}\boldsymbol{\Phi}_S^{\mathsf{T}}\boldsymbol{\Lambda}_S^{-1}\boldsymbol{\Phi}_S\mathbf{V}_w\boldsymbol{\Phi}^{\mathsf{T}}\boldsymbol{\Lambda}^{-1}\boldsymbol{\rho} - \mathbf{V}_w^{\backslash S}\boldsymbol{\Phi}_S^{\mathsf{T}}\boldsymbol{\Lambda}_S^{-1}\boldsymbol{\rho}_S\\
&= \mathbf{m}_w + \mathbf{V}_w^{\backslash S}\boldsymbol{\Phi}_S^{\mathsf{T}}\boldsymbol{\Lambda}_S^{-1}(\boldsymbol{\Phi}_S\mathbf{m}_w - \boldsymbol{\rho}_S). && \text{(A12)}
\end{aligned}
$$

These formulas are used also later to estimate the leave-one-subject-out predictive performance of the classifier. To obtain the leave-i-out posterior approximation $\tilde{q}^{\setminus i}(\mathbf{w}) \propto \tilde{q}(\mathbf{w})/\tilde{g}_i(\mathbf{w})$ needed for expectation propagation, select $S = \{i\}$:

$$\mathbf{V}_w^{\setminus i} = \mathbf{V}_w + \frac{(\mathbf{V}_w t_i \mathbf{x}_i)(\mathbf{V}_w t_i \mathbf{x}_i)^{\mathsf{T}}}{\sigma_i - t_i \mathbf{x}_i^{\mathsf{T}} \mathbf{V}_w t_i \mathbf{x}_i}, \tag{A13}$$

$$\mathbf{m}_w^{\setminus i} = \mathbf{m}_w + (\mathbf{V}_w^{\setminus i} t_i \mathbf{x_i}) \sigma_i^{-1} (t_i \mathbf{x}_i^{\mathsf{T}} \mathbf{m}_w - \rho_i). \tag{A14}$$

For further calculations, however, it is more convenient to deal with a linear transformation $f_i(\mathbf{w}) = t_i \mathbf{w}^{\mathsf{T}} \mathbf{x}_i = \mathbf{w}^{\mathsf{T}} t_i \mathbf{x}_i$. Since the leave-$i$-out posterior approximation $\tilde{q}^{\setminus i}(\mathbf{w}) = \mathcal{N}(\mathbf{w}; \mathbf{m}_w^{\setminus i}, \mathbf{V}_w^{\setminus i})$ is a multivariate Gaussian over $\mathbf{w}$, the corresponding marginal distribution over $f_i(\mathbf{w})$ is also Gaussian with parameters $\mathrm{E}[f_i(\mathbf{w})] = a_i^{\setminus i} = (\mathbf{m}_w^{\setminus i})^{\mathsf{T}} t_i \mathbf{x}_i$ and $\mathrm{Var}[f_i(\mathbf{w})] = b_i^{\setminus i} = t_i \mathbf{x}_i^{\mathsf{T}} \mathbf{V}_w^{\setminus i} t_i \mathbf{x}_i$, implying

$$\tilde{q}^{\setminus i}(\mathbf{w})\, \mathrm{d}\mathbf{w} = \mathcal{N}(\mathbf{w}; \mathbf{m}_w^{\setminus i}, \mathbf{V}_w^{\setminus i})\, \mathrm{d}\mathbf{w} = \mathcal{N}(f_i(\mathbf{w}); a_i^{\setminus i}, b_i^{\setminus i})\, \mathrm{d}f_i(\mathbf{w}). \tag{A15}$$

Thus, at this point it is only necessary to store scalars $a_i^{\setminus i}$ and $b_i^{\setminus i}$, which can be calculated directly in terms of $a_i = \mathbf{m}_w^{\mathsf{T}} t_i \mathbf{x}_i$ and $b_i = t_i \mathbf{x}_i^{\mathsf{T}} \mathbf{V}_w t_i \mathbf{x}_i$ by rewriting equations A13 and A14:

$$b_i^{\setminus i} = b_i + \frac{b_i^2}{\sigma_i - b_i}, \tag{A16}$$

$$a_i^{\setminus i} = a_i + \frac{b_i^{\setminus i}(a_i - \rho_i)}{\sigma_i}. \tag{A17}$$

Minimisation of KL-divergence

Because the posterior approximation $\tilde{q}(\mathbf{w})$ is constrained to be Gaussian, minimising the Kullback-Leibler divergence

$$\mathrm{D}_{\mathrm{KL}}(\hat{q}(\mathbf{w}) \parallel \tilde{q}^*(\mathbf{w})) = \int_{\mathbf{w}} \hat{q}(\mathbf{w}) \frac{\hat{q}(\mathbf{w})}{\tilde{q}^*(\mathbf{w})}\, \mathrm{d}\mathbf{w} \tag{A18}$$

between the target distribution

$$\hat{q}(\mathbf{w}) = \frac{1}{Z_i} g_i(\mathbf{w}) \tilde{q}^{\setminus i}(\mathbf{w}) = \frac{1}{Z_i} \Psi(t_i \mathbf{w}^{\mathsf{T}} \mathbf{x}_i) \mathcal{N}(\mathbf{w}; \mathbf{m}_w^{\setminus i}, \mathbf{V}_w^{\setminus i}) \tag{A19}$$

and the new posterior approximation

$$\tilde{q}^*(\mathbf{w}) = \mathcal{N}(\mathbf{w}; \mathbf{m}_w^*, \mathbf{V}_w^*) = \frac{1}{Z_i} \tilde{g}_i{}^*(\mathbf{w}) \tilde{q}^{\setminus i}(\mathbf{w}) \tag{A20}$$

is equivalent to matching the first two moments of the distributions by setting $\mathbf{m}_w^* = \mathrm{E}_{\hat{q}}[\mathbf{w}]$ and $\mathbf{V}_w^* = \mathrm{Cov}_{\hat{q}}[\mathbf{w}]$ (Rasmussen and Williams 2006, pp. 203–204). To avoid working out multivariate integrals over $\mathbf{w}$, I use again the linear transformation $f_i(\mathbf{w})$, ending up in

$$\tilde{q}^*(\mathbf{w})\, \mathrm{d}\mathbf{w} = \mathcal{N}(\mathbf{w}; \mathbf{m}_w^*, \mathbf{V}_w^*)\, \mathrm{d}\mathbf{w} = \mathcal{N}(f_i(\mathbf{w}); a_i^*, b_i^*)\, \mathrm{d}f_i(\mathbf{w}), \tag{A21}$$

where $a_i^* = (\mathbf{m}_w^*)^\mathsf{T} t_i \mathbf{x}_i = (\mathrm{E}_{\hat{q}}[\mathbf{w}])^\mathsf{T} t_i \mathbf{x}_i$ and $b_i^* = t_i \mathbf{x}_i^\mathsf{T} \mathbf{V}_w^* t_i \mathbf{x}_i = t_i \mathbf{x}_i^\mathsf{T} (\mathrm{Cov}_{\hat{q}}[\mathbf{w}]) t_i \mathbf{x}_i$. Thus, to figure out the optimal new term approximation $\tilde{g}_i{}^*(\mathbf{w})$, it is sufficient to only derive expressions for a_i^*, b_i^* and Z_i. The normalisation constant Z_i is first obtained from $\int_\mathbf{w} \hat{q}(\mathbf{w})\,\mathrm{d}\mathbf{w} = 1$ by using the same formula as for equation 30, derived in the book by Rasmussen and Williams (2006, p. 74):

$$\begin{aligned} Z_i &= \int_\mathbf{w} \Psi(t_i \mathbf{w}^\mathsf{T} \mathbf{x}_i) \mathcal{N}(\mathbf{w}; \mathbf{m}_w^{\setminus i}, \mathbf{V}_w^{\setminus i})\,\mathrm{d}\mathbf{w} = \int_{-\infty}^{\infty} \Psi(f_i(\mathbf{w})) \mathcal{N}(f_i(\mathbf{w}); a_i^{\setminus i}, b_i^{\setminus i})\,\mathrm{d}f_i(\mathbf{w}) \\ &= \Psi\left(\frac{a_i^{\setminus i}}{\sqrt{1 + b_i^{\setminus i}}}\right) = \Psi(z_i), \end{aligned} \tag{A22}$$

where

$$z_i = \frac{a_i^{\setminus i}}{\sqrt{1 + b_i^{\setminus i}}}. \tag{A23}$$

Scalars a_i^* and b_i^* are obtained, respectively, by using further derivations by Rasmussen and Williams (2006, p. 75):

$$\begin{aligned} a_i^* &= (\mathrm{E}_{\hat{q}}[\mathbf{w}])^\mathsf{T} t_i \mathbf{x}_i = \int_\mathbf{w} \frac{1}{Z_i} \Psi(t_i \mathbf{w}^\mathsf{T} \mathbf{x}_i) \mathcal{N}(\mathbf{w}; \mathbf{m}_w^{\setminus i}, \mathbf{V}_w^{\setminus i}) \mathbf{w}^\mathsf{T} t_i \mathbf{x}_i\,\mathrm{d}\mathbf{w} \\ &= \int_{-\infty}^{\infty} \frac{1}{Z_i} \Psi(f_i(\mathbf{w})) \mathcal{N}(f_i(\mathbf{w}); a_i^{\setminus i}, b_i^{\setminus i}) f_i(\mathbf{w})\,\mathrm{d}f_i(\mathbf{w}) = a_i^{\setminus i} + \frac{b_i^{\setminus i} \mathcal{N}(z_i; 0, 1)}{\Psi(z_i)\sqrt{1 + b_i^{\setminus i}}} \\ &= a_i^{\setminus i} + \alpha_i b_i^{\setminus i}, \end{aligned} \tag{A24}$$

$$\begin{aligned} b_i^* &= t_i \mathbf{x}_i^\mathsf{T} (\mathrm{Cov}_{\hat{q}}[\mathbf{w}]) t_i \mathbf{x}_i = t_i \mathbf{x}_i^\mathsf{T} \left(\mathrm{E}_{\hat{q}}[\mathbf{w}\mathbf{w}^\mathsf{T}] - (\mathrm{E}_{\hat{q}}[\mathbf{w}])(\mathrm{E}_{\hat{q}}[\mathbf{w}])^\mathsf{T} \right) t_i \mathbf{x}_i \\ &= \int_\mathbf{w} \frac{1}{Z_i} \Psi(t_i \mathbf{w}^\mathsf{T} \mathbf{x}_i) \mathcal{N}(\mathbf{w}; \mathbf{m}_w^{\setminus i}, \mathbf{V}_w^{\setminus i}) t_i \mathbf{x}_i^\mathsf{T} \mathbf{w}\mathbf{w}^\mathsf{T} t_i \mathbf{x}_i\,\mathrm{d}\mathbf{w} - t_i \mathbf{x}_i^\mathsf{T} (\mathrm{E}_{\hat{q}}[\mathbf{w}])(\mathrm{E}_{\hat{q}}[\mathbf{w}])^\mathsf{T} t_i \mathbf{x}_i \\ &= \int_{-\infty}^{\infty} \frac{1}{Z_i} \Psi(f_i(\mathbf{w})) \mathcal{N}(f_i(\mathbf{w}); a_i^{\setminus i}, b_i^{\setminus i})\,(f_i(\mathbf{w}))^2\,\mathrm{d}f_i(\mathbf{w}) - (a_i^*)^2 \\ &= b_i^{\setminus i} - \frac{(b_i^{\setminus i})^2 \mathcal{N}(z_i; 0, 1)}{(1 + b_i^{\setminus i})\Psi(z_i)} \left(z_i + \frac{\mathcal{N}(z_i; 0, 1)}{\Psi(z_i)} \right) \\ &= b_i^{\setminus i} - \frac{(b_i^{\setminus i})^2 \mathcal{N}(z_i; 0, 1)}{(1 + b_i^{\setminus i})\Psi(z_i)} \left(\frac{a_i^{\setminus i}}{\sqrt{1 + b_i^{\setminus i}}} + \frac{(1 + b_i^{\setminus i}) \mathcal{N}(z_i; 0, 1)}{\sqrt{1 + b_i^{\setminus i}}\Psi(z_i)\sqrt{1 + b_i^{\setminus i}}} \right) \\ &= b_i^{\setminus i} - \frac{(b_i^{\setminus i})^2}{(1 + b_i^{\setminus i})} \frac{\mathcal{N}(z_i; 0, 1)}{\Psi(z_i)\sqrt{1 + b_i^{\setminus i}}} \left(a_i^{\setminus i} + \frac{b_i^{\setminus i} \mathcal{N}(z_i; 0, 1)}{\Psi(z_i)\sqrt{1 + b_i^{\setminus i}}} + \frac{\mathcal{N}(z_i; 0, 1)}{\Psi(z_i)\sqrt{1 + b_i^{\setminus i}}} \right) \\ &= b_i^{\setminus i} - \frac{(b_i^{\setminus i})^2}{(1 + b_i^{\setminus i})} \alpha_i \left(a_i^* + \alpha_i \right), \end{aligned} \tag{A25}$$

where

$$\alpha_i = \frac{\mathcal{N}(z_i; 0, 1)}{\Psi(z_i)\sqrt{1 + b_i^{\setminus i}}}. \tag{A26}$$

New Site Parameters

The new term approximation $\tilde{g_i}^*(\mathbf{w})$ and expressions for the new site parameters $(\rho_i^*, \sigma_i^*, \varsigma_i^*)$ are now obtained by using equations A20, A21 and A15:

$$
\begin{aligned}
\tilde{g_i}^*(\mathbf{w}) &= \frac{Z_i \tilde{q}^*(\mathbf{w})}{\tilde{q}^{\backslash i}(\mathbf{w})} = \frac{Z_i \tilde{q}^*(\mathbf{w})\,\mathrm{d}\mathbf{w}}{\tilde{q}^{\backslash i}(\mathbf{w})\,\mathrm{d}\mathbf{w}} = \frac{Z_i \mathcal{N}(f_i; a_i^*, b_i^*)\,\mathrm{d}f_i(\mathbf{w})}{\mathcal{N}(f_i; a_i^{\backslash i}, b_i^{\backslash i})\,\mathrm{d}f_i(\mathbf{w})} = \frac{Z_i \mathcal{N}(f_i; a_i^*, b_i^*)}{\mathcal{N}(f_i; a_i^{\backslash i}, b_i^{\backslash i})} \\
&= \frac{Z_i \frac{1}{\sqrt{2\pi b_i^*}} \mathrm{e}^{-\frac{1}{2}(b_i^*)^{-1}(f_i - a_i^*)^2}}{\frac{1}{\sqrt{2\pi b_i^{\backslash i}}} \mathrm{e}^{-\frac{1}{2}(b_i^{\backslash i})^{-1}(f_i - a_i^{\backslash i})^2}} \\
&= Z_i \sqrt{\frac{(b_i^*)^{-1}}{(b_i^{\backslash i})^{-1}}} \mathrm{e}^{-\frac{1}{2}\left(\left((b_i^*)^{-1} - (b_i^{\backslash i})^{-1}\right) f_i^2 - 2\left((b_i^*)^{-1} a_i^* - (b_i^{\backslash i})^{-1} a_i^{\backslash i}\right) f_i + (b_i^*)^{-1}(a_i^*)^2 - (b_i^{\backslash i})^{-1}(a_i^{\backslash i})^2\right)} \\
&= Z_i \sqrt{\frac{(b_i^*)^{-1}}{(b_i^{\backslash i})^{-1}}} \mathrm{e}^{-\frac{1}{2}\left((\sigma_i^*)^{-1} f_i(\mathbf{w})^2 - 2(\sigma_i^*)^{-1}\rho_i^* f_i(\mathbf{w}) + (b_i^*)^{-1}(a_i^*)^2 - (b_i^{\backslash i})^{-1}(a_i^{\backslash i})^2\right)} \\
&= Z_i \sqrt{\frac{(b_i^*)^{-1}}{(b_i^{\backslash i})^{-1}}} \mathrm{e}^{-\frac{1}{2}\left((b_i^*)^{-1}(a_i^*)^2 - (b_i^{\backslash i})^{-1}(a_i^{\backslash i})^2 - (\sigma_i^*)^{-1}(\rho_i^*)^2\right)} \mathrm{e}^{-\frac{1}{2}(\sigma_i^*)^{-1}(f_i(\mathbf{w}) - \rho_i^*)^2} \\
&= \varsigma_i^* \mathrm{e}^{-\frac{1}{2\sigma_i^*}(t_i \mathbf{w}^\mathsf{T} \mathbf{x}_i - \rho_i^*)^2},
\end{aligned}
\tag{A27}
$$

where

$$
\begin{aligned}
\sigma_i^* &= \left((b_i^*)^{-1} - (b_i^{\backslash i})^{-1}\right)^{-1} = \frac{b_i^{\backslash i} b_i^*}{b_i^{\backslash i} - b_i^*} = \frac{b_i^{\backslash i}\left(b_i^{\backslash i} - \frac{(b_i^{\backslash i})^2}{(1 + b_i^{\backslash i})}\alpha_i(a_i^* + \alpha_i)\right)}{b_i^{\backslash i} - \left(b_i^{\backslash i} - \frac{(b_i^{\backslash i})^2}{(1 + b_i^{\backslash i})}\alpha_i(a_i^* + \alpha_i)\right)} \\
&= \frac{1 + b_i^{\backslash i}}{\alpha_i(a_i^* + \alpha_i)} - b_i^{\backslash i},
\end{aligned}
\tag{A28}
$$

$$
\begin{aligned}
\rho_i^* &= \sigma_i^* \left((b_i^*)^{-1} a_i^* - (b_i^{\backslash i})^{-1} a_i^{\backslash i}\right) = \sigma_i^* \left(\left((b_i^{\backslash i})^{-1} + (\sigma_i^*)^{-1}\right)(a_i^{\backslash i} + \alpha_i b_i^{\backslash i}) - (b_i^{\backslash i})^{-1} a_i^{\backslash i}\right) \\
&= a_i^{\backslash i} + \alpha_i b_i^{\backslash i} + \alpha_i \sigma_i^* = a_i^* + \alpha_i \sigma_i^*,
\end{aligned}
\tag{A29}
$$

$$
\begin{aligned}
\varsigma_i^* &= Z_i \sqrt{\frac{(b_i^*)^{-1}}{(b_i^{\backslash i})^{-1}}} \mathrm{e}^{-\frac{1}{2}\left((b_i^*)^{-1}(a_i^*)^2 - (b_i^{\backslash i})^{-1}(a_i^{\backslash i})^2 - (\sigma_i^*)^{-1}(\rho_i^*)^2\right)} \\
&= \Psi(z_i) \sqrt{\frac{(b_i^{\backslash i})^{-1} + (\sigma_i^*)^{-1}}{(b_i^{\backslash i})^{-1}}} \mathrm{e}^{-\frac{1}{2}\left(\left((b_i^{\backslash i})^{-1} + (\sigma_i^*)^{-1}\right)(a_i^*)^2 - (b_i^{\backslash i})^{-1}(a_i^* - \alpha_i b_i^{\backslash i})^2 - (\sigma_i^*)^{-1}(a_i^* + \alpha_i \sigma_i^*)^2\right)} \\
&= \Psi(z_i) \sqrt{1 + \frac{b_i^{\backslash i}}{\sigma_i^*}} \mathrm{e}^{\frac{1}{2}\left(\alpha_i^2 b_i^{\backslash i} + \alpha_i^2 \sigma_i^*\right)} = \Psi(z_i) \sqrt{1 + \frac{b_i^{\backslash i}}{\sigma_i^*}} \mathrm{e}^{\frac{1}{2}\left(\alpha_i^2 b_i^{\backslash i} + \alpha_i^2\left(\frac{1 + b_i^{\backslash i}}{\alpha_i(a_i^* + \alpha_i)} - b_i^{\backslash i}\right)\right)} \\
&= \Psi(z_i) \sqrt{1 + \frac{b_i^{\backslash i}}{\sigma_i^*}} \mathrm{e}^{\frac{1}{2}\alpha_i \frac{1 + b_i^{\backslash i}}{a_i^* + \alpha_i}}.
\end{aligned}
\tag{A30}
$$

New Posterior Parameters

Finally, to update parameters for the posterior approximation, the new term approximation $\tilde{g}_i{}^*(\mathbf{w})$ is put back together with the leave-i-out posterior approximation $\tilde{q}^{\setminus i}(\mathbf{w})$. An expression for the new variance $\mathbf{V}_w^*$ is obtained in a similar way as equation A13 by adding observation i back into equation A8 and using the Woodbury formula (Press et al. 2002, pp. 78–80):

$$\mathbf{V}_w^* = \left(\mathbf{V}_w^{\setminus i} + t_i\mathbf{x}_i(\sigma_i^*)^{-1}t_i\mathbf{x}_i^\mathsf{T}\right)^{-1} = \mathbf{V}_w^{\setminus i} - \frac{(\mathbf{V}_w^{\setminus i}t_i\mathbf{x}_i)(\mathbf{V}_w^{\setminus i}t_i\mathbf{x}_i)^\mathsf{T}}{\sigma_i^* + t_i\mathbf{x}_i^\mathsf{T}\mathbf{V}_w^{\setminus i}t_i\mathbf{x}_i}. \tag{A31}$$

The new mean $\mathbf{m}_w^*$ can be directly solved from equation A24 by substituting $a_i^{\setminus i} = (\mathbf{m}_w^*)^\mathsf{T}t_i\mathbf{x}_i$ and $b_i^* = t_i\mathbf{x}_i^\mathsf{T}\mathbf{V}_w^*t_i\mathbf{x}_i$:

$$\mathbf{m}_w^* = \mathbf{m}_w^{\setminus i} + \alpha_i\mathbf{V}_w^{\setminus i}t_i\mathbf{x}_i. \tag{A32}$$

These can be computed efficiently by defining the following auxiliary vectors, connected by rewriting equation A13:

$$\mathbf{c}_i = \mathbf{V}_w t_i\mathbf{x}_i, \tag{A33}$$

$$\mathbf{c}_i^{\setminus i} = \mathbf{V}_w^{\setminus i}t_i\mathbf{x}_i = \mathbf{c}_i + \frac{\mathbf{c}_i b_i}{\sigma_i - b_i} = \mathbf{c}_i\left(1 + \frac{b_i}{\sigma_i - b_i}\right). \tag{A34}$$

The parameters $\mathbf{V}_w^*$ and $\mathbf{m}_w^*$ can now be written in the following forms:

$$\begin{aligned}
\mathbf{V}_w^* &= \mathbf{V}_w + \frac{\mathbf{c}_i\mathbf{c}_i^\mathsf{T}}{\sigma_i - b_i} - \frac{\mathbf{c}_i^{\setminus i}(\mathbf{c}_i^{\setminus i})^\mathsf{T}}{\sigma_i^* + b_i^{\setminus i}} \\
&= \mathbf{V}_w + \frac{\mathbf{c}_i\mathbf{c}_i^\mathsf{T}}{\sigma_i - b_i} - \frac{\mathbf{c}_i\mathbf{c}_i^\mathsf{T}}{\frac{1+b_i^{\setminus i}}{\alpha_i(a_i^*+\alpha_i)} - b_i^{\setminus i} + b_i^{\setminus i}}\left(1 + \frac{b_i}{\sigma_i - b_i}\right)^2 \\
&= \mathbf{V}_w + \left(\frac{1}{\sigma_i - b_i} - \frac{\alpha_i(a_i^* + \alpha_i)}{1 + b_i^{\setminus i}}\left(1 + \frac{b_i}{\sigma_i - b_i}\right)^2\right)\mathbf{c}_i\mathbf{c}_i^\mathsf{T},
\end{aligned} \tag{A35}$$

$$\mathbf{m}_w^* = \mathbf{m}_w + \mathbf{c}_i^{\setminus i}(\sigma_i)^{-1}(a_i - \rho_i) + \alpha_i\mathbf{c}_i^{\setminus i} = \mathbf{m}_w + \left(\frac{a_i - \rho_i}{\sigma_i} + \alpha_i\right)\mathbf{c}_i^{\setminus i}. \tag{A36}$$

Marginal Likelihood

After the algorithm has converged and found stable parameters for the approximations of the likelihood and the posterior, an approximation for the marginal likelihood can be computed as the normalisation constant $\tilde{Z}_P$ in equation A2:

$$\begin{aligned}
p(\mathbf{t}|\mathbf{X},\mathbf{v}) &= \frac{p(\mathbf{t}|\mathbf{w},\mathbf{X})p(\mathbf{w}|\mathbf{v})}{p(\mathbf{w}|\mathbf{t},\mathbf{X},\mathbf{v})} = Z_P \\
&\approx \tilde{Z}_P = \left(\prod_{i=1}^{N}\varsigma_i\right)|\mathbf{V}|^{-\frac{1}{2}}\mathrm{e}^{-\frac{1}{2}(\boldsymbol{\rho}^\mathsf{T}\boldsymbol{\Lambda}^{-1}\boldsymbol{\rho}-\mathbf{m}_w^\mathsf{T}\mathbf{V}_w^{-1}\mathbf{m}_w)}|\mathbf{V}_w|^{\frac{1}{2}}.
\end{aligned} \tag{A37}$$

This can be computed efficiently by using the Cholesky decomposition (Press et al. 2002, pp. 99–101) of $\mathbf{V}_w^{-1} = \mathbf{\Phi}^\mathsf{T}\mathbf{\Lambda}^{-1}\mathbf{\Phi} + \mathbf{V}^{-1} = \mathbf{L}_1\mathbf{L}_1^\mathsf{T}$ and taking the natural logarithm:

$$\begin{aligned}
p(\mathbf{t}|\mathbf{X},\mathbf{v}) &\approx \left(\prod_{i=1}^{N}\varsigma_i\right)\left(\prod_{j=1}^{D}v_j\right)^{-\frac{1}{2}} |\mathbf{V}_w^{-1}|^{-\frac{1}{2}}\mathrm{e}^{\frac{1}{2}(\mathbf{m}_w^\mathsf{T}\mathbf{V}_w^{-1}\mathbf{m}_w - \sum_{i=1}^{N}\frac{\rho_i^2}{\sigma_i})} \\
&= \left(\prod_{i=1}^{N}\varsigma_i\right)\left(\prod_{j=1}^{D}v_j\right)^{-\frac{1}{2}} |\mathbf{L}_1\mathbf{L}_1^\mathsf{T}|^{-\frac{1}{2}}\mathrm{e}^{\frac{1}{2}\left(\mathbf{m}_w^\mathsf{T}(\mathbf{L}_1\mathbf{L}_1^\mathsf{T})\mathbf{m}_w - \sum_{i=1}^{N}\frac{\rho_i^2}{\sigma_i}\right)} \\
&= \left(\prod_{i=1}^{N}\varsigma_i\right)\left(\prod_{j=1}^{D}v_j\right)^{-\frac{1}{2}} |\mathbf{L}_1|^{-1}\mathrm{e}^{\frac{1}{2}\left((\mathbf{L}_1^\mathsf{T}\mathbf{m}_w)^\mathsf{T}(\mathbf{L}_1^\mathsf{T}\mathbf{m}_w) - \sum_{i=1}^{N}\frac{\rho_i^2}{\sigma_i}\right)},
\end{aligned} \tag{A38}$$

$$\ln p(\mathbf{t}|\mathbf{X},\mathbf{v}) \approx \sum_{i=1}^{N}\ln\varsigma_i - \frac{1}{2}\sum_{j=1}^{D}\ln v_j - \ln|\mathbf{L}_1| + \frac{1}{2}\left((\mathbf{L}_1^\mathsf{T}\mathbf{m}_w)^\mathsf{T}(\mathbf{L}_1^\mathsf{T}\mathbf{m}_w) - \sum_{i=1}^{N}\frac{\rho_i^2}{\sigma_i}\right).$$

Estimates for Predictive Performance

In addition to the marginal likelihood, EP offers also an opportunity to estimate the leave-one-out predictive performance without carrying out the actual cross-validation. The two measures introduced in section 4.2, predictive classification accuracy (equation 31) and mean log predictive probability (equation 33), are now applied for the training data itself by using the corresponding leave-i-out posterior approximation for each observation i. The following estimates for leave-one-out CA and leave-one-out MLPP require only the auxiliary variables z_i, for each observation $i = 1, \ldots, N$:

$$\tilde{\mathrm{CA}}_{\mathrm{loo}} = \frac{1}{N}\sum_{i=1}^{N}\mathcal{H}\left(\frac{t_i(\mathbf{m}_w^{\backslash i})^\mathsf{T}\mathbf{x}_i}{\sqrt{1+\mathbf{x}_i^\mathsf{T}\mathbf{V}_w^{\backslash i}\mathbf{x}_i}}\right) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{H}(z_i), \tag{A39}$$

$$\mathrm{ML\tilde{P}P}_{\mathrm{loo}} = \frac{1}{N}\sum_{i=1}^{N}\ln\Psi\left(\frac{t_i(\mathbf{m}_w^{\backslash i})^\mathsf{T}\mathbf{x}_i}{\sqrt{1+\mathbf{x}_i^\mathsf{T}\mathbf{V}_w^{\backslash i}\mathbf{x}_i}}\right) = \frac{1}{N}\sum_{i=1}^{N}\ln\Psi(z_i). \tag{A40}$$

If the data includes several observations from each subject, it is more appropriate to replace $\tilde{\mathrm{CA}}_{\mathrm{loo}}$ and $\mathrm{ML\tilde{P}P}_{\mathrm{loo}}$ with the corresponding leave-one-subject-out estimates. Denote S_k as the set of observations i that belong to subject k and use equations A11 and A12 to obtain the leave-S_k-out posterior approximation parameters $\mathbf{V}_w^{\backslash S_k}$ and $\mathbf{m}_w^{\backslash S_k}$ for each subject $k = 1, \ldots, K$. The computation is enhanced by defining $\mathbf{C}_{S_k} = \mathbf{\Phi}_{S_k}\mathbf{V}_w$ and by using again the Cholesky decomposition (Press et al. 2002, pp. 99–101) of $\mathbf{\Lambda}_{S_k} - \mathbf{\Phi}_{S_k}\mathbf{V}_w\mathbf{\Phi}_{S_k}^\mathsf{T} = \mathbf{\Lambda}_{S_k} - \mathbf{C}_{S_k}\mathbf{\Phi}_{S_k}^\mathsf{T} = \mathbf{L}_2\mathbf{L}_2^\mathsf{T}$:

$$\begin{aligned}
\mathbf{V}_w^{\backslash S_k} &= \mathbf{V}_w + \mathbf{V}_w\mathbf{\Phi}_{S_k}^\mathsf{T}(\mathbf{\Lambda}_{S_k} - \mathbf{\Phi}_{S_k}\mathbf{V}_w\mathbf{\Phi}_{S_k}^\mathsf{T})^{-1}\mathbf{\Phi}_{S_k}\mathbf{V}_w \\
&= \mathbf{V}_w + \mathbf{C}_{S_k}^\mathsf{T}(\mathbf{L}_2\mathbf{L}_2^\mathsf{T})^{-1}\mathbf{C}_{S_k} = \mathbf{V}_w + (\mathbf{L}_2^{-1}\mathbf{C}_{S_k})^\mathsf{T}(\mathbf{L}_2^{-1}\mathbf{C}_{S_k}),
\end{aligned} \tag{A41}$$

$$\mathbf{m}_w^{\backslash S_k} = \mathbf{m}_w + \mathbf{V}_w^{\backslash S_k}\mathbf{\Phi}_{S_k}^\mathsf{T}\mathbf{\Lambda}_{S_k}^{-1}(\mathbf{\Phi}_{S_k}\mathbf{m}_w - \boldsymbol{\rho}_{S_k}). \tag{A42}$$

The estimates for leave-one-subject-out CA (equation 51) and leave-one-subject-out MLPP (equation 52) can be now computed:

$$\tilde{\mathrm{CA}}_{\mathrm{loso}} = \frac{1}{N}\sum_{k=1}^{K}\sum_{i\in S_k}\mathcal{H}\left(\frac{t_i(\mathbf{m}_w^{\setminus S_k})^{\mathsf{T}}\mathbf{x}_i}{\sqrt{1+\mathbf{x}_i^{\mathsf{T}}\mathbf{V}_w^{\setminus S_k}\mathbf{x}_i}}\right) = \frac{1}{N}\sum_{k=1}^{K}\sum_{i\in S_k}\mathcal{H}(z_i^{\setminus S_k}), \tag{A43}$$

$$\mathrm{ML\tilde{P}P}_{\mathrm{loso}} = \frac{1}{N}\sum_{k=1}^{K}\sum_{i\in S_k}\ln\Psi\left(\frac{t_i(\mathbf{m}_w^{\setminus S_k})^{\mathsf{T}}\mathbf{x}_i}{\sqrt{1+\mathbf{x}_i^{\mathsf{T}}\mathbf{V}_w^{\setminus S_k}\mathbf{x}_i}}\right) = \frac{1}{N}\sum_{k=1}^{K}\sum_{i\in S_k}\ln\Psi(z_i^{\setminus S_k}), \tag{A44}$$

where

$$z_i^{\setminus S_k} = \frac{t_i(\mathbf{m}_w^{\setminus S_k})^{\mathsf{T}}\mathbf{x}_i}{\sqrt{1+\mathbf{x}_i^{\mathsf{T}}\mathbf{V}_w^{\setminus S_k}\mathbf{x}_i}}. \tag{A45}$$

B MAP Estimate for a Single Relevance Hyperparameter in ARDEP

The fast sequential hyperparameter optimisation scheme used in the ARDEP approximate inference method (see 5.1 Automatic Relevance Determination by Expectation Propagation) searches for a MAP estimate for the relevance hyperparameter vector $\mathbf{v}$ by maximising $p(\mathbf{t}|\mathbf{X},\mathbf{v})p(\mathbf{v}|\lambda)$ with respect to one hyperparameter at a time, using the approximate expression produced by the previous EP run. In this appendix, I derive in detail the update rules for a single hyperparameter v_j.

As described in subsection 5.1.4, ARDEP runs EP with sparsified input hyperparameter vector $\bar{\mathbf{v}}$ (or $\bar{\mathbf{V}}$ as a diagonal matrix form) and data matrix $\bar{\mathbf{\Phi}}$ including only features $m \in F$, where $F = \{m : v_m > 0\}$. When deriving the optimal v_j below, however, I stick to dealing with all the D features, assuming that they are all positive, but may still be infinitesimal with the same effect as being equal to zero. This effect is easily seen by denoting the m^{th} column of the data matrix $\mathbf{\Phi}$ as $\boldsymbol{\phi}_m = (t_1[\mathbf{x}_1]_m, \ldots, t_N[\mathbf{x}_N]_m)^\mathsf{T}$ and rewriting $p(\mathbf{t}|\mathbf{X},\mathbf{v})p(\mathbf{v}|\lambda)$ with respect to

$$\mathbf{\Omega} = \mathbf{\Lambda}+\mathbf{\Phi}\mathbf{V}\mathbf{\Phi}^\mathsf{T} = \mathbf{\Lambda}+\sum_{m=1}^{D} \boldsymbol{\phi}_m v_m \boldsymbol{\phi}_m^\mathsf{T} \to \mathbf{\Lambda}+\sum_{m\in F} \boldsymbol{\phi}_m v_m \boldsymbol{\phi}_m^\mathsf{T} = \mathbf{\Lambda}+\bar{\mathbf{\Phi}}\bar{\mathbf{V}}\bar{\mathbf{\Phi}}^\mathsf{T} = \bar{\mathbf{\Omega}}. \tag{B1}$$

Using the Woodbury formula (Press et al. 2002, pp. 78–80) and equation 45, the inverse of $\mathbf{\Omega}$ becomes

$$\begin{aligned}\mathbf{\Omega}^{-1} &= \left(\mathbf{\Lambda}+\mathbf{\Phi}\mathbf{V}\mathbf{\Phi}^\mathsf{T}\right)^{-1} = \mathbf{\Lambda}^{-1} - \mathbf{\Lambda}^{-1}\mathbf{\Phi}\left(\mathbf{\Phi}^\mathsf{T}\mathbf{\Lambda}^{-1}\mathbf{\Phi}+\mathbf{V}^{-1}\right)^{-1}\mathbf{\Phi}^\mathsf{T}\mathbf{\Lambda}^{-1}\\ &= \mathbf{\Lambda}^{-1} - \mathbf{\Lambda}^{-1}\mathbf{\Phi}\mathbf{V}_w\mathbf{\Phi}^\mathsf{T}\mathbf{\Lambda}^{-1},\end{aligned} \tag{B2}$$

and the approximate expression for $p(\mathbf{t}|\mathbf{X},\mathbf{v})p(\mathbf{v}|\lambda)$ can be rewritten from equation 47:

$$\begin{aligned}p(\mathbf{t}|\mathbf{X},\mathbf{v})p(\mathbf{v}|\lambda) &\approx \left(\prod_{i=1}^{N}\varsigma_i\right)|\mathbf{V}|^{-\frac{1}{2}}\mathrm{e}^{-\frac{1}{2}(\boldsymbol{\rho}^\mathsf{T}\mathbf{\Lambda}^{-1}\boldsymbol{\rho}-\mathbf{m}_w^\mathsf{T}\mathbf{V}_w^{-1}\mathbf{m}_w)}|\mathbf{V}_w|^{\frac{1}{2}}\prod_{j=1}^{D}\frac{1}{2\lambda^2}\mathrm{e}^{-\frac{v_j}{2\lambda^2}}\\ &= \left(\prod_{i=1}^{N}\varsigma_i\right)(2\lambda^2)^{-D}\left(\frac{|\mathbf{V}_w^{-1}|}{|\mathbf{V}^{-1}|}\right)^{-\frac{1}{2}}\mathrm{e}^{-\frac{1}{2}\left(\boldsymbol{\rho}^\mathsf{T}\mathbf{\Lambda}^{-1}\boldsymbol{\rho}-(\mathbf{V}_w\mathbf{\Phi}^\mathsf{T}\mathbf{\Lambda}^{-1}\boldsymbol{\rho})^\mathsf{T}\mathbf{V}_w^{-1}(\mathbf{V}_w\mathbf{\Phi}^\mathsf{T}\mathbf{\Lambda}^{-1}\boldsymbol{\rho})\right)}\prod_{j=1}^{D}\mathrm{e}^{-\frac{v_j}{2\lambda^2}}\\ &= \left(\prod_{i=1}^{N}\varsigma_i\right)(2\lambda^2)^{-D}\left(\frac{|\mathbf{\Phi}^\mathsf{T}\mathbf{\Lambda}^{-1}\mathbf{\Phi}+\mathbf{V}^{-1}|}{|\mathbf{V}^{-1}|}\right)^{-\frac{1}{2}}\mathrm{e}^{-\frac{1}{2}\boldsymbol{\rho}^\mathsf{T}(\mathbf{\Lambda}^{-1}-\mathbf{\Lambda}^{-1}\mathbf{\Phi}\mathbf{V}_w\mathbf{\Phi}^\mathsf{T}\mathbf{\Lambda}^{-1})\boldsymbol{\rho}}\prod_{j=1}^{D}\mathrm{e}^{-\frac{v_j}{2\lambda^2}}\\ &= \left(\prod_{i=1}^{N}\varsigma_i\right)(2\lambda^2)^{-D}\left(\frac{|\mathbf{\Lambda}+\mathbf{\Phi}\mathbf{V}\mathbf{\Phi}^\mathsf{T}||\Lambda^{-1}||\mathbf{V}^{-1}|}{|\mathbf{V}^{-1}|}\right)^{-\frac{1}{2}}\mathrm{e}^{-\frac{1}{2}\boldsymbol{\rho}^\mathsf{T}(\mathbf{\Lambda}+\mathbf{\Phi}\mathbf{V}\mathbf{\Phi}^\mathsf{T})^{-1}\boldsymbol{\rho}}\prod_{j=1}^{D}\mathrm{e}^{-\frac{v_j}{2\lambda^2}}\\ &= \left(\prod_{i=1}^{N}\varsigma_i\right)|\mathbf{\Lambda}|^{\frac{1}{2}}(2\lambda^2)^{-D}|\mathbf{\Omega}|^{-\frac{1}{2}}\mathrm{e}^{-\frac{1}{2}\boldsymbol{\rho}^\mathsf{T}\mathbf{\Omega}^{-1}\boldsymbol{\rho}}\prod_{j=1}^{D}\mathrm{e}^{-\frac{v_j}{2\lambda^2}}.\end{aligned} \tag{B3}$$

To separate the terms depending on v_j, decompose $\mathbf{\Omega}$ into

$$\mathbf{\Omega} = \mathbf{\Lambda}+\sum_{m\neq j}\boldsymbol{\phi}_m v_m \boldsymbol{\phi}_m^\mathsf{T} + \boldsymbol{\phi}_j v_j \boldsymbol{\phi}_j^\mathsf{T} = \mathbf{\Omega}^{\backslash j} + \boldsymbol{\phi}_j v_j \boldsymbol{\phi}_j^\mathsf{T}, \tag{B4}$$

where

$$\boldsymbol{\Omega}^{\backslash j} = \boldsymbol{\Lambda} + \sum_{m \neq j} \boldsymbol{\phi}_m v_m \boldsymbol{\phi}_m^{\mathsf{T}}. \tag{B5}$$

The inverse and logarithm of $\boldsymbol{\Omega}$ decompose, respectively, as follows:

$$\boldsymbol{\Omega}^{-1} = (\boldsymbol{\Omega}^{\backslash j} + \boldsymbol{\phi}_j v_j \boldsymbol{\phi}_j^{\mathsf{T}})^{-1} = (\boldsymbol{\Omega}^{\backslash j})^{-1} - \frac{(\boldsymbol{\Omega}^{\backslash j})^{-1}\boldsymbol{\phi}_j\boldsymbol{\phi}_j^{\mathsf{T}}(\boldsymbol{\Omega}^{\backslash j})^{-1}}{v_j^{-1} + \boldsymbol{\phi}_j^{\mathsf{T}}(\boldsymbol{\Omega}^{\backslash j})^{-1}\boldsymbol{\phi}_j}, \tag{B6}$$

$$\begin{aligned}\ln|\boldsymbol{\Omega}| &= \ln|\boldsymbol{\Omega}^{\backslash j} + \boldsymbol{\phi}_j v_j \boldsymbol{\phi}_j^{\mathsf{T}}| = \ln\left(|\boldsymbol{\Omega}^{\backslash j}|\left(1 + \frac{\boldsymbol{\phi}_j^{\mathsf{T}}(\boldsymbol{\Omega}^{\backslash j})^{-1}\boldsymbol{\phi}_j}{v_j^{-1}}\right)\right) \\ &= \ln|\boldsymbol{\Omega}^{\backslash j}| + \ln\left(1 + \boldsymbol{\phi}_j^{\mathsf{T}}(\boldsymbol{\Omega}^{\backslash j})^{-1}\boldsymbol{\phi}_j v_j\right).\end{aligned} \tag{B7}$$

Maximising $p(\mathbf{t}|\mathbf{X}, \mathbf{v})p(\mathbf{v}|\lambda)$ is equivalent to maximising its logarithm, which can now be written in the following form:

$$\begin{aligned}\mathcal{L}(\mathbf{v}) =& \sum_{i=1}^{N} \ln \varsigma_i + \frac{1}{2}\ln|\boldsymbol{\Lambda}| - D\ln(2\lambda^2) - \frac{1}{2}\ln|\boldsymbol{\Omega}| - \frac{1}{2}\boldsymbol{\rho}^{\mathsf{T}}\boldsymbol{\Omega}^{-1}\boldsymbol{\rho} - \frac{1}{2\lambda^2}\sum_{m=1}^{D} v_m \\ =& \sum_{i=1}^{N} \ln \varsigma_i + \frac{1}{2}\ln|\boldsymbol{\Lambda}| - D\ln(2\lambda^2) - \frac{1}{2}\left(\ln|\boldsymbol{\Omega}^{\backslash j}| + \ln\left(1 + \boldsymbol{\phi}_j^{\mathsf{T}}(\boldsymbol{\Omega}^{\backslash j})^{-1}\boldsymbol{\phi}_j v_j\right)\right) \\ & - \frac{1}{2}\boldsymbol{\rho}^{\mathsf{T}}\left((\boldsymbol{\Omega}^{\backslash j})^{-1} - \frac{(\boldsymbol{\Omega}^{\backslash j})^{-1}\boldsymbol{\phi}_j\boldsymbol{\phi}_j^{\mathsf{T}}(\boldsymbol{\Omega}^{\backslash j})^{-1}}{v_j^{-1} + \boldsymbol{\phi}_j^{\mathsf{T}}(\boldsymbol{\Omega}^{\backslash j})^{-1}\boldsymbol{\phi}_j}\right)\boldsymbol{\rho} - \frac{1}{2\lambda^2}\sum_{m \neq j} v_m - \frac{1}{2\lambda^2}v_j \\ =& \sum_{i=1}^{N} \ln \varsigma_i + \frac{1}{2}\ln|\boldsymbol{\Lambda}| - D\ln(2\lambda^2) - \frac{1}{2}\ln|\boldsymbol{\Omega}^{\backslash j}| - \frac{1}{2}\boldsymbol{\rho}^{\mathsf{T}}(\boldsymbol{\Omega}^{\backslash j})^{-1}\boldsymbol{\rho} - \frac{1}{2\lambda^2}\sum_{m \neq j} v_m \\ & - \frac{1}{2}\ln\left(1 + \boldsymbol{\phi}_j^{\mathsf{T}}(\boldsymbol{\Omega}^{\backslash j})^{-1}\boldsymbol{\phi}_j v_j\right) + \frac{1}{2}\frac{\left(\boldsymbol{\phi}_j^{\mathsf{T}}(\boldsymbol{\Omega}^{\backslash j})^{-1}\boldsymbol{\rho}\right)^2 v_j}{\left(1 + \boldsymbol{\phi}_j^{\mathsf{T}}(\boldsymbol{\Omega}^{\backslash j})^{-1}\boldsymbol{\phi}_j v_j\right)} - \frac{1}{2\lambda^2}v_j \\ =& \mathcal{L}(\mathbf{v}^{\backslash j}) - \frac{1}{2}\ln(1 + r_j v_j) + \frac{h_j^2}{2}\frac{v_j}{(1 + r_j v_j)} - \frac{1}{2\lambda^2}v_j,\end{aligned} \tag{B8}$$

where

$$r_j = \boldsymbol{\phi}_j^{\mathsf{T}}(\boldsymbol{\Omega}^{\backslash j})^{-1}\boldsymbol{\phi}_j, \tag{B9}$$

$$h_j = \boldsymbol{\phi}_j^{\mathsf{T}}(\boldsymbol{\Omega}^{\backslash j})^{-1}\boldsymbol{\rho}. \tag{B10}$$

To analyse $\mathcal{L}(\mathbf{v})$ with respect to a single hyperparameter v_j, I calculate first the corresponding gradient:

$$\begin{aligned}\frac{\partial\mathcal{L}(\mathbf{v})}{\partial v_j} &= -\frac{r_j}{2}\frac{1}{(1 + r_j v_j)} + \frac{h_j^2}{2}\frac{(1 + r_j v_j) - v_j r_j}{(1 + r_j v_j)^2} - \frac{1}{2\lambda^2} \\ &= \frac{-\lambda^2 r_j - \lambda^2 r_j^2 v_j + \lambda^2 h_j^2 - 1 - 2r_j v_j - r_j^2 v_j^2}{2\lambda^2(1 + r_j v_j)^2} \\ &= \frac{(-r_j^2)v_j^2 + (-\lambda^2 r_j^2 - 2r_j)v_j + (-\lambda^2 r_j + \lambda^2 h_j^2 - 1)}{2\lambda^2(1 + r_j v_j)^2}.\end{aligned} \tag{B11}$$

Since the nominator of the above expression is a parabola opening downward and the denominator always positive, the gradient $\frac{\partial \mathcal{L}(\mathbf{v})}{\partial v_j}$ is positive between its roots and negative before and after them. To find the roots, the gradient is set to be equal to zero:

$$\begin{aligned}
&\frac{\partial \mathcal{L}(\mathbf{v})}{\partial v_j} = 0 \iff \\
&v_j = \frac{-(-\lambda^2 r_j^2 - 2r_j) \pm \sqrt{(-\lambda^2 r_j^2 - 2r_j)^2 - 4(-r_j^2)(-\lambda^2 r_j + \lambda^2 h_j^2 - 1)}}{2(-r_j^2)} \\
&= \frac{-\lambda^2 r_j^2 - 2r_j \pm \sqrt{\lambda^4 r_j^4 + 4\lambda^2 r_j^3 + 4r_j^2 - 4\lambda^2 r_j^3 + 4\lambda^2 r_j^2 h_j^2 - 4r_j^2}}{2r_j^2} \\
&= -\frac{\lambda^2}{2} - \frac{1}{r_j} \pm \sqrt{\frac{\lambda^4}{4} + \frac{\lambda^2 h_j^2}{r_j^2}}. \qquad \text{(B12)}
\end{aligned}$$

Because $\mathbf{\Omega}^{\setminus j}$ is a symmetric and positive-definite matrix, $r_j = \boldsymbol{\phi}_j^{\mathsf{T}}(\mathbf{\Omega}^{\setminus j})^{-1}\boldsymbol{\phi}_j$ is always positive. Thus, $-\frac{\lambda^2}{2} - \frac{1}{r_j} - \sqrt{\frac{\lambda^4}{4} + \frac{\lambda^2 h_j^2}{r_j^2}}$ is always negative. Since v_j can have only positive values, there are now two possible alternatives for the global maximum of $\mathcal{L}(\mathbf{v})$, depending on the sign of

$$\hat{v}_j = -\frac{\lambda^2}{2} - \frac{1}{r_j} + \sqrt{\frac{\lambda^4}{4} + \frac{\lambda^2 h_j^2}{r_j^2}}, \qquad \text{(B13)}$$

which is determined by the following inequality:

$$\begin{aligned}
&\hat{v}_j = -\frac{\lambda^2}{2} - \frac{1}{r_j} + \sqrt{\frac{\lambda^4}{4} + \frac{\lambda^2 h_j^2}{r_j^2}} > 0 \iff \frac{\lambda^4}{4} + \frac{\lambda^2 h_j^2}{r_j^2} > \frac{\lambda^4}{4} + \frac{\lambda^2}{r_j} + \frac{1}{r_j^2} \iff \\
&h_j^2 - r_j - \frac{1}{\lambda^2} > 0. \qquad \text{(B14)}
\end{aligned}$$

If $h_j^2 - r_j - \frac{1}{\lambda^2} > 0$, $\mathcal{L}(\mathbf{v})$ has its maximum at $v_j = \hat{v}_j = -\frac{\lambda^2}{2} - \frac{1}{r_j} + \sqrt{\frac{\lambda^4}{4} + \frac{\lambda^2 h_j^2}{r_j^2}}$. If instead $h_j^2 - r_j - \frac{1}{\lambda^2} \leq 0$, $\mathcal{L}(\mathbf{v})$ increases monotonically as v_j decreases and approaches its maximum at the limit $v_j \to 0$.

In practice, r_j and h_j can be computed efficiently by using the current value of v_j and variables $R_j = \boldsymbol{\phi}_j^{\mathsf{T}}\mathbf{\Omega}^{-1}\boldsymbol{\phi}_j \to \boldsymbol{\phi}_j^{\mathsf{T}}\bar{\mathbf{\Omega}}^{-1}\boldsymbol{\phi}_j$ and $H_j = \boldsymbol{\phi}_j^{\mathsf{T}}\mathbf{\Omega}^{-1}\boldsymbol{\rho} \to \boldsymbol{\phi}_j^{\mathsf{T}}\bar{\mathbf{\Omega}}^{-1}\boldsymbol{\rho}$. Using equations B2 and 46, these can be written directly with respect to the sparsified $\bar{\mathbf{m}}_w$ and $\bar{\mathbf{V}}_w$, produced by an EP run with the sparsified input hyperparameter vector $\bar{\mathbf{v}}$ and data matrix $\bar{\mathbf{\Phi}}$:

$$\begin{aligned}
R_j &\to \boldsymbol{\phi}_j^{\mathsf{T}}\bar{\mathbf{\Omega}}^{-1}\boldsymbol{\phi}_j = \boldsymbol{\phi}_j^{\mathsf{T}}\mathbf{\Lambda}^{-1}\boldsymbol{\phi}_j - \boldsymbol{\phi}_j^{\mathsf{T}}\mathbf{\Lambda}^{-1}\bar{\mathbf{\Phi}}\bar{\mathbf{V}}_w\bar{\mathbf{\Phi}}^{\mathsf{T}}\mathbf{\Lambda}^{-1}\boldsymbol{\phi}_j, \qquad \text{(B15)} \\
H_j &\to \boldsymbol{\phi}_j^{\mathsf{T}}\bar{\mathbf{\Omega}}^{-1}\boldsymbol{\rho} = \boldsymbol{\phi}_j^{\mathsf{T}}\mathbf{\Lambda}^{-1}\boldsymbol{\rho} - \boldsymbol{\phi}_j^{\mathsf{T}}\mathbf{\Lambda}^{-1}\bar{\mathbf{\Phi}}\bar{\mathbf{V}}_w\bar{\mathbf{\Phi}}^{\mathsf{T}}\mathbf{\Lambda}^{-1}\boldsymbol{\rho} \\
&= \boldsymbol{\phi}_j^{\mathsf{T}}\mathbf{\Lambda}^{-1}\boldsymbol{\rho} - \boldsymbol{\phi}_j^{\mathsf{T}}\mathbf{\Lambda}^{-1}\bar{\mathbf{\Phi}}\bar{\mathbf{m}}_w. \qquad \text{(B16)}
\end{aligned}$$

By using the decomposition in equation B6, we get

$$R_j = \boldsymbol{\phi}_j^{\mathsf{T}} \boldsymbol{\Omega}^{-1} \boldsymbol{\phi}_j = \boldsymbol{\phi}_j^{\mathsf{T}} (\boldsymbol{\Omega}^{\setminus j})^{-1} \boldsymbol{\phi}_j - \frac{\left(\boldsymbol{\phi}_j^{\mathsf{T}} (\boldsymbol{\Omega}^{\setminus j})^{-1} \boldsymbol{\phi}_j\right)^2}{v_j^{-1} + \boldsymbol{\phi}_j^{\mathsf{T}} (\boldsymbol{\Omega}^{\setminus j})^{-1} \boldsymbol{\phi}_j} = r_j - \frac{v_j r_j^2}{1 + v_j r_j} \Longleftrightarrow$$

$$R_j + v_j R_j r_j = r_j + v_j r_j^2 - v_j r_j^2 \Longleftrightarrow r_j = \frac{R_j}{1 - v_j R_j} \tag{B17}$$

and

$$H_j = \boldsymbol{\phi}_j^{\mathsf{T}} \boldsymbol{\Omega}^{-1} \boldsymbol{\rho} = \boldsymbol{\phi}_j^{\mathsf{T}} (\boldsymbol{\Omega}^{\setminus j})^{-1} \boldsymbol{\rho} - \frac{\boldsymbol{\phi}_j^{\mathsf{T}} (\boldsymbol{\Omega}^{\setminus j})^{-1} \boldsymbol{\phi}_j \boldsymbol{\phi}_j^{\mathsf{T}} (\boldsymbol{\Omega}^{\setminus j})^{-1} \boldsymbol{\rho}}{v_j^{-1} + \boldsymbol{\phi}_j^{\mathsf{T}} (\boldsymbol{\Omega}^{\setminus j})^{-1} \boldsymbol{\phi}_j} = h_j - \frac{v_j r_j h_j}{1 + v_j r_j}$$

$$= h_j - \frac{v_j \frac{R_j}{1 - v_j R_j} h_j}{1 + v_j \frac{R_j}{1 - v_j R_j}} = h_j - v_j R_j h_j \Longleftrightarrow h_j = \frac{H_j}{1 - v_j R_j}. \tag{B18}$$

If the current v_j has been set equal to zero and removed from the model, it is noticed both from the above expressions and directly from equation B4, that r_j and h_j reduce to R_j and H_j.