

Preference based personalization of hearing aids

Jens Brehm Nielsen - s042189

Kongens Lyngby 2010
IMM-M.Sc.-2010-61

Technical University of Denmark
Informatics and Mathematical Modelling
Building 321, DK-2800 Kongens Lyngby, Denmark
Phone +45 45253351, Fax +45 45882673
reception@imm.dtu.dk
www.imm.dtu.dk

Abstract

The procedure involved in fitting hearing aids has become highly extensive, due to the vast number of parameters in modern hearing aids. An interactive system that automatically optimizes the hearing aid setting for individual users is an interesting alternative in comparison with manual hearing aid fitting procedures.

In this thesis, an iterative interactive framework for personalization of hearing aids based on user preferences is presented. For a particular user, the framework models a preference function over hearing aid settings with a Gaussian process based on a minimum of observations. An observation is a subjective rating of the *overall preference* of the processed sound resulting from a particular hearing aid setting. New observations are suggested based a novel active learning criterion developed in this project. With the novel active learning criterion the next subjectively rated setting becomes the setting for which the preference has the highest probability of being larger than the preference for the currently preferred setting given a Gaussian process estimated preference function.

Simulations and a pilot experiment show that the framework discovers a personalized setting in few iterations compared with the number of possible settings. Furthermore, the framework has the capability to model complex preference functions, although an improved interactive experimental paradigm is required to account for inconsistent subjective preference assessments.

Resumé

Den procedure, der kræves for at tilpasse høreapparater, er blevet særdeles omfattende pga. det store antal parametre i moderne høreapparater. Et interaktivt system, som automatisk optimerer høreapparatsindstillinger for individuelle brugere, er et interessant alternativ til manuelle høreapparats tilpasningsprocedurer.

I dette speciale præsenteres en interaktiv metode til præference baseret høreapparatspersonliggørelse. For en given bruger modelleres en præferencefunktion over høreapparatsindstillinger med en Gaussisk process baseret på et minimum af observationer. En observation er en subjektiv vurdering af den *overordnet præference* af den resulterende lyd givet en specifik høreapparatsindstilling. Nye observationer foreslås baseret på et nyt aktivt læringskriterium, som er udviklet i dette projekt. Med det nye aktive læringskriterium bliver den næste subjektive vurderede indstilling, den indstilling for hvilken præferencen har den største sandsynlighed for at være større end præferencen for den nuværende foretrukne indstilling givet en Gaussisk process estimeret præferencefunktion.

Simuleringer og et pilot forsøg viser, at metoden finder en personlig indstilling efter få iterationer sammenlignet med antallet af mulige indstillinger. Endvidere har metoden evnen til at modellere komplekse præferencefunktioner, selvom et forbedret interaktivt forsøgsparadigme er nødvendigt for at tage højde for inkonsistente subjektive præference vurderinger.

Preface

This thesis was prepared at the Cognitive Systems group at DTU Informatics, Technical University of Denmark, in partial fulfillment of the requirements for acquiring the Master of Science degree in Electrical Engineering. The project was conducted in cooperation with Widex A/S in the period from December 1st, 2009, to August 20th, 2010. 40 percent of the time was spent at the facilities of Widex A/S in Værløse (until February 2010) and Vassingerød (from February 2010) and 60 percent of the time was spent at the facilities of DTU Informatics in Kongens Lyngby. The workload corresponded to 40 ECTS points.

Supervisors:

- Associate Professor Jan Larsen, Department of Informatics and Mathematical Modeling
- Ph.D. student Bjørn Sand Jensen, Department of Informatics and Mathematical Modeling
- Research Engineer Georg Stiefenhofer, Audiological Research & Innovation department at Widex A/S

Acknowledgements

First of all, I owe great thanks to Associate Professor Jan Larsen at Department of Informatics and Mathematical Modeling, who agreed to be my supervisor. The meetings and discussions regarding technical issues during the project were a great help and his engagement in the outcome of the thesis has been very motivating.

Next, I want to express my deepest appreciation of the help and support given during the entire project by Ph.D. student Bjørn Sand Jensen at the Cognitive System group. Without his guidance and extensive support the outcome of the project would not have been taken to the same level.

In addition, I want to give great thanks to Research Engineer Georg Stiefenhofer at the Audiological Research & Innovation department at Widex A/S, who has put a great effort into this project. He always took time to discuss sudden problems and to help solving them.

Finally, I want to thank Widex A/S, and especially Group Leader Morten Nordahn at the Audiological Research & Innovation department at Widex A/S, for given me the opportunity to do my Master thesis in cooperation with Widex A/S.

Contents

| | |
|---|------------|
| Abstract | i |
| Resumé | iii |
| Preface | v |
| Acknowledgements | vii |
| 1 Introduction | 1 |
| 1.1 Motivation | 1 |
| 1.2 Project Aim | 2 |
| 1.3 Structure | 3 |
| 2 Background | 5 |
| 2.1 Typical Hearing Aid Fitting Procedure | 5 |
| 2.2 Perceptual and Performance Measures | 6 |
| 3 Machine Learning Theory | 9 |
| 3.1 Bayesian Learning | 10 |
| 3.2 Gaussian Processes for Regression | 15 |
| 4 Active Learning Theory | 23 |
| 4.1 Maximize Total Information Gain | 24 |
| 4.2 Minimize Generalization Error | 24 |
| 4.3 Optimize for Maximum Preference | 25 |
| 4.4 2D Simulation | 28 |

| | | |
|----------|--|-----------|
| 5 | User-Driven Personalization Framework | 51 |
| 5.1 | Bayesian Utility Elicitation | 51 |
| 5.2 | Modeling Preference Functions | 56 |
| 5.3 | Transform Pair-Wise Judgments to Preferences | 57 |
| 5.4 | Active Learning | 63 |
| 5.5 | 2D Baseline Simulation | 63 |
| 5.6 | Summary | 68 |
| 6 | Pilot Experiment | 69 |
| 6.1 | Experimental Method | 69 |
| 6.2 | Results | 75 |
| 6.3 | Discussion | 85 |
| 7 | Future Work | 89 |
| 7.1 | Experimental Paradigm | 89 |
| 7.2 | Altering the Hyper-Parameters | 90 |
| 7.3 | Induce a Global Search and Stop Criterion | 90 |
| 7.4 | Preference Elicitation Strategy | 91 |
| 7.5 | Test-Retest Experiments | 91 |
| 7.6 | Use the Algorithm in a Realistic Setup with Hearing Impaired Subjects | 92 |
| 7.7 | Cluster Hearing Impaired Subjects based on the Obtained Pref- erence Function | 92 |
| 7.8 | Use Cluster Preference Function as Prior over Functions in the GP | 92 |
| 8 | Conclusion | 95 |
| A | Relevant Experimental Data from the Pilot Experiment | 97 |
| A.1 | JBN | 97 |
| A.2 | GST | 101 |
| A.3 | OHA | 104 |
| A.4 | AWE | 107 |

Introduction

1.1 Motivation

Entering the digital era has been a major breakthrough for the hearing aid (HA) industry, and it has extensively increased the sound processing possibilities in a HA. Consequently, a wide range of algorithms that do potentially improve the HA performance have been developed over the recent decades. Improved noise suppression algorithms, advanced compressors, adaptive beam-formers and context classifiers to mention a few. However, a comprehensive amount of free adjustable HA parameters has emerged from these improvements making the HA fitting procedure increasingly complex.

It is believed that bad HA performance is often associated with improper fitting and therefore it is believed that more intelligent and efficient HA fitting methods can have a positive effect on user satisfaction. In addition, current fitting paradigms do not account for individual user preferences among hearing impaired (HI) persons, but is merely concerned with hearing loss (HL) compensation. Because of the fitting complexity the dispenser is left with little freedom for individual user personalization and the quality of the fitting and the degree of personalization are strongly affected by the commitment of the dispenser.

Recent studies have confirmed that individual user preferences do exist. Başkent

et al. (2007) used a *Genetic Algorithm* (GA) to find individual personal preferences among subjects for the setting of a vocoder including the number of vocoder channels, the amount of spectral shift and the amount of spectral compression/expansion. The study showed that subjects obtained different solutions and generally preferred their own solution among the solutions obtained by other subjects. Durant et al. (2004) also used a GA to adjust a feed-back canceler to individual preferences and showed that subjects generally preferred the individualized setting found by the GA over settings found for other subjects. Since fitting rationales do not account for individual preferences, the need for a simple user-driven personalization approach in the fitting procedure arises. However, elicitation of subjective preferences among hearing aid settings is in practice not trivial. As an example, Ricketts and Hornsby (2005) compared speech recognition results and sound quality results for speech in background noise with and without a noise suppression algorithm applied. The results showed that the speech recognition scores were almost unaffected by the introduction of the noise suppression algorithm, but the sound quality increased considerably with the presence of the noise suppression algorithm.

1.2 Project Aim

Recent studies (Başkent et al., 2007; Durant et al., 2004; Ricketts and Hornsby, 2005) show that user preferences can be captured and give reason to believe that the quality of the HA fitting due to increased personalization can be improved. Therefore, the overall aim of this study is to investigate the possibilities for an intelligent user-driven active-learning method with a simple interactive user interface, used to capture individual user preferences for a subset of HA parameters and discover the optimal setting for individual subjects.

The concept is to model a preference function over HA settings based on a minimum of observations of subjectively assessed preference values for particular HA settings. The observations are made iteratively and individual observations are chosen actively to discover the optimal setting of the HAs. The strategy is to study relevant *machine learning* and *active learning* theory, resulting in a developed baseline framework. Since, an optimal individualized setting is essentially not known in advance for any test subject, simulations are used to estimate performance. Finally, the implemented method is tested through a pilot experiment and evaluated with respect to robustness and convergence time. Furthermore, the validity of the estimated preference functions is discussed, in conjugation with the advantages and disadvantages of the method.

1.3 Structure

Initially, in chapter 2 useful background information is presented. In chapter 3 the relevant machine learning theory is explained, followed by chapter 4 containing the active learning theory including a novel active learning concept developed in this project and a simulation study which verifies the method. The baseline algorithm is proposed in chapter 5 and results from the pilot experiments are presented in chapter 6. Following this, future work and research areas are discussed in chapter 7. Finally, in chapter 8 the conclusion is contained.

Background

In this chapter background concepts within HA fitting, preference judgments, perceptual measures and psycho-acoustics will be presented to make the reader aware of factors that are important for HA personalization. However, these concepts are not within the main focus of this project. The concepts presented in the this chapter are important, but it has been outside the scope of this project to include them thoroughly.

2.1 Typical Hearing Aid Fitting Procedure

This section briefly describes the traditional HA fitting methods (for further details see for instance Dillon (2001)). Currently, HA fitting is based mainly upon prescriptive methods called *rationales*, which map hearing threshold measurements (audiograms) to a target gain in a given frequency range.

Initially, the type and degree of HL are determined based on a measured audiogram. Dependent on the HL the dispenser chooses the most suitable HA style for the HI. Each style defines some limits and possibilities in terms of ease of insertion, visibility, amount of gain, sensitivity to wind noise, directivity, telephone compatibility and avoidance of occlusion and feedback (Dillon, 2001,

chapter 10). Some of these properties are associated with practicality, while others are directly associated with choice of features.

Next, the dispenser takes an imprint for the ear mold and orders the HAs from the manufacturer. When the HAs are received from the manufacturer, the dispenser is ready to fit the HAs. Some manufacturers have developed rationales specifically for their HAs including additional diagnosis besides the audiogram, e.g. *loudness recruitment*, *discomfort levels*, *cognitive skills* based on a questionnaire etc.

A very difficult and vague part of the fitting process is to decide whether or not a particular feature should be enabled and how it should be adjusted to satisfy the needs of the hearing impaired. Normally, the manufacturer has limited the flexibility such that the dispenser can only adjust meta-parameters, e.g. the degree of adaptive noise suppression, which then for each setting defines all the parameters in the adaptive noise reduction algorithm. The preferences concerning feature adjustment for individual subjects are very diverse and it can be difficult for the dispenser to elicit useful information about the optimal adjustment. In regards, a lot is based on dispenser intuition and experience, hence the quality of the feature adjustments can vary.

Additional fine-tuning is typically performed after a wearing period of approximately three months. Based upon the statements from the patient, the dispenser tries to deduce to what extent any imperfections require additional adjustments. For instance, new HA users can typically not tolerate the amount of high-frequency gain according to their prescriptions, because they perceive impact sounds as “too sharp”. Consequently, audiologists often provide less high-frequency gain in the initial fitting and increase it to the prescribed target gain when fine-tuning the HAs.

It is practically infeasible to accommodate improper fitting in all situations due to the restrictions regarding type of HA resulting from a particular HL combined with the individual attitude towards impairment and personal preferences about how a HA should sound. Therefore, trade-offs are inevitably.

2.2 Perceptual and Performance Measures

The main focus in this thesis is to develop a suitable algorithm for further preference optimization, assuming that it is possible to subjectively assess preference. In this process it is convenient to assume that there exists a one dimensional preference measure, i.e., an internal scale on which decisions is made favoring

one setting over another. Furthermore, it is assumed that this preference measure is a mixture of different attributes, such as *speech intelligibility*, *listening effort*, *sound quality* etc.

Realistically, this assumption will not be valid. Probably, user preference is related to a complex conjugation of attributes, hence to expect that preference can be captured in one perceptual measure is unrealistic. To make a simple decision regarding these issues, subjects are in this thesis supposed to provide their opinion about what they prefer in a completely general sense, without further instructions about what to focus on in given sound environment (context). This is referred to as *overall preference*. Presumably, this leads to inconsistencies in the user assessments, because test subjects will not always be fully aware of his or her intention. In addition, the intention by subjects are naturally affected by the context.

In the next sections three different attributes contributing to the overall preference are briefly reviewed. In general, there exist well-known methods to exploit these attributes alternative to an overall preference. However, for the sake of simplicity the overall preference measure is assumed in this thesis.

2.2.1 Speech Intelligibility

Speech intelligibility (SI) is a objective performance criterion describing how well a subject is able to understand the words that are pronounced. Normally, SI measurements are carried out with speech in background noise and measured as the percentage of correctly understood words resulting in a score. The score depends on the type and shape of the noise as well as the speech material itself. Generally, the score follows a psychometric curve as a function of the *Signal to Noise Ratio* (SNR).

Traditionally, SI scores were the dominating measure to describe performance of hearing impaired persons, since intuitively the goal of a HA system would be to improve the speech communication ability for the HI. However, SI is hardly a subjective preference. Instead, the perceptual experience of speech intelligibility is related to personal preference in noisy-speech environments, since HI subjects have an extensive cognitive load in these situations. Therefore, in speech recognition research the expression *listening effort* is a subjective alternative to speech intelligibility.

2.2.2 Listening Effort

Traditionally, listening effort is a perceptual measure related to the amount of cognitive load used by a test subject in a noisy speech environment. There does not exist any explicit definition of listening effort nor a standardized method of measuring it.

Typically, the measure is used subjectively, but Baddeley (1992) defines the term *working memory*, which assumes that in complex recognition tasks the brain has to simultaneously process and store information. Hence, if the brain uses a lot of its capacity to process, the working memory is reduced. This has been used to objectively measure cognitive loads in complex environment by performing tasks, where subjects are suppose to remember the first and last word of a sentence, while understanding the meaning as well (Andersson and Lyxell, 1999). Thus, it is possible to use listening effort as an objective performance measure by introducing the concepts of working memory.

2.2.3 Sound Quality

Sound quality is a perceptual measure referring to the overall quality of the presented stimulus, but can alternatively be used to subjectively rate particular features in the stimulus. Therefore, speech quality, spaciousness quality etc. are within the field of sound quality. For this reason there exist different standards for the assessment of sound quality dependent on the situation. The Telecommunication Standardization Sector of the International Telecommunication Union (ITU-T) has made a recommendation regarding the area of sound quality in speech communication systems that include noise suppression algorithms (ITU-T P.835, 2003). Also, ITU-T has developed a recommendations for assessment of quality for speech (ITU-T P.862, 2001).

Commonly, sound quality is measured as a subjective rating of the overall quality of the presented sound. Therefore, dependent on the user preferences and on the instructions given by the experimenter, sound quality results can vary among subjects.

Machine Learning Theory

Several practical problems arise in modern engineering where a predicted output f_* from an unknown system (*Machine*) given a new set of inputs $\mathbf{x}_* = \{x_1, x_2, \dots, x_D\}$, where D is the dimension of the input, is requested. The knowledge about the physical nature of the unknown system is typically limited. Instead, it might be possible to collect input-output (\mathbf{x}, y) measurements or observations (collected in a *data set* $\mathcal{D} = \{\mathbf{X}, \mathbf{y}\}$) for the unknown system. \mathbf{X} contains the inputs $\mathbf{x}_n, n = 1, 2, 3, \dots, N$ for the N number of observations and $\mathbf{y}_n, n = 1, 2, 3, \dots, N$ contains the corresponding noisy targets. To imitate the unknown system (*Learning*) and predict outputs f_* for new inputs \mathbf{x}_* , a mapping from inputs to outputs should be learned based in the measured or observed data set \mathcal{D} . Generally, (supervised) learning involves two parts:

- 1) **Model selection:** Selection of a particular model - *parameterized* or *non-parameterized*.
- 2) **Training:** Optimize model parameters (collected in the vector \mathbf{w}) given the data set \mathcal{D} .

The simplest model is a parameterized linear model $f(\mathbf{x}, \mathbf{w}) = \mathbf{x}^\top \mathbf{w}$, where the output is a linear combination of the inputs. For more complex systems, non-parameterized models using *kernels* are frequently used (referred to as *kernel*

machines). A kernel machine is a flexibly non-linear method, where no particular functional model form (parameterization) is specified. Instead, a *kernel function* is specified, which computes a scalar expressing the similarity between input points. A predicted output is determined by the similarity between the new input and all the observations through the kernel function. The parameters \mathbf{w} in the kernel function is referred to as *hyper-parameters*.

In this thesis, machine learning is used to model the preference by HA users for individual HA settings by mapping from HA settings to overall preference. This chapter will present and derive the machine learning theory relevant for this project. First, an introduction to Bayesian learning will be given in section 3.1. Following this, a thorough presentation of a *Gaussian Process* (GP) is presented in section 3.2, including a suitable extension developed in this thesis in section 3.2.3. Additional details about these concepts can be found in Bishop (2006) and Rasmussen and Williams (2006).

3.1 Bayesian Learning

Bayesian learning is a major area within probabilistic models and has emerged from *Bayes' theorem* which is directly obtained from the rule of factorization of joint probabilities $P(X, Y) = P(Y|X)P(X) = P(X|Y)P(Y)$,

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)}, \quad (3.1.1)$$

where X and Y are stochastic variables, $P(X)$ is the probability of X and $P(X|Y)$ is the conditional probability of X given Y . Bayes' theorem is also valid for multi dimensional continuous stochastic variables, in a machine learning context in terms of a model parameter set \mathbf{w} and an observed or measured data set \mathcal{D}

$$p(\mathbf{w}|\mathcal{D}) = \frac{p(\mathcal{D}|\mathbf{w})p(\mathbf{w})}{p(\mathcal{D})}, \quad (3.1.2)$$

where lowercase p refers to a probability density function. The term $p(\mathbf{w})$ in the nominator on the right hand side called the *prior* contains a priori information about the behavior of the parameters and the term $p(\mathcal{D}|\mathbf{w})$ called the *likelihood* expresses the probability of the data set \mathcal{D} as a function of the model parameters \mathbf{w} . The term $p(\mathcal{D})$ in the denominator on the right hand side is a normalization factor ensuring that the *posterior* distribution $p(\mathbf{w}|\mathcal{D})$ is a valid probability distribution that integrates to 1. By integrating both sides of equation (3.1.2)

with respect to \mathbf{w} the normalization factor can be rewritten

$$\begin{aligned}\int p(\mathbf{w}|\mathcal{D})d\mathbf{w} &= \int \frac{p(\mathcal{D}|\mathbf{w})p(\mathbf{w})}{p(\mathcal{D})}d\mathbf{w} \\ 1 &= \frac{1}{p(\mathcal{D})} \int p(\mathcal{D}|\mathbf{w})p(\mathbf{w})d\mathbf{w} \\ p(\mathcal{D}) &= \int p(\mathcal{D}|\mathbf{w})p(\mathbf{w})d\mathbf{w}\end{aligned}\tag{3.1.3}$$

The normalization term is in general called the *marginal likelihood*, due to the marginalization of the likelihood with respect to the parameters. Alternatively, the normalization term is referred to as the *model evidence*, since it expresses the evidence for one particular model given the observed data (Bishop, 2006, section 3.4). An illustrative example of the Bayesian formalism applied to a simple linear regression model is given in (Bishop, 2006, Figure 3.7).

To gain further insight into the behavior of the marginal likelihood, a simple approximation to the integral in equation (3.1.3) can be made. Assume that a model containing only one adaptive parameter w has a posterior distribution over parameters which is sharply peaked around the most probable value w_{MAP} and has a width $\Delta w_{posterior}$. Further, it is assumed that the prior is flat having a width of Δw_{prior} , hence $p(w) = 1/\Delta w_{prior}$. Then the marginal likelihood can be approximated by

$$p(\mathcal{D}) = \int p(\mathcal{D}|w)p(w)dw \simeq p(\mathcal{D}|w_{MAP}) \frac{\Delta w_{posterior}}{\Delta w_{prior}}.\tag{3.1.4}$$

Finally, taking the natural log to obtain

$$\ln p(\mathcal{D}) \simeq \ln p(\mathcal{D}|w_{MAP}) + \ln \frac{\Delta w_{posterior}}{\Delta w_{prior}}.\tag{3.1.5}$$

Due to the fact that probabilities are naturally always between zero and one, the first term is always less or equal to zero, and since the posterior will be more narrow than the prior the second term will be less or equal to one as well. Further, the second term will have a large magnitude if the posterior is closely tuned to the data. Hence, the second term can be interpreted as penalty or regularization term, which increases in magnitude as the posterior becomes more closely tuned compared to the prior. Expanding these assumption to models containing M adaptive parameters and assuming that all parameters have the same ratio of $\frac{\Delta w_{posterior}}{\Delta w_{prior}}$ the log marginal likelihood becomes

$$\ln p(\mathcal{D}) \simeq \ln p(\mathcal{D}|\mathbf{w}_{MAP}) + M \ln \frac{\Delta w_{posterior}}{\Delta w_{prior}}.\tag{3.1.6}$$

When a more complex model is used the data will normally be fitted more accurately, hence the first term will decrease in magnitude, but the second term

(regularization term) will in this simple approximation increase linearly with the number of parameters M . Thus, a Bayesian framework automatically embeds regularization of the model complexity and should ideally avoid over fitting, i.e., avoid fitting the noise in the data and instead estimate the function that has generated the data. This concept is illustrated in figure 3.1. More attention

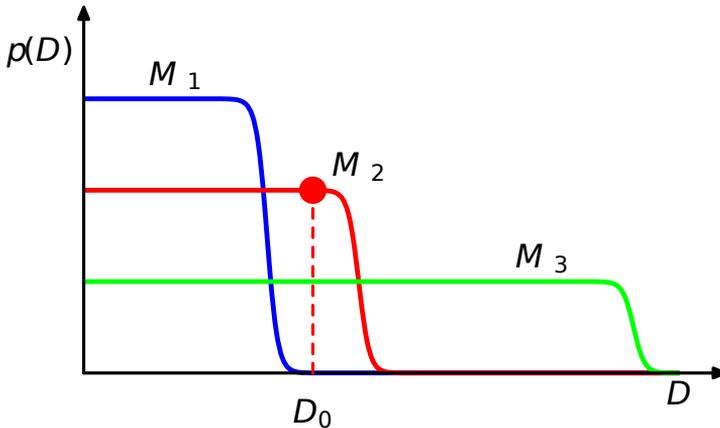


Figure 3.1: Marginal likelihood (y-axis) for three different models with different complexity in which M_1 is the simplest model. The x-axis expresses the complexity of the observed data. When the complexity increases the simple models fit the data poorly, hence the marginal likelihood suddenly decreases. The regularization term reduces the overall marginal likelihood for complex models. For a particular data set D_0 the model with intermediate complexity is favored by the marginal likelihood, because it is the simplest model that can fit the data. (Bishop, 2006, Figure 3.13).

will be drawn towards the marginal likelihood in section 3.2.4 when considering Gaussian Processes.

In cases of sequentially observed data a Bayesian framework can effectively be used to update the probabilities over model parameter settings every time new data becomes available. In such situations the concept of *conjugated priors* arises naturally (Bishop, 2006, section 2.4). Conjugated priors refer to distributions for which the posterior has the same functional form as the prior. Therefore, when looking for a conjugated prior it must be conjugated to the likelihood so that the posterior distribution has the same functional form. The exponential family is an example of commonly used conjugated priors. Generally, distributions contained in the exponential family have the form

$$p(\mathbf{x}|\eta) = h(\mathbf{x})g(\eta) \exp\{\eta^T \mathbf{u}(\mathbf{x})\}, \quad (3.1.7)$$

where \mathbf{x} can either be a scalar or a vector. η is referred to as the *natural parameters* of the distribution and $\mathbf{u}(\mathbf{x})$ and $h(\mathbf{x})$ is some function of \mathbf{x} . $g(\eta)$ ensures that the distribution is normalized and therefore satisfies

$$g(\eta) \int h(\mathbf{x}) \exp\{\eta^T \mathbf{u}(\mathbf{x})\} d\mathbf{x} = 1 \quad (3.1.8)$$

To model the natural parameters of the exponential family there exists a conjugated prior of the form

$$p(\eta|\chi, \nu) = f(\chi, \nu) g(\eta)^\nu \exp\{\nu \eta^T \chi\} \quad (3.1.9)$$

which multiplied with the likelihood function given by (Bishop, 2006, equation 2.227)

$$p(\mathbf{X}|\eta) = \left(\prod_{n=1}^N h(\mathbf{x}_n) \right) g(\eta)^N \exp \left\{ \eta^T \sum_{n=1}^N \mathbf{u}(\mathbf{x}_n) \right\} \quad (3.1.10)$$

gives

$$p(\eta|\mathbf{X}, \chi, \nu) \propto g(\eta)^{\nu+N} \exp \left\{ \eta^T \left(\sum_{n=1}^N \mathbf{u}(\mathbf{x}_n) + \nu \chi \right) \right\}. \quad (3.1.11)$$

Despite an normalization factor, this expression has the same functional form as the prior given by equation (3.1.9). It should be observed that the parameter ν can be interpreted as the effective number of pseudo-observations in the prior. Each observation has a value of the *sufficient statistic* $\mathbf{u}(\mathbf{x})$ given by χ (Bishop, 2006, section 2.4.1).

There exist different ways to make use of the posterior distribution $p(\mathbf{w}|\mathcal{D})$ over model parameters after the Bayesian inference. The simplest way to use the Bayesian inference is to use what is referred to as *maximum a posterior* estimate or simple the *MAP* estimate of the model parameters. In the MAP estimate the model parameters \mathbf{w}_{MAP} that maximize the posterior distribution are used as a point estimate in addition with the given model to make predictions for new inputs. To gain an understanding of the MAP estimate assume that the observed targets y have a Gaussian distribution with mean given by a model prediction $f(\mathbf{x}, \mathbf{w})$

$$p(y|\mathbf{x}, \mathbf{w}, \beta) = \mathcal{N}(y|f(\mathbf{x}, \mathbf{w}), \beta^{-1}) \quad (3.1.12)$$

$$= \sqrt{\frac{\beta}{2\pi}} \exp \left\{ -\frac{\beta}{2} [f(\mathbf{x}, \mathbf{w}) - y]^2 \right\}, \quad (3.1.13)$$

where \mathbf{x} is the input and the precision $\beta = 1/\sigma^2$ is equal to the inverse variance. The likelihood function $p(\mathbf{y}|\mathbf{X}, \mathbf{w}, \beta)$ will be given by the product rule if the

data is assumed to be drawn independently

$$p(\mathbf{y}|\mathbf{X}, \mathbf{w}, \beta) = \prod_{n=1}^N \mathcal{N}(y_n | f(\mathbf{x}_n, \mathbf{w}), \beta^{-1}) \quad (3.1.14)$$

$$= \left(\frac{\beta}{2\pi}\right)^{N/2} \exp\left\{-\frac{\beta}{2} \sum_{n=1}^N [f(\mathbf{x}_n, \mathbf{w}) - y_n]^2\right\}. \quad (3.1.15)$$

where \mathbf{X} is the matrix containing all the N input vectors $\mathbf{x}_n, n = 1, 2, 1, \dots, N$ and \mathbf{y} is the vector containing the N corresponding targets $y_n, n = 1, 2, 3, \dots, N$. Next, put a zero-mean Gaussian distribution with precision α over the model parameters

$$p(\mathbf{w}|\alpha) = \mathcal{N}(\mathbf{w}|\mathbf{0}, \alpha^{-1}\mathbf{I}) \quad (3.1.16)$$

$$= \left(\frac{\alpha}{2\pi}\right)^{M/2} \exp\left\{-\frac{\alpha}{2} \mathbf{w}^T \mathbf{w}\right\}, \quad (3.1.17)$$

where M is the dimension of \mathbf{w} . Recall, from equation (3.1.2) that the posterior distribution is proportional to the likelihood multiplied by the prior. Also, notice that taking the natural logarithm of the posterior does not change the maximization with respect to \mathbf{w} , only the trick simplifies the derivations. Finally, it is also convenient and completely equivalent to minimize the negative logarithm of the posterior instead of maximizing the logarithm of the posterior directly. Hence, the problem boils down to minimize the negative-log-posterior proportional to

$$-\ln p(\mathbf{w}|\mathbf{X}, \mathbf{y}, \alpha, \beta) \propto -\ln p(\mathbf{y}|\mathbf{X}, \mathbf{w}, \beta) - \ln p(\mathbf{w}|\alpha) \quad (3.1.18)$$

$$= -\frac{N}{2} \ln \beta + \frac{N}{2} \ln 2\pi + \frac{\beta}{2} \sum_{n=1}^N [f(\mathbf{x}_n, \mathbf{w}) - y_n]^2 \quad (3.1.19)$$

$$- \frac{M}{2} \ln \alpha + \frac{M}{2} \ln 2\pi + \frac{\alpha}{2} \mathbf{w}^T \mathbf{w}. \quad (3.1.20)$$

Including only the terms that depend on \mathbf{w} the MAP estimate corresponds to minimize

$$\frac{\beta}{2} \sum_{n=1}^N [f(\mathbf{x}_n, \mathbf{w}) - y_n]^2 + \frac{\alpha}{2} \mathbf{w}^T \mathbf{w}, \quad (3.1.21)$$

which is equivalent to minimize the regularized sum of squared errors function given by

$$\frac{1}{2} \sum_{n=1}^N [f(\mathbf{x}_n, \mathbf{w}) - y_n]^2 + \frac{\lambda}{2} \mathbf{w}^T \mathbf{w}, \quad (3.1.22)$$

with regularization parameter $\lambda = \alpha/\beta$ (Bishop, 2006, section 1.2.5).

In case of a non-informative uniform prior on w the MAP estimate is identical to the non-Bayesian probabilistic estimate called *maximum likelihood* (ML), which only maximizes the likelihood with respect to \mathbf{w} or equivalently minimizes the negative-log-likelihood. Further details about ML solutions can be found in for instance Bishop (2006).

Because the MAP estimate is a point estimate making use of only the highest mode of the posterior distribution, it does not necessarily reflect the true behavior of the data. Therefore, a more thorough use of the Bayesian inference is also possible. This approach is referred to as a *full Bayesian approach*. In the full Bayesian approach a new prediction f_* for a future input \mathbf{x}_* requires an integration over \mathbf{w} , and in general such marginalization is the corner stone of classical Bayesian methods. The integration with respect to \mathbf{w} results in a predictive distribution $p(f_*|\mathbf{x}_*, \mathbf{X}, \mathbf{y}, \alpha, \beta)$ given by

$$p(f_*|\mathbf{x}_*, \mathbf{X}, \mathbf{y}, \alpha, \beta) = \int_{\mathbf{w}} p(f_*|\mathbf{x}_*, \mathbf{w}, \beta)p(\mathbf{w}|\mathbf{X}, \mathbf{y}, \alpha, \beta)d\mathbf{w}. \quad (3.1.23)$$

One approach based on a full Bayesian treatment is the *Gaussian process*, which will be described in the regression case in section 3.2.

3.2 Gaussian Processes for Regression

A Gaussian process (GP) is a full Bayesian approach for which a prediction f_* for a future input \mathbf{x}_* is sampled from a Gaussian distribution conditioned on the observed (training) data (\mathbf{X}, \mathbf{y})

$$p(f_*|\mathbf{x}_*, \mathbf{X}, \mathbf{y}) \sim \mathcal{N}(m, K), \quad (3.2.1)$$

where $\mathbf{X} = \{\mathbf{x}_i|i = 1..n\}$ is the matrix containing the input vector of dimension D for the n observations and $\mathbf{y} = \{y_i|i = 1..n\}$ are the noisy function targets for the n observations. m and K denote the mean and covariance of the predictive distribution, which will be functions of the observed data (\mathbf{X}, \mathbf{y}) and the new input vector \mathbf{x}_* .

This section explains the fundamentals of Gaussian Processes in a weight space view and in a functions space view in section 3.2.1 and section 3.2.2, respectively. Next, in section 3.2.3 an appropriate method to include non-zero-mean functions in a Gaussian process is developed. In this way, more informative priors over functions are incorporated in a Gaussian Process. Finally, training of a GP based on the marginal likelihood is shown in section 3.2.4.

3.2.1 Weight Space View

An intuitive procedure to derive the equations describing a GP is to begin with the standard linear regression model for which the function output is a linear combination of the inputs

$$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{w}. \quad (3.2.2)$$

In the case of additive noise ϵ on the observed function values the targets y are given by

$$y = f(\mathbf{x}) + \epsilon. \quad (3.2.3)$$

In the case of a GP it is assumed that the additive noise follows an independent Gaussian distribution with zero-mean and variance σ_n^2

$$\epsilon \sim \mathcal{N}(0, \sigma_n^2). \quad (3.2.4)$$

In a traditional Bayesian viewpoint the likelihood term is defined as the conditional probability of the observed data \mathcal{D} given the parameters, recall equation (3.1.2). Here, the likelihood term will alternatively be conditioned on the inputs \mathbf{X} . Because of the independent noise assumption, the likelihood $p(\mathbf{y}|\mathbf{X}, \mathbf{w})$ is given by factorizing over each observation in the observed data

$$\begin{aligned} p(\mathbf{y}|\mathbf{X}, \mathbf{w}) &= \prod_{i=1}^n p(y_i|\mathbf{x}_i, \mathbf{w}) = \prod_{i=1}^n \frac{1}{\sqrt{2\pi}\sigma_n} \exp\left(-\frac{(y_i - \mathbf{x}_i^\top \mathbf{w})^2}{2\sigma_n^2}\right) \\ &= \frac{1}{(2\pi\sigma_n^2)^{n/2}} \exp\left(-\frac{1}{2\sigma_n^2} \|\mathbf{y} - \mathbf{X}^\top \mathbf{w}\|^2\right) \\ &= \mathcal{N}(\mathbf{X}^\top \mathbf{w}, \sigma_n^2 \mathbf{I}). \end{aligned} \quad (3.2.5)$$

Now, placing a zero-mean Gaussian prior $p(\mathbf{w})$ with covariance matrix Σ_p on the weights \mathbf{w}

$$p(\mathbf{w}) = \mathcal{N}(\mathbf{0}, \Sigma_p), \quad (3.2.6)$$

yields

$$p(\mathbf{w}|\mathbf{X}, \mathbf{y}) \propto \exp\left[-\frac{1}{2\sigma_n^2} (\mathbf{y} - \mathbf{X}^\top \mathbf{w})^\top (\mathbf{y} - \mathbf{X}^\top \mathbf{w}) - \frac{1}{2} \mathbf{w}^\top \Sigma_p^{-1} \mathbf{w}\right]. \quad (3.2.7)$$

By “completing the square”, the posterior becomes proportional to (Rasmussen and Williams, 2006, equation 2.7)

$$p(\mathbf{w}|\mathbf{X}, \mathbf{y}) \propto \exp\left(-\frac{1}{2} (\mathbf{w} - \bar{\mathbf{w}})^\top \left(\frac{1}{\sigma_n^2} \mathbf{X} \mathbf{X}^\top + \Sigma_p^{-1}\right) (\mathbf{w} - \bar{\mathbf{w}})\right), \quad (3.2.8)$$

where $\bar{\mathbf{w}} = \sigma_n^{-2} \left(\sigma_n^{-2} \mathbf{X} \mathbf{X}^\top + \Sigma_p^{-1} \right)^{-1} \mathbf{X} \mathbf{y}$. This expression has the form of a Gaussian distribution given by

$$p(\mathbf{w} | \mathbf{X}, \mathbf{y}) \sim \mathcal{N} \left(\bar{\mathbf{w}} = \frac{1}{\sigma_n^2} \mathbf{A}^{-1} \mathbf{X} \mathbf{y}, \mathbf{A}^{-1} \right), \quad (3.2.9)$$

where $\mathbf{A} = \sigma_n^{-1} \mathbf{X} \mathbf{X}^\top + \Sigma_p^{-1}$. The predictive distribution $p(f_* | \mathbf{x}_*, \mathbf{X}, \mathbf{y})$ resulting from the full bayesian treatment (recall equation (3.1.23)) is given by (Rasmussen and Williams, 2006, equation 2.9)

$$p(f_* | \mathbf{x}_*, \mathbf{X}, \mathbf{y}) = \int_W p(f_* | \mathbf{x}_*, \mathbf{w}) p(\mathbf{w} | \mathbf{X}, \mathbf{y}) d\mathbf{w} \quad (3.2.10)$$

$$= \mathcal{N} \left(\frac{1}{\sigma_n^2} \mathbf{x}_*^\top \mathbf{A}^{-1} \mathbf{X} \mathbf{y}, \mathbf{x}_*^\top \mathbf{A}^{-1} \mathbf{x}_* \right). \quad (3.2.11)$$

The linear model described in the previous will fail to model non-linear generated data, hence to describe such data more complex models need to be included. This is done by introducing a non-linear projecting of the input data onto a possible higher dimensional feature space. The feature space mapping is described by the function $\phi(\mathbf{x})$ mapping from the D -dimensional input space to a N -dimensional feature space. The idea of the feature mapping is that although the data has a non-linear behavior in input space it might be linear in the feature space. Hence, by projecting the data to a high dimensional feature space the linear model can be applied there instead, resulting in similar results as before for the GP. The linear model applied in feature space will be given by

$$f(\mathbf{x}) = \phi(\mathbf{x})^\top \mathbf{w}. \quad (3.2.12)$$

The notation $\Phi(\mathbf{X})$ will be used to denote the N by n dimensional data matrix where the i 'th column contains the i 'th input data point projected to feature space $\phi(\mathbf{x}_i)$. Substituting \mathbf{X} with $\Phi(\mathbf{X})$ in the expression for the predictive distribution for the linear model and rewriting, results in the following expression for the predictive distribution for the non-linear model (Rasmussen and Williams, 2006, equation 2.12)

$$p(f_* | \mathbf{x}_*, \mathbf{X}, \mathbf{y}) = \mathcal{N} \left(\phi_*^\top \Sigma_p \Phi (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{y}, \phi_*^\top \Sigma_p \phi_* - \phi_*^\top \Sigma_p \Phi (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \Phi^\top \Sigma_p \phi_* \right), \quad (3.2.13)$$

where the shorthands $\Phi = \Phi(\mathbf{X})$ and $\phi_* = \phi(\mathbf{x}_*)$ have been used.

Observe, that in equation (3.2.13) the feature space mapping always enters in the form $\phi_*^\top \Sigma_p \Phi$ or $\phi_*^\top \Sigma_p \phi_*$ which are inner products. This enables the use of the *kernel trick* where instead of computing the feature mapping for all input vectors a *kernel* or *covariance function* $k(x, x') = \phi(\mathbf{x})^\top \Sigma_p \phi(\mathbf{x}')$ being an inner

product is computed instead. This trick has the advantage that everything is described in terms of scalar products between data points given by the covariance function $k(\cdot, \cdot)$. In section 3.2.2 the GP will be derived using this trick.

3.2.2 Function Space View

In this section an equivalent kernel representation of a GP is described. The kernel representation is the current state-of-the-art formulation of a GP.

A GP is defined as a collection of random variables and in this case the random variables are function values $f(\mathbf{x})$ at location \mathbf{x} . With the linear feature space model (equation (3.2.12)) and a Gaussian prior on the weights $p(\mathbf{w}) = \mathcal{N}(\mathbf{0}, \Sigma_p)$ the mean and covariance of the GP prior is given by

$$\mathbb{E}[f(\mathbf{x})] = \phi(\mathbf{x})^\top \mathbb{E}[\mathbf{w}] = 0, \quad (3.2.14)$$

$$\mathbb{E}[f(\mathbf{x})f(\mathbf{x}')] = \phi(\mathbf{x})^\top \mathbb{E}[\mathbf{w}\mathbf{w}^\top] \phi(\mathbf{x}') = \phi(\mathbf{x})^\top \Sigma_p \phi(\mathbf{x}') \quad (3.2.15)$$

Thus, the distributions of $f(\mathbf{x})$ and $f(\mathbf{x}')$ are jointly Gaussian with zero-mean and covariance given by $\phi(\mathbf{x})^\top \Sigma_p \phi(\mathbf{x}')$. There exists different possibilities for the covariance function or kernel function. One common choice is to use the *squared exponential* (SE) covariance function defined by

$$\text{cov}(f(\mathbf{x}), f(\mathbf{x}')) = k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left[-\frac{1}{2l^2} (\mathbf{x} - \mathbf{x}')^2\right], \quad (3.2.16)$$

where l and σ_f^2 is referred to as the *length scale* and the *signal variance*, respectively. The distribution of function values \mathbf{f}_* at points collected in \mathbf{X}_* drawn from the prior will now be given by

$$p(\mathbf{f}_* | \mathbf{X}_*) = \mathcal{N}(\mathbf{0}, K(\mathbf{X}_*, \mathbf{X}_*)) \quad (3.2.17)$$

Naturally, it will normally not be very interesting to sample function values from the prior. Instead, it is possible to write the joint Gaussian distribution between noise free observations (\mathbf{f}, \mathbf{X}) and test points $(\mathbf{f}^*, \mathbf{X}^*)$ as these are sampled from the same distribution.

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} K(\mathbf{X}, \mathbf{X}) & K(\mathbf{X}, \mathbf{X}_*) \\ K(\mathbf{X}_*, \mathbf{X}) & K(\mathbf{X}_*, \mathbf{X}_*) \end{bmatrix}\right) \quad (3.2.18)$$

Fortunately, there exist simple relations between the joint distribution of two Gaussian random variables and the conditional and marginal distributions of the two random variables (Rasmussen and Williams, 2006, appendix A.2), hence the

predictive distribution of the test cases conditioned on the observations can be written as

$$p(\mathbf{f}_* | \mathbf{X}_*, \mathbf{X}, \mathbf{f}) = \mathcal{N}(\bar{\mathbf{f}}_*, \text{cov}(\mathbf{f}_*)), \quad (3.2.19)$$

where the mean $\bar{\mathbf{f}}_*$ and covariance $\text{cov}(\mathbf{f}_*)$ is given by

$$\bar{\mathbf{f}}_* = K(\mathbf{X}_*, \mathbf{X}) K(\mathbf{X}, \mathbf{X})^{-1} \mathbf{f} \quad (3.2.20)$$

$$\text{cov}(\mathbf{f}_*) = K(\mathbf{X}_*, \mathbf{X}_*) - K(\mathbf{X}_*, \mathbf{X}) K(\mathbf{X}, \mathbf{X})^{-1} K(\mathbf{X}, \mathbf{X}_*) \quad (3.2.21)$$

In practice, there is noise on the observations y , hence $y = f(x) + \epsilon$. Again, assuming that the observation noise is independent Gaussian noise with zero-mean and variance σ_n^2 , the covariance function for the observations \mathbf{y} becomes

$$\text{cov}(\mathbf{y}) = K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}, \quad (3.2.22)$$

which yields

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I} & K(\mathbf{X}, \mathbf{X}_*) \\ K(\mathbf{X}_*, \mathbf{X}) & K(\mathbf{X}_*, \mathbf{X}_*) \end{bmatrix}\right) \quad (3.2.23)$$

Finally, the predictive distribution for new test cases is derived similar to equation (3.2.19)

$$p(\mathbf{f}_* | \mathbf{X}_*, \mathbf{X}, \mathbf{y}) = \mathcal{N}(\bar{\mathbf{f}}_*, \text{cov}(\mathbf{f}_*)), \quad (3.2.24)$$

where

$$\bar{\mathbf{f}}_* = K(\mathbf{X}_*, \mathbf{X}) [K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}]^{-1} \mathbf{y} \quad (3.2.25)$$

$$\text{cov}(\mathbf{f}_*) = K(\mathbf{X}_*, \mathbf{X}_*) - K(\mathbf{X}_*, \mathbf{X}) [K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}]^{-1} K(\mathbf{X}, \mathbf{X}_*) \quad (3.2.26)$$

Notice, that this result is identical to equation (3.2.13) when $K(C, D) = \Phi(C)^\top \Sigma_p \Phi(D)$, where C and D are either \mathbf{X} or \mathbf{X}_* . Also, for a particular feature mapping the equivalent kernel can be computed as $k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^\top \Sigma_p \phi(\mathbf{x}')$, but normally some standard kernel or covariance functions will be used as for instance the squared exponential kernel shown previously. For a particular kernel there exists a possible infinite expansion in terms of basis functions, hence it should (at least in theory) be possible to transform back and forth between the weight space representation and the function space representation.

Another important thing to notice is that the mean function from equation (3.2.25) is a linear predictor of the underlying function and this function has the same representation as traditional *kernel machines* defined by

$$\bar{f}(\mathbf{x}_*) = \sum_{i=1}^n \alpha_i k(\mathbf{x}_i, \mathbf{x}_*), \quad (3.2.27)$$

where \mathbf{x}_i is the i 'th observation point and for the GP linear predictor it is seen that $\alpha = (K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I})^{-1} \mathbf{y}$. Hence, the prediction $\bar{f}(\mathbf{x}_*)$ for a new test case \mathbf{x}_* is written as a linear combination of n kernel functions located at each of the observation points \mathbf{x}_i .

3.2.3 Incorporating Non-Zero-Mean Functions

In the previous formulation of a GP it has been assumed that the observation and the test point share a zero-mean joint Gaussian distribution. Generally, this does not need to be the case and in this section a method developed in this thesis to incorporate non-zero-mean functions is presented.

If an explicit mean function $\mathbf{m}(\mathbf{x})$ is specified, the predictive mean from equation (3.2.25) simple becomes

$$\bar{\mathbf{f}}_* = \mathbf{m}(\mathbf{X}_*) + K(\mathbf{X}_*, \mathbf{X}) [K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}]^{-1} \mathbf{y}, \quad (3.2.28)$$

and the predictive variance from equation (3.2.26) will be left unchanged (Rasmussen and Williams, 2006, Section 2.7). Rasmussen and Williams (2006) derives a method to incorporated a mean function in a GP in terms of a set of fixed basis functions with coefficients learned from data (Rasmussen and Williams, 2006, Equation 2.39 - 2.42).

For the outline of this thesis it will be more desirable to include what will be referred to as the *initial preference function* $h(\mathbf{x})$, containing a mean function $m(\mathbf{x})$ and a variance $V(\mathbf{x})$ over function values at a particular point \mathbf{x} . It is assumed that the distribution of function values $h(\mathbf{x})$ at two points \mathbf{x} and \mathbf{x}' are independent, hence there is no covariance between two points \mathbf{x} and \mathbf{x}' - only a variance. The distribution of $\mathbf{h}(\mathbf{x})$ is now given by

$$\mathbf{h} \sim \mathcal{N}(\mathbf{m}(\mathbf{X}), \mathbf{V}(\mathbf{X})\mathbf{I}) \quad (3.2.29)$$

In this thesis it is proposed to model this function by a traditional zero-mean GP determined beforehand as the average preference function for a group of subjects. At this point, it is important to understand that the function mean $m(\mathbf{x})$ of $h(\mathbf{x})$ serves as an initial guess of a personal preference function $g(\mathbf{x})$ including the uncertainty $V(\mathbf{x})$. The resulting preference function is given by $g(\mathbf{x}) = h(\mathbf{x}) + f(\mathbf{x})$, where the residual $f(\mathbf{x})$ is modeled by a zero-mean GP with covariance function $K(\mathbf{X}, \mathbf{X})$.

$$\mathbf{f} \sim \mathcal{N}(\mathbf{0}, K(\mathbf{X}, \mathbf{X})) \quad (3.2.30)$$

and

$$y(\mathbf{x}) = h(\mathbf{x}) + f(\mathbf{x}) + \epsilon, \quad (3.2.31)$$

where $y(\mathbf{x})$ is the observation of the function $g(\mathbf{x})$ and $\epsilon \sim \mathcal{N}(0, \sigma_n^2)$ is the contaminating noise. It is now possible to express the Gaussian distribution for both the observations $\mathbf{y}(\mathbf{X})$ and the test case $\mathbf{g}(\mathbf{X}_*)$ at \mathbf{X}_*

$$\mathbf{y} \sim \mathcal{N}(\mathbf{m}(\mathbf{X}), K(\mathbf{X}, \mathbf{X}) + \mathbf{V}(\mathbf{X})\mathbf{I} + \sigma_n^2\mathbf{I}) \quad (3.2.32)$$

$$\mathbf{g}_* \sim \mathcal{N}(\mathbf{m}(\mathbf{X}_*), K(\mathbf{X}_*, \mathbf{X}_*) + \mathbf{V}(\mathbf{X}_*)\mathbf{I}) \quad (3.2.33)$$

As in section 3.2.2 the two distributions share a joint distribution given by

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{g}_* \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} \mathbf{m}(\mathbf{X}) \\ \mathbf{m}(\mathbf{X}_*) \end{bmatrix}, \begin{bmatrix} K(\mathbf{X}, \mathbf{X}) + \mathbf{V}(\mathbf{X})\mathbf{I} + \sigma_n^2\mathbf{I} & K(\mathbf{X}, \mathbf{X}_*) \\ K(\mathbf{X}_*, \mathbf{X}) & K(\mathbf{X}_*, \mathbf{X}_*) + \mathbf{V}(\mathbf{X}_*)\mathbf{I} \end{bmatrix} \right) \quad (3.2.34)$$

Again, it is possible to use the identity for Gaussian distributions (Rasmussen and Williams, 2006, appendix A.2) and derive the predictive distribution for \mathbf{g}_* conditioned on the observations

$$p(\mathbf{g}_* | \mathbf{X}_*, \mathbf{X}, \mathbf{y}) = \mathcal{N}(\bar{\mathbf{g}}_*, \text{cov}(\mathbf{g}_*)), \quad (3.2.35)$$

where

$$\begin{aligned} \bar{\mathbf{g}}_* &= \mathbf{m}(\mathbf{X}_*) \\ &+ K(\mathbf{X}_*, \mathbf{X}) [K(\mathbf{X}, \mathbf{X}) + \mathbf{V}(\mathbf{X})\mathbf{I} + \sigma_n^2\mathbf{I}]^{-1} (\mathbf{y} + \mathbf{m}(\mathbf{X})) \end{aligned} \quad (3.2.36)$$

and

$$\begin{aligned} \text{cov}(\mathbf{g}_*) &= K(\mathbf{X}_*, \mathbf{X}_*) + \mathbf{V}(\mathbf{X}_*)\mathbf{I} \\ &- K(\mathbf{X}_*, \mathbf{X}) [K(\mathbf{X}, \mathbf{X}) + \mathbf{V}(\mathbf{X})\mathbf{I} + \sigma_n^2\mathbf{I}]^{-1} K(\mathbf{X}, \mathbf{X}_*). \end{aligned} \quad (3.2.37)$$

3.2.4 Learning the Hyper-Parameters

A GP has a weight space interpretation with a parameterized model (ref. section 3.2.1), but a GP will in general be formulated as a non-parameterized kernel machine in function space (ref. section 3.2.2). Although, the function space interpretation generally decreases the number of free parameters compared with the weight space interpretation, the parameters in the covariance function called hyper-parameters must be learned from data. This step is referred to as *training the GP*. This section describes the underlying theory for training a GP based on the log of the marginal likelihood (Rasmussen and Williams, 2006, section 4.5.1).

Recall, that the marginal likelihood is the integral over the likelihood multiplied with the prior and is a normalization constant ensuring that the posterior integrates to one and thereby becomes a valid distribution. For a GP model the prior is a Gaussian over functions $\mathbf{f}|\mathbf{X} \sim \mathcal{N}(\mathbf{0}, K(\mathbf{X}, \mathbf{X}))$ and the likelihood is a factorized Gaussian over targets $\mathbf{y}|\mathbf{f} \sim \mathcal{N}(\mathbf{f}, \sigma_n^2 \mathbf{I})$ given by

$$p(\mathbf{y}|\mathbf{f}) = \frac{1}{(2\pi)^{-n/2}} \cdot \frac{1}{\sqrt{\sigma_n^2}} \exp \left[\frac{1}{2\sigma_n^2} (\mathbf{y} - \mathbf{f})^\top (\mathbf{y} - \mathbf{f}) \right], \quad (3.2.38)$$

where n is the number of observations. This can also be written as a distribution over $\mathbf{f} \sim \mathcal{N}(\mathbf{y}, \sigma_n^2 \mathbf{I})$. Thus, the likelihood multiplied with the prior can be written as the product of two Gaussians in \mathbf{f} . Using the identities for the product of two Gaussians (Rasmussen and Williams, 2006, equation A.7 and A.8) the marginal likelihood $p(\mathbf{y}|\mathbf{X})$ is given by the normalization constant

$$p(\mathbf{y}|\mathbf{X}) = (2\pi)^{-n/2} |K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}|^{-1/2} \exp \left[-\frac{1}{2} \mathbf{y}^\top (K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I})^{-1} \mathbf{y} \right]. \quad (3.2.39)$$

Normally, the log marginal likelihood is given instead

$$\begin{aligned} \log p(\mathbf{y}|\mathbf{X}) &= -\frac{n}{2} \log 2\pi - \frac{1}{2} \log |K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}| \\ &\quad - \frac{1}{2} \mathbf{y}^\top (K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I})^{-1} \mathbf{y}. \end{aligned} \quad (3.2.40)$$

To learn the hyper-parameters the log marginal likelihood should be maximized. This training method is referred to as a *maximum marginal likelihood estimate* of the hyper-parameters. It is in principle not trivial to find the global maximum of the marginal likelihood and normally the maximization can easily end up in a local maximum. Further details about how to actually do the maximization will not be given here, but more details can be found in (Rasmussen and Williams, 2006, chapter 5).

The log marginal likelihood from equation (3.2.40) consists of three terms, each having individual roles. The first term is a normalization constant, the second term is a complexity penalty term (regularization term), which only depends on the covariance function, and the last term is the actual data fit containing the observation points. Thereby, the marginal likelihood embeds regularization of the model complexity and therefore the optimal hyper-parameter set is a natural trade-off between fitting the actual data, while keeping the model complexity in a reasonable shape.

CHAPTER 4

Active Learning Theory

In machine learning it is normally assumed that observations (input-output pairs) are available beforehand, hence the problem is to find the model that gives the best performance considering all the available data. However, for some physical problems it might be expensive to measure or test the output for particular inputs, because new experiments are time consuming, unpleasant, costly etc. In such situations it is absolutely necessary only to acquire a new observation if it is believed that the resulting observation gives significant information about the unknown function. The information is normally expressed in terms of a particular cost/goal function, hence active learning or active data selection refers to the concept of performing experiments that optimize a cost or, equivalently, goal function. Active data selection is often used iteratively to suggest one experiment at a time, but it can also be applied for a bunch of experiment or a “route” of experiments resulting in the largest reward from a cost function (Boutillier, 2002). The latter is typically referred to as experimental planning.

4.1 Maximize Total Information Gain

The first rather simply yet very intuitive strategy for active data selection is to maximize what is referred to as the *total information gain* (Mackay, 1992, Section 4.3). That is, to select a new observation in order to gain as much information about the predictor as possible, i.e. to reduce the uncertainty of the posterior the most. Mackay (1992) expresses the total information gain as the expected *change in entropy* $E[\Delta S] = E[S_N - S_{N+1}]$ with respect to the the data set between the distributions over the model parameters with and without a particular observation, where the entropy S_N reduces to

$$S_N = \int p_N(\mathbf{w}) \log \frac{1}{p_N(\mathbf{w})} d\mathbf{w}, \quad (4.1.1)$$

Further, Mackay shows that this strategy results in picking the next datum at the position where the point-wise variance of the predictor is largest given the assumption that the observation noise is independent Gaussian noise. This criterion will in the reminder of this report be referred to as *ALM*.

For a Gaussian process the variance of the predictor is directly available through equation (3.2.26) or alternatively equation (3.2.37), hence it becomes extremely easy to select the new datum at the position where the variance of a particular state of the GP is largest.

4.2 Minimize Generalization Error

Another concept by Cohn described in Seo et al. (2000) aims at selecting the next datum \tilde{x} in order to minimize the error at a reference point ξ . The idea is that information at one particular point may influence the uncertainty in other points. Therefore, this concept is referred to as *minimizing generalization error*. Based on the assumption that the current model is correct the mean square error (MSE) is assumed to be dominated by the variance term. Hence, to minimize the MSE the candidate that minimizes the overall variance should be chosen as the next datum. Given a certain covariance function $k(\cdot, \cdot)$, the overall variance given a new datum \tilde{x} can be estimated from

$$K_{N+1} = \begin{bmatrix} K(\mathbf{X}, \mathbf{X}) & K(\mathbf{X}, \tilde{\mathbf{x}}) \\ K(\mathbf{X}, \tilde{\mathbf{x}})^\top & k(\tilde{\mathbf{x}}, \tilde{\mathbf{x}}) \end{bmatrix} \quad (4.2.1)$$

The change in variance $\Delta\sigma_{\hat{y}(\xi)}^2$ at the reference point ξ as a function of the candidate $\tilde{\mathbf{x}}$ is given by

$$\Delta\sigma_{\hat{y}(\xi)}^2 = \frac{\left(K(\xi, \mathbf{X})K(\mathbf{X}, \mathbf{X})^{-1}K(\mathbf{X}, \tilde{\mathbf{x}}) - K(\tilde{\mathbf{x}}, \xi)\right)^2}{K(\tilde{\mathbf{x}}, \tilde{\mathbf{x}}) - K(\mathbf{X}, \tilde{\mathbf{x}})^\top K(\mathbf{X}, \mathbf{X})^{-1}K(\mathbf{X}, \tilde{\mathbf{x}})} \quad (4.2.2)$$

This criterion will in the remainder of this project be referred to as *ALC*. Ideally, the change in variance should be averaged over an input density $p(\mathbf{x})$ or with respect to a density $q(\mathbf{x})$ expressing the importance of different regions in input space. One possible procedure would be to normalize the mean of the GP predictive distribution and use this as the density $q(\mathbf{x})$. Consequently, regions with high user preference would be weighted as more important than regions with low user preference. An obvious problem will inevitably occur if a particular state of the GP is not a true description of the latent preference function. A possible result of this would be that the active search focuses too early on regions which are believed to have high importance based on an improper description of the latent preferences function. Thereby, the active search could get stuck in a less efficient local maximum. Therefore, care should be taken about not assigning high importance to particular regions without the required information.

4.3 Optimize for Maximum Preference

The majority of machine learning problems for which active learning is applied are concerned with the problem of given the best overall prediction performance for all possible inputs given as few observations as possible. This is particularly the basis in the concept by Mackay (1992). Alternatively, it will sometimes not be too expensive to obtain information about the distribution of the input points without having obtained the corresponding function values from the expensive experiments. In such cases, active learning is concerned about given the best overall prediction performance averaged with respect to the input distribution. This is particular the idea behind the concept by Cohn (Seo et al., 2000). Notice, that common for both of these concepts is that the function models the output from an unknown system and the goal is to be able to predict the **output** from the system given a new input. This is fundamentally different from preference learning in this thesis, where the function models preference for particular settings and the goal is to predict the setting and hence the **input** for which the preference is largest. Therefore, the two active learning concepts described above are actually the right answer to the wrong question.

In this section (section 4.3) a novel active learning method developed during this project is proposed. The method is particularly suitable for the field of

preference learning with a GP and is based on the idea of query data points $\tilde{\mathbf{x}}$ that have the highest probability of obtaining higher preference than the setting with current highest preference \mathbf{x}_{max} given the current model. The criterion will be referred to as *ALP*. The function values for the two inputs $\tilde{\mathbf{x}}$ and \mathbf{x}_{max} have a joint distribution resulting from equation (3.2.25) and equation (3.2.26)

$$\mathbf{f} = \begin{bmatrix} f(\tilde{\mathbf{x}}) \\ f(\mathbf{x}_{max}) \end{bmatrix} \sim \mathcal{N}(\bar{\mathbf{f}}_{\tilde{\mathbf{x}},max}, \text{cov}(\mathbf{f}_{\tilde{\mathbf{x}},max})), \quad (4.3.1)$$

where

$$\bar{\mathbf{f}}_{\tilde{\mathbf{x}},max} = K(\mathbf{X}_{\tilde{\mathbf{x}},max}, \mathbf{X}) [K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}]^{-1} \mathbf{y}, \quad (4.3.2)$$

$$\begin{aligned} \text{cov}(\mathbf{f}_{\tilde{\mathbf{x}},max}) &= K(\mathbf{X}_{\tilde{\mathbf{x}},max}, \mathbf{X}_{\tilde{\mathbf{x}},max}) \\ &\quad - K(\mathbf{X}_{\tilde{\mathbf{x}},max}, \mathbf{X}) [K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}]^{-1} K(\mathbf{X}, \mathbf{X}_{\tilde{\mathbf{x}},max}), \end{aligned} \quad (4.3.3)$$

the matrix $\mathbf{X}_{\tilde{\mathbf{x}},max}$ contains the actual maximum point \mathbf{x}_{max} and one particular query candidate $\tilde{\mathbf{x}}$, \mathbf{X} are the observation points and \mathbf{y} are the corresponding targets in a given iteration. To calculate the probability $P(f(\tilde{\mathbf{x}}) > f(\mathbf{x}_{max}))$ (from now on referred to as *max probability*) that the query candidate $\tilde{\mathbf{x}}$ obtains larger preference than the current maximum \mathbf{x}_{max} the joint distribution from equation (4.3.1) should be integrated over the area above the line where $f(\mathbf{x}_{max}) = f(\tilde{\mathbf{x}})$ as illustrated in figure 4.1

$$P(f(\tilde{\mathbf{x}}) > f(\mathbf{x}_{max})) = \int_{A \in \{f(\tilde{\mathbf{x}}) > f(\mathbf{x}_{max})\}} \mathcal{N}(\bar{\mathbf{f}}_{\tilde{\mathbf{x}},max}, \text{cov}(\mathbf{f}_{\tilde{\mathbf{x}},max})) d\mathbf{f}. \quad (4.3.4)$$

No closed form solutions exist for solving this integration. Instead, sampling from the distribution is used to approximate $P(f(\tilde{\mathbf{x}}) > f(\mathbf{x}_{max}))$. Since the joint distribution is only two dimensional it is fairly easy to get a proper approximation. In this thesis, 10000 samples are drawn from the distribution to provide an estimate.

Obviously, the optimal experiment is the candidate that has the highest probability of having larger preference than the current maximum, i.e., highest max probability. In practice, the function value means of the GP over the entire input space are calculated at every iteration, together with the point-wise variance. Hence, to be able to carry out the calculations needed to use this concept, only the covariance between the maximum point and all the other points in input space must be calculated additionally. Together with the point-wise variance,

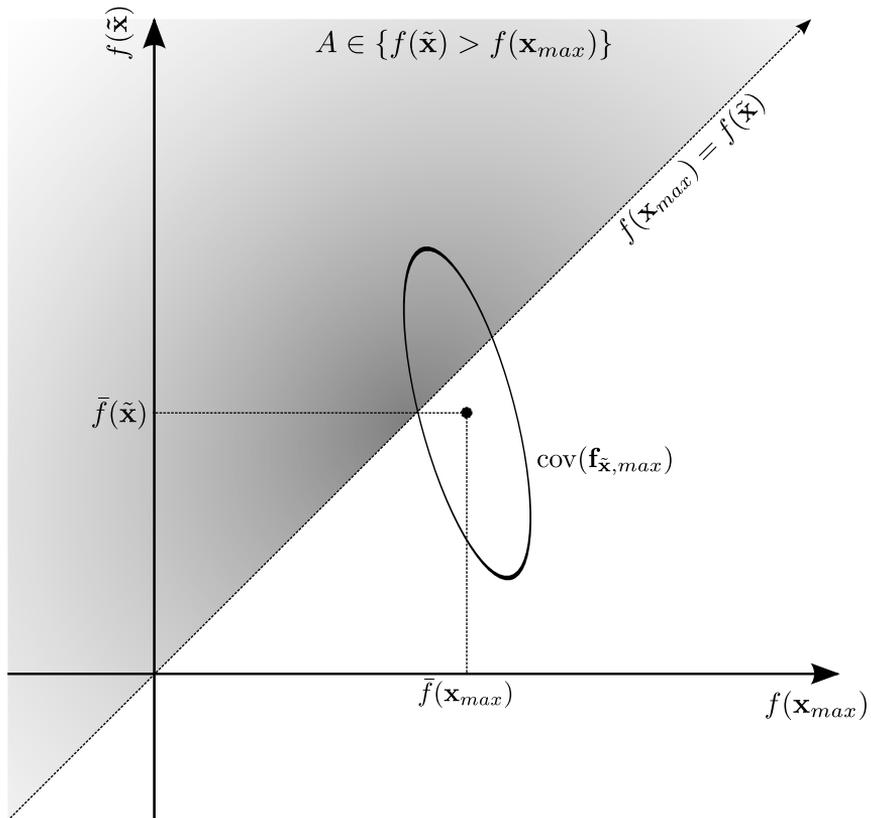


Figure 4.1: Schematic of the 2-D Gaussian distribution, with means $\bar{f}(\tilde{\mathbf{x}})$ and $\bar{f}(\mathbf{x}_{max})$ and covariance $\text{cov}(\mathbf{f}_{\tilde{\mathbf{x}},max})$. The integration area to compute $P(f(\tilde{\mathbf{x}}) > f(\mathbf{x}_{max}))$ is illustrated with the shaded area.

this forms the following matrix

$$\begin{bmatrix} \sigma_{\mathbf{x}_{max}}^2 & \text{COV}_{\mathbf{x}_{max}, \tilde{\mathbf{x}}_1} & \text{COV}_{\mathbf{x}_{max}, \tilde{\mathbf{x}}_2} & \cdots & \text{COV}_{\mathbf{x}_{max}, \tilde{\mathbf{x}}_{n-1}} \\ \text{COV}_{\tilde{\mathbf{x}}_1, \mathbf{x}_{max}} & \sigma_{\tilde{\mathbf{x}}_1}^2 & 0 & \cdots & 0 \\ \text{COV}_{\tilde{\mathbf{x}}_2, \mathbf{x}_{max}} & 0 & \sigma_{\tilde{\mathbf{x}}_2}^2 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ \text{COV}_{\tilde{\mathbf{x}}_{n-1}, \mathbf{x}_{max}} & 0 & \cdots & 0 & \sigma_{\tilde{\mathbf{x}}_{n-1}}^2 \end{bmatrix} \quad (4.3.5)$$

This matrix contains all the sub-covariance matrices for the 2 dimensional joint distributions between the maximum point and another point $\tilde{\mathbf{x}}_k$, where $k = 1, 2, 3, \dots, n - 1$. To get the 2×2 covariance matrix needed to approximate the max probability for $\tilde{\mathbf{x}}_k$ the four elements being in the first and in the $k + 1$ columns and rows should be used. The means for all the relevant 2 dimensional joint distribution are collected similarly.

As for most other maximization algorithms, the method presented here also suffers from getting stuck in a local maximum. However, since the next experiment is strictly based on the joint Gaussian distribution over function values, and thus the covariance function, changing the hyper-parameters in the covariance function can possible force the algorithm out of a local maximum. Assume, that the algorithm is stuck in a local maximum given a particular marginal likelihood estimate of the hyper-parameter in a GP with a SE kernel. Now, reducing the length scale l of the SE kernel (equation (3.2.16)) reduces the similarity between points distant from the observations, and consequently the variance in distant points increases. Alternatively, increasing the signal variance σ_f^2 in the SE kernel increases the uncertainty at points dissimilar to the observation points. Thereby, changing the hyper-parameters learned from the observations enables a possibility to take action and modify the active learning algorithm towards a search that emphasize uncertainty over high preference. In the limit where the length scale is very small, all unobserved points will be given the same chance of having higher preference than the current maximum, whereas observation points have no probability of having higher preference than the current maximum. Hence, in that limit the search is ideally random. Naturally, action should only be taken whenever it is detected that the search is stuck in a possible local maximum. The active learning algorithm presented here will be investigated further in section 4.4 by numerous simulation examples.

4.4 2D Simulation

This section investigates the active learning algorithm with the ALP criterion developed in this work by a 2D simulation study. The goal is to find the optimum

of a pre-defined underlying preference function, without any knowledge about the function.

4.4.1 Simulations method

The estimate of the underlying preference function is modeled with a zero-mean GP with a SE covariance function. When noise is added to the observations the covariance function additionally contains an independent Gaussian noise term. The hyper-parameters are learned from the observed data available at each iteration by maximizing the marginal likelihood as described in section 3.2.4. In the simulations the observations are direct elicited preference values from the pre-defined underlying preference function, i.e., an observation at a given point consists of an input and a corresponding preference value determined by the underlying preference function directly. Noisy observations are simulated by adding independent zero-mean Gaussian noise to the preference values.

The pre-defined preference function is a modified version of the so-called *Griewangk function*¹. The standard Griewangk function has depending on the sign a maximum/minimum at (0,0), whereas it contains local extrema almost identical in size as the global maximum/minimum. In these simulations the sign is always defined to obtain a global maximum in (0,0). The modifications made to the standard Griewangk function decrease the local maxima relative to the global maximum. Further, the four local maxima do not have identical function values, as seen in the top-left plot in figure 4.2. This is done in order not only to have improvement, i.e., higher preference value, between the local maxima and the global maximum, but between two different local maxima as well. The expression for the two-dimensional objective function $f(x, y)$ used as the underlying unknown preference function in the simulations is defined by

$$f(x, y) = \frac{x + y}{50} - \frac{x^2 + y^3}{200} - \frac{x^2 + y^2}{4000} + \cos(x) \cos\left(\frac{y}{\sqrt{2}}\right) - 1. \quad (4.4.1)$$

This expression will be used throughout this work to simulate an underlying preference function, because it contains several local maxima at which the algorithm can possibly not escape. Therefore, the function is suitable for testing how well the algorithm can escape a local maximum. Both dimensions x_1 and x_2 of the two-dimensional input space are restricted to a range between -5 and 5 and the resolution in each dimension is 41 uniformly spaced points. Thereby, the number of possible inputs for which the one with the highest preference value is to be discovered, is $41 \cdot 41 = 1681$.

¹The expression for the standard Griewangk function in 2 dimensions is given by $f(x, y) = \frac{x^2 + y^2}{4000} - \cos(x) \cos\left(\frac{y}{\sqrt{2}}\right) + 1$

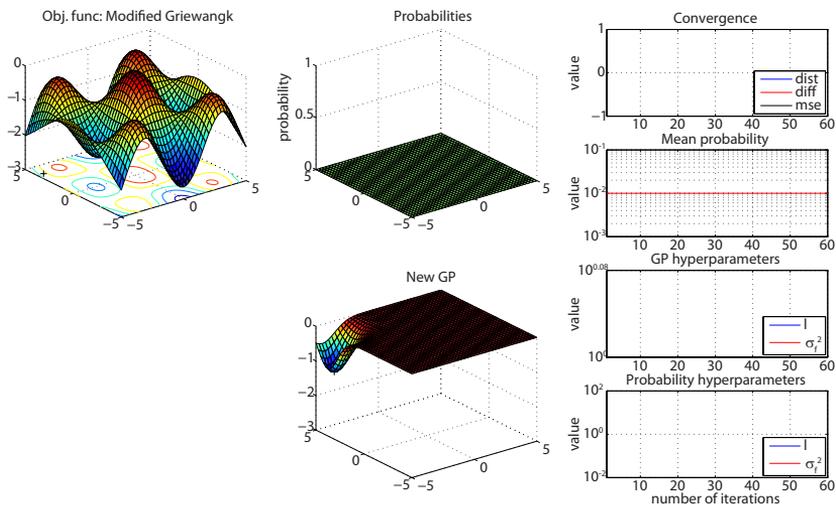


Figure 4.2: 2D simulation plot for the first iteration with a modified Griewangk function as the unknown underlying preference function. The top-left plot shows the objective preference function. In the bottom-left plot the old GP fit will be plotted resulting in the max probabilities shown in the top-middle plot. The bottom-middle plot shows the new GP fit resulting from adding the new observation suggested by the active learning algorithm to the observations. The right plots show the convergence measures (normalized distance and normalized preference value difference between current maximum point and the objective maximum point and the mean square error mse between the GP fit and the objective preference function over the entire input space), the mean of the max probabilities, the GP hyper-parameters and the hyper-parameters used to calculate the probabilities

The first observation point is always chosen randomly among all input points. Hereafter, the active learning algorithm chooses the remaining observations iteratively, i.e., based on the max probabilities calculated in each iteration.

Next, the mean of the max probabilities m_{maxthres} over all points is calculated in each iteration. If this mean drops below a certain threshold value the hyper-parameters are altered until the mean is above the threshold. When the mean is above the threshold value, the resulting max probabilities are used to suggest the next experiment. The value of m_{maxthres} , which hyper-parameters that are altered and how they are altered will be explained later in this section.

The simulation study is divided into two parts. In the first part (section 4.4.2) there is no noise on the observations of the underlying preference function values. Therefore, the active learning algorithm is not allowed to suggest the same observation twice, because this would be redundant information. Instead, if the preference for the setting with the highest max probability has already been observed, the setting with the second highest max probability is suggested and so forth until a yet unobserved new point is suggested. The ALP criterion does not define a max probability value for the current maximum point. To avoid an observation in the current maximum point a max probability value of 0 is assigned to the current maximum point in each iteration.

The threshold of the mean of the max probabilities m_{maxthres} is set to 10^{-2} . When the mean of the max probabilities is below this value the signal variance $\sigma_{f,\text{maxprob}}^2$ for the calculations of the max probabilities is increased in finite steps controlled by the integer $k_{\sigma_f^2}$ such that

$$\sigma_{f,\text{maxprob}}^2 = \sigma_f^{2(1+2 \cdot k_{\sigma_f^2})}, \quad (4.4.2)$$

where σ_f^2 is the signal variance in the GP fit learned from the observed data by marginal likelihood maximization. Hence, when $m_{\text{maxthres}} < 10^{-2}$, $k_{\sigma_f^2}$ will be set to 2. If still $m_{\text{maxthres}} < 10^{-2}$, then $k_{\sigma_f^2}$ will be set to 3 and so forth until $m_{\text{maxthres}} \geq 10^{-2}$. A summary of the simulation conditions for the noise-free simulations is shown in table 4.1

In the second part (section 4.4.3) the observations are noisy. Therefore, it is convenient to allow for several observations in the same point to reduce the uncertainty of the observations in important regions with assumed high preference. However, to prevent getting stuck in a local maximum, different strategies will apply. One possibility is to do something similar as in the noise-free case. That is to suggest the point with the second highest probability instead of suggesting the point with the highest probability, whenever the same observation point has been suggested through several iterations. Another possibility is to alter the

| | |
|--------------------------------------|--|
| Underlying preference function | Modified Griewangk |
| Input space dimensions | 2D |
| x range | -5;5 |
| y range | -5;5 |
| Resolution in each dimension | 41 uniformly spaced points |
| Number of iterations | 60 |
| Covariance function | SE |
| Covariance function hyper-parameters | l, σ_f^2 |
| m_{maxthres} | 10^{-2} |
| Altered hyper-parameter | $\sigma_{f,\text{maxprob}}^2 = \sigma_f^2(1+2 \cdot k \cdot \sigma_f^2)$ |
| Repeat observations | no |

Table 4.1: Simulation conditions for noise-free simulations

hyper-parameters when the maximum point has been the same throughout several iterations, because it can thereby be assumed that the search has converged to a possible local maximum. The simulations will not investigate all of these different possibilities nor trying to find and optimize the best strategy. Instead, the algorithm is allowed to perform infinitively many observations in the same point, i.e., no ad-hoc procedure is implemented to detect if the algorithm is caught and keeps suggesting the same setting as the optimal one. This is done to present the baseline performance and give an impression of what could be a promising strategy to pursue for modifying and optimizing the algorithm in a realistic noisy setup using real subjects.

The addition of noise to the observations obviously makes the problem more ill-posed. Consequently, the GP model itself might not be able to make a proper estimate of the underlying function as easy as in the noise-free case.

Additionally, if the maximum point of a current GP fit is a point where no observations have been performed, the max probability will in the maximum point exceptionally not be set to zero, but to 1 in this point. This is done to make sure that an observation is performed in this point in the next iteration. Without this constraint, problems will occur if the predicted maximum is very large compared with other points and at the same time has a very high uncertainty. The ALP criterion assumes that the fit around the current maximum is correct, which is important to verify. This can only be done by an observation. A

drawback from this is that additional experiments will be wasted on identifying that a predicted maximum actually has a high preference value.

Finally, the threshold of the mean of the max probabilities is increased to 10^{-1} , because the independent Gaussian noise included in the GP will always add some additional uncertainty to the fit, and thereby the mean of the max probabilities will not reduce so dramatically as in the noise-free case. Furthermore, both the signal variance σ_f^2 and the length scale l of the SE covariance function will be altered simultaneously to increase the max probabilities more dramatically. The signal variance for the max probability calculations $\sigma_{f,\text{maxprob}}^2$ is again defined by equation (4.4.2). The length scale l_{maxprob} for the max probability calculations is defined by

$$l_{\text{maxprob}} = l^{(1/2)^{k_l}}, \quad (4.4.3)$$

where l is the marginal likelihood estimated length scale used in the GP fit. $k_l = 1, 2, 3, \dots$ is a integer controlling the step size in the same manner that $k_{\sigma_f^2}$ controls the step size of the signal variance for the max probability calculations (see equation (4.4.2)). The reason for reducing the length scale in addition to the signal variance is that otherwise the algorithm has a tendency to get caught more frequently. It should be kept in mind that these simulations serve to present a baseline performance. How exactly the hyper-parameters should be altered must be determined more thoroughly in the future (ref. section 7.2). Note, that the adjustment of l_{maxprob} (equation (4.4.3)) is possibly not appropriate if the marginal likelihood estimated length scale l is smaller than 1. In that case the modified length scale l_{maxprob} will be larger than the estimated l , which is not the intention. The intention is to reduce the length scale to increase the similarity between observation and candidate points. Fortunately, the marginal likelihood estimated length scale l is never below zero in any of the simulations, hence the imperfection has not influenced the noisy simulations significantly.

The noisy simulation is performed for five different noise levels with independent zero-mean Gaussian noise as mentioned previously. The standard deviations σ_n for the five noise levels are all set to be a fraction k_σ of the standard deviation σ_{obj} of the underlying preference function (modified Griewangk), thus

$$\sigma_n = k_\sigma \cdot \sigma_{obj} \quad (4.4.4)$$

The five simulations correspond to $k_\sigma = \{0.1; 0.3; 0.5; 0.7; 0.9\}$. In table 4.2 a summary of the noisy simulation conditions is shown.

| | |
|--------------------------------------|--|
| Underlying preference function | Modified Griewangk |
| Input space dimensions | 2D |
| x range | -5;5 |
| y range | -5;5 |
| Resolution in each dimension | 41 uniformly spaced points |
| Number of iterations | 60 (visual examples) 100 (convergence comparison) |
| Covariance function | SE and independent noise |
| Covariance function hyper-parameters | l, σ_f^2, σ_n |
| $m_{\max\text{thres}}$ | 10^{-1} |
| Altered hyper-parameter | $\sigma_{f,\max\text{prob}}^2 = \sigma_f^2(1+2 \cdot k_\sigma \sigma_f^2)$ $l_{\max\text{prob}} = l(1/2)^{k_l}$ |
| k_σ | 0.1;0.3;0.5;0.7;0.9 |
| Repeat observations | yes (infinitely) |

Table 4.2: Simulation conditions for noisy simulations

4.4.2 Noise-Free Observations Results

As the objective with these simulations is to study the behavior of the active learning algorithm, it will be interesting to study the evolution in the simulations. To visualize this, the current state of the GP fit is plotted in each iteration together with the max probability for all points and the previous GP fit resulting in the probabilities. Additionally, three convergence measures are plotted; the normalized distance (*dist*) and the normalized preference value difference (*diff*) between the current maximum point and the actual maximum point of the underlying function and the mean square difference (*mse*) between the GP fit and the objective preference function over the entire input space. The three convergence measures are given by

$$dist = \frac{|\mathbf{x}_{obj,max} - \mathbf{x}_{max}|}{\sqrt{200}}, \quad (4.4.5)$$

$$diff = \frac{f_{obj,max} - f_{obj}(\mathbf{x}_{max})}{f_{obj,max} - f_{obj,min}} \quad \text{and} \quad (4.4.6)$$

$$mse = \sqrt{(\mathbf{f}_{obj} - \mathbf{f})^2}, \quad (4.4.7)$$

where $\mathbf{x}_{obj,max}$ is the maximum point of the underlying preference function, \mathbf{x}_{max} is the current maximum point of the GP fit, $f_{obj,max}$ and $f_{obj,min}$ are the maximum and minimum preference values of the underlying preference function, respectively, and $f_{obj}(\mathbf{x}_{max})$ is the preference function value of the underlying preference function at the location of the maximum point of the current GP fit.

Next, the mean of the max probabilities, the hyper-parameters of the current GP fit and the hyper-parameters for the probability calculations are plotted. In figure 4.2 this configuration is shown.

The inclusion of the first randomly chosen observation results in the GP fit shown in the bottom-left (previous GP state) plot in figure 4.3, where the data from the second iteration is depicted. Notice, that the shape of the max probabilities (top-middle plot in figure 4.3) is similar to the shape of the previous GP state for which the max probabilities are based. Furthermore, it can be seen that points far from the first observation point obtain the same max probability of 0.5. This is due to the fact that the marginal likelihood optimization of the hyper-parameters does not have enough observations and thus information to make a proper estimate of the underlying function. Therefore, the prior over functions is the “best guess” in areas distant from the observation. The large uncertainty resulting from the prior causes the max probabilities to increase in these areas as seen in the top-middle plot of figure 4.3. As a result, the next observation point will be chosen randomly from areas distant from any of the observations, whenever there are not enough observations to make a proper estimate of the

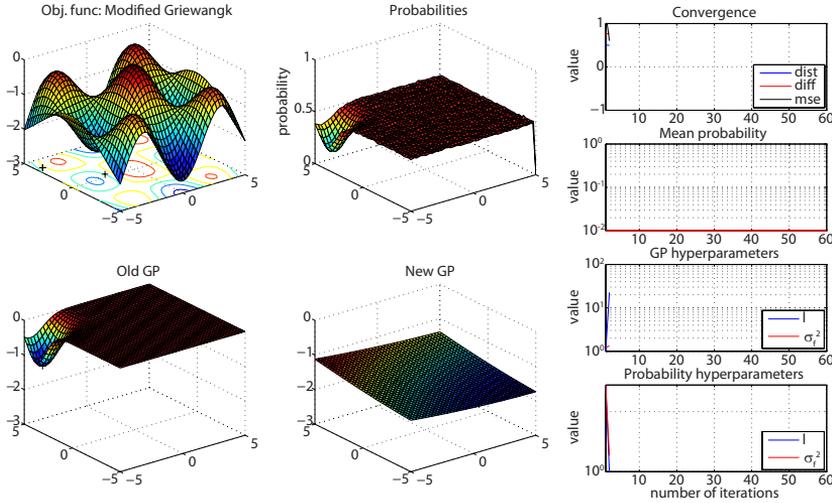


Figure 4.3: 2D simulation plot for the second iteration with a modified Griewangk function as the unknown underlying preference function.

underlying preference function. This is a convenient property of the algorithm, because the observations will ideally be evenly spread in input space until a proper estimate of the underlying preference function can be provided. The small fluctuations seen in the max probability plot are due to the sampling method approximating the integral from equation (4.3.4).

At a certain iteration the marginal likelihood estimate of the hyper-parameters fits the observations, whereby predictions are not merely drawn from the prior anymore. This is because the observations influence the predictions everywhere in input space. The data depicted in figure 4.4 shows the 5'th iteration. Here the old GP (previous GP state) is the first fit, where the marginal likelihood estimate of the hyper-parameters “smooths” the data. Obviously, the smoothed fit is different from the underlying preference function, but at this point in the search it is the best estimate the GP can provide. Notice, that the shape of the max probabilities over the input space is different from the shape of the old GP fit for which the probabilities are based. The “east” corner is far from any of the observation, hence the uncertainty is large, while the mean of the GP (predictor) is also relatively large, causing the max probabilities in this area to be large. In the “west” corner the mean of the GP is largest, thus even though the area is close to an observation point and the uncertainty thereby is small, the max probabilities are still large in this area due to the large mean values. Thereby, the active learning algorithm appears to be a natural weighting between a high predicted preference and uncertainty.

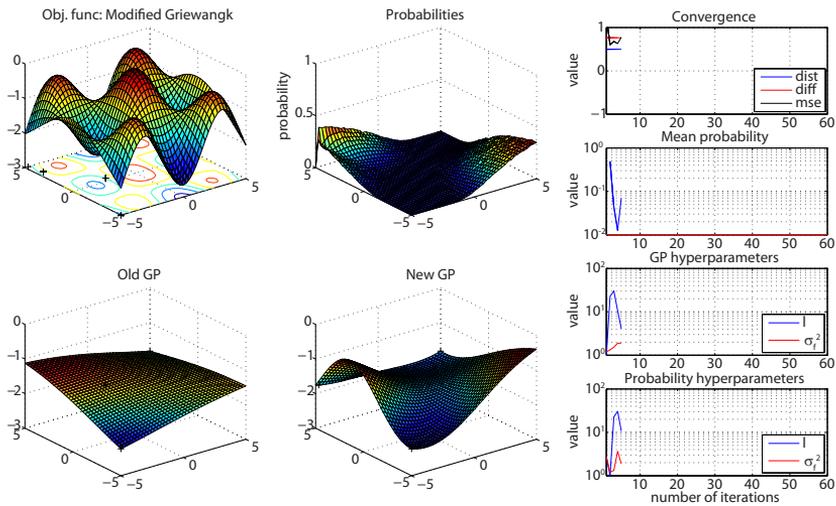


Figure 4.4: 2D simulation plot for the 5th iteration with a modified Griewangk function as the unknown underlying preference function.

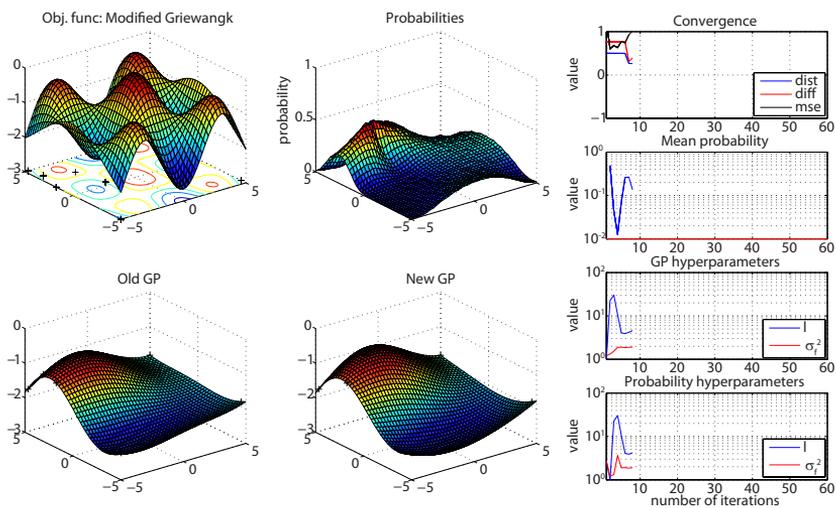


Figure 4.5: 2D simulation plot for the 8th iteration with a modified Griewangk function as the unknown underlying preference function.

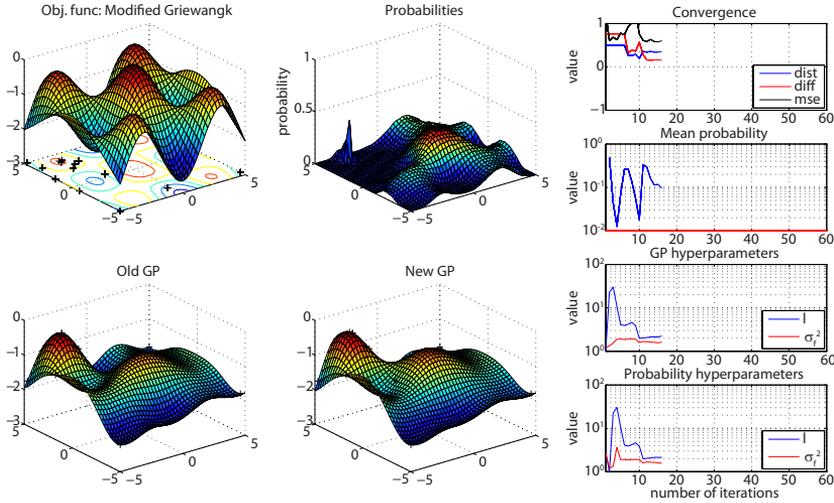


Figure 4.6: 2D simulation plot for the 16'th iteration with a modified Griewangk function as the unknown underlying preference function.

In the 8'th iteration (see figure 4.5) the search has located a possible maximum. Notice, how the shape of the max probabilities is sharpened in the area of the possible located maximum. In subsequent iterations the shape of the max probabilities becomes increasingly sharpened around a decreasingly narrow area around the possible local maximum. In this way the correct location of the local maximum is discovered. In the 16'th iteration (see figure 4.6) the local maximum is correctly identified. Notice, that the shape of the max probabilities has an imperceptible narrow peak next to the center of the local maximum. The evolution from the 8'th (figure 4.5) to the 16'th iteration (figure 4.6) shows that the algorithm quickly discovers a maximum whenever the possibility of obtaining one is identified. This is a very satisfying property within the area of preference learning, where the number of observations can be restricted such that it might not be possible always to locate the global maximum within the allowed number of observations. In these cases, it is absolutely necessary that the found optimal setting is as good as possible. Therefore, a fast located local maximum is preferable compared to less efficient settings based on an incorrect overall estimate of the underlying preference function.

The global maximum is detected in the 21'st iteration depicted in figure 4.7. At this point it is interesting to observe the locations of the observations. Besides the underlying preference function, the top-left plot also contains crosses indicating the locations of the observations. The inputs of the observations are as expected not spread equally in input space, but instead the observations are

more densely distributed in regions with high preference. Hence, the algorithm only spends further observations if a region turns out to have high preference, whereas it uses a minimum of observations to discover regions with low preferences. This behavior is exactly the desired behavior. In the 21'st iteration the mean of the max probabilities based on the marginal likelihood estimate of the hyper-parameters has become smaller than the threshold of 10^{-2} . Consequently the signal variance $\sigma_{f,\text{maxprob}}^2$ used for the max probability calculations (bottom plot in the right side of figure 4.7) is increased relative to the signal variance σ_f^2 of the GP fit (second plot from the bottom in the right side of figure 4.7) (ref. equation (4.4.2)). As mentioned previously, this causes the mean of the max probabilities to get above the threshold resulting in the max probabilities seen in the top-middle plot. In this way the uncertainty is weighted higher than the uncertainty resulting from the marginal likelihood maximization, and the active learning algorithm continues outside the current maximum.

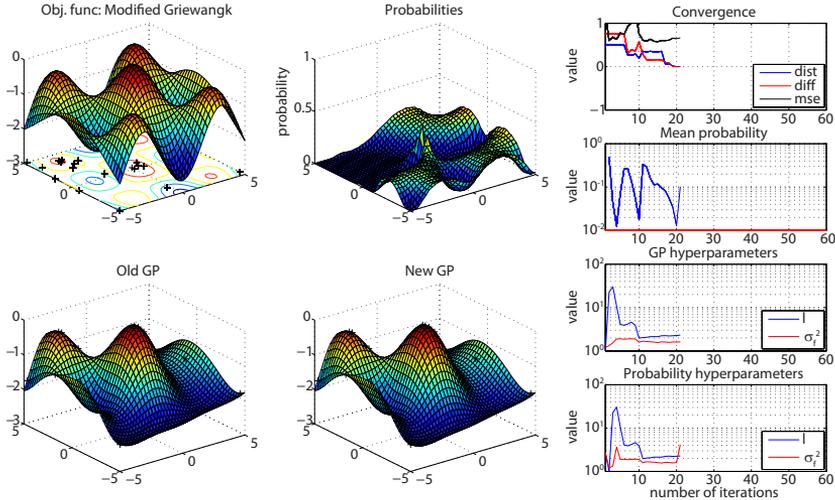


Figure 4.7: 2D simulation plot for the 21'st iteration with a modified Griewangk function as the unknown underlying preference function.

In figure 4.8 the 26'th iteration is presented showing how the signal variance for the max probability calculations remains larger than the marginal likelihood estimate of the function variance used for the GP fit to keep the mean of the max probabilities above the threshold. This forces the algorithm to search outside regions with already found high preference. Actually, the four regions with high max probabilities are seen to coincide properly with the three undiscovered local maxima, although this might only be a coincidence. Behavior like this will presumably be very dependent on the underlying unknown preference

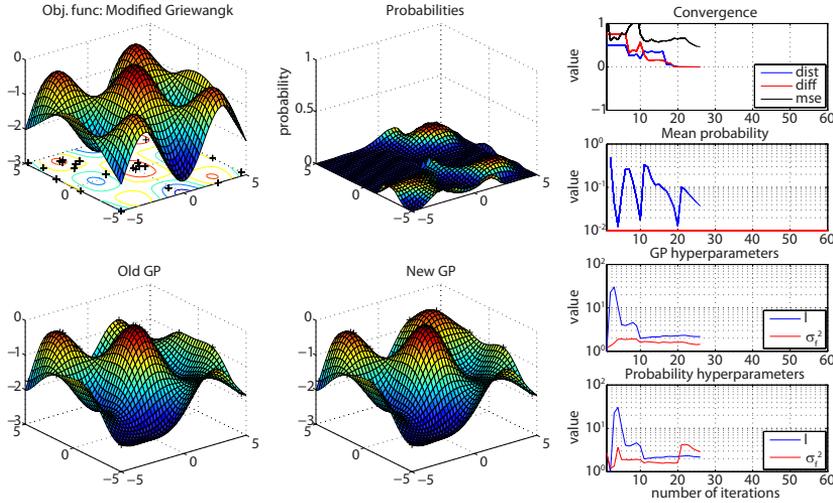


Figure 4.8: 2D simulation plot for the 26'th iteration with a modified Griewangk function as the unknown underlying preference function.

function. By looking at the mse error plotted in the top of the plots in the right side of figure 4.8, it is seen that the algorithm begins to reduce the mse error and thus the generalization error of the entire preference function. This is even more profound in the 41'st iteration depicted in figure 4.9 where the mse error has degraded from the point where the global maximum was discovered. In that iteration the underlying preference function is almost completely identified. Notice, how the signal variance for the max probability calculations has increased accordingly. Thus, it turns out that first the active learning algorithm focuses on high preference, and second by increasing the uncertainty through the hyper-parameters the algorithm puts more emphasis in reducing the generalization error and discovers the entire underlying preference function. By comparing the locations of the observations in the 41'st iteration and in the converging 26'st iteration, it can be seen that even though the observations have been spread more equally in input space after the search has converged, the locations are still distributed more densely in high preference regions than in low preference regions.

Although this particular run (figure 4.3 - figure 4.9) is actually fairly representative for the behavior of the active learning algorithm, it is important to see the average behavior over many runs. In the following, the average performance over 100 simulations is analyzed by presenting identical convergence measures as those shown in the top-right plots of the figures, i.e., *dist*, *diff* and *mse*. In order to compare the in this thesis developed active learning criterion (ALP)

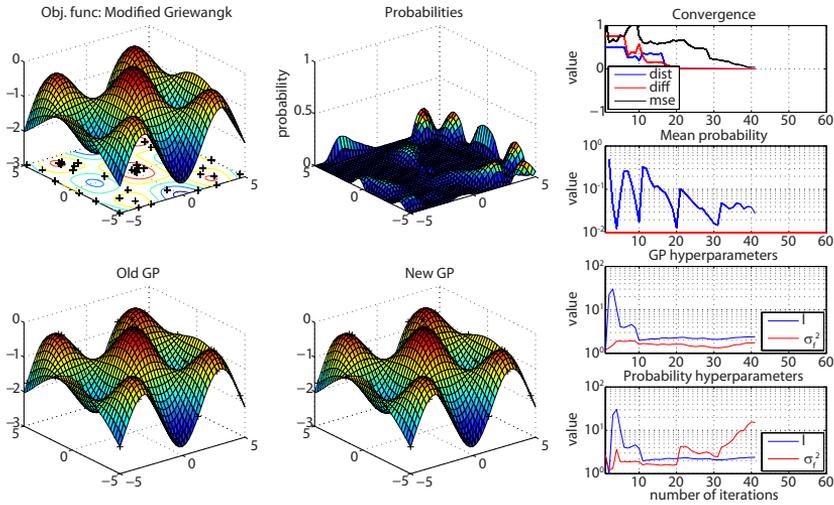


Figure 4.9: 2D simulation plot for the 41'st iteration with a modified Griewangk function as the unknown underlying preference function.

with other algorithm types, two further simulation types with different active learning criteria are conducted. Again, 100 simulations are performed for each of the criteria. The first criterion consists of choosing new observations randomly and in the second criterion new observations are chosen according to Mackay's criterion, i.e., new observations are selected where the point-wise variance is largest. In figure 4.10 the mean of the three convergence measures over the 100 simulations have been plotted for each of the three active learning criterion.

First of all, it is seen that selecting new observation points randomly is as expected not a proper strategy - the convergence time is considerably longer than both of the two other criterion (see red curves in figure 4.10). Therefore, no further attention will be given to the random strategy.

Comparing the ALP and ALM criteria, it can be seen that for this particular objective function the two methods converge on average equally fast towards the global maximum - both criteria result in nearly global convergence after 30 iterations (black and blue curves in the middle plot of figure 4.10). However, if the search is finished before the global convergence point, i.e., before the 30'th iteration, the ALP criterion results in an on average considerably better proposed best setting than the ALM criterion. On average, the ALP criterion results in a proposed best setting with a preference value that is only 10 percent lower than the preference value of the global maximum after 17 iterations. In contrast, with the ALM criterion 24 iterations are used on average to reach the

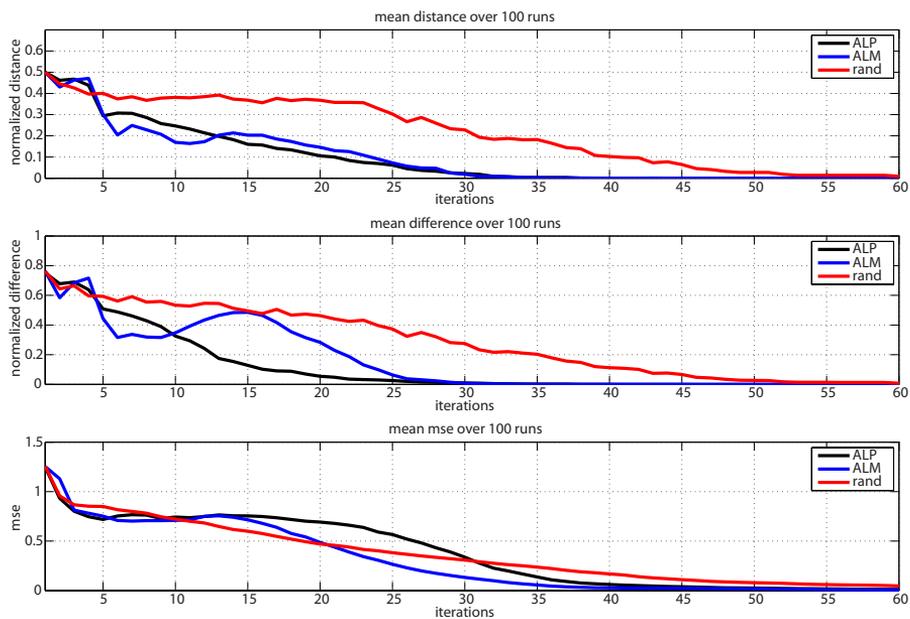


Figure 4.10: Mean over 100 runs for each of the three data selecting criteria - ALP, ALM and random. The top plot shows the normalized distance, the middle plot shows the normalized preference difference and the bottom plot shows the mse.

same preference value for the current best setting. When studying the average performance of the ALP criterion, it can once again be seen that a good solution in relation to high preference is quickly discovered compared with the two other criteria. As mentioned previously, this is exactly the desired behavior, since it might be impossible in a real setting to perform enough experiments to locate the global maximum point. The ALP method seems to have the ability to obtain a solution with a good preference value in few iterations compared with the ALM method - at least in this idealized simulation.

Another interesting observation to draw from figure 4.10 is that the two active learning methods reduce the generalization error (mse) differently. The two methods seem to converge completely in the same number of iterations, yet the ALM method reduces the mse faster than the ALP method. Hence, this indicates that the ALP method contains a natural trade-off between finding the maximum point of an underlying function and generalizing the entire function itself. The behavior seen previously, where the ALP criterion first discovers the global optimum and then afterwards generalizes the entire underlying function, is also observed here over 100 runs.

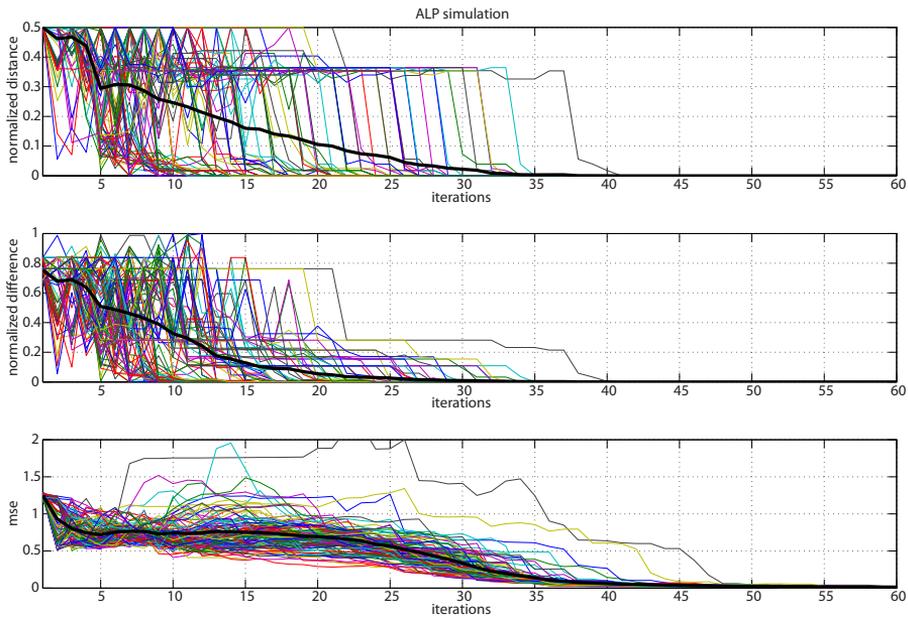


Figure 4.11: 100 similar runs with the ALP method together with the corresponding mean plotted with a thick line. The top plot shows the normalized distance, the middle plot shows the normalized preference difference and the bottom plot shows the mse.

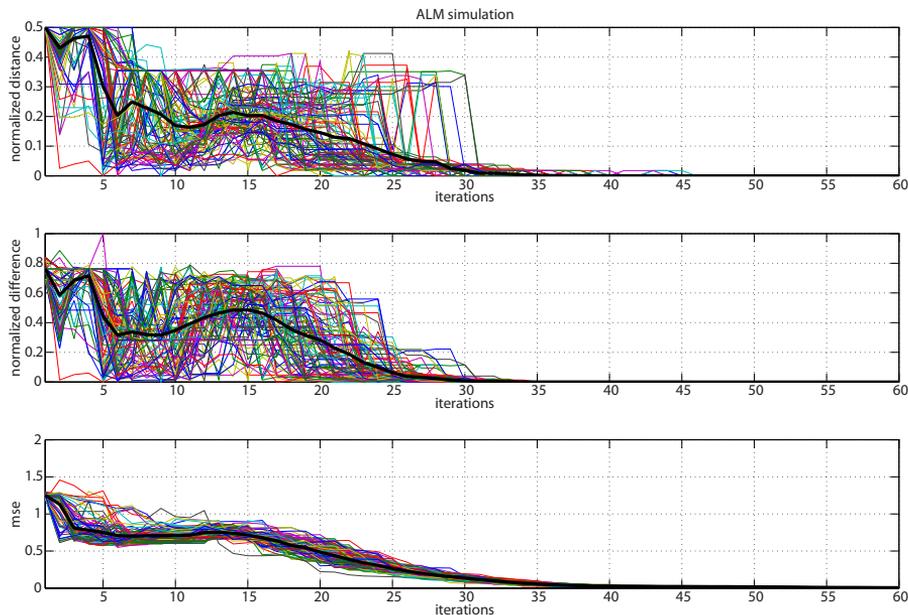


Figure 4.12: 100 similar runs with the ALM method together with the corresponding mean plotted with a thick line. The top plot shows the normalized distance, the middle plot shows the normalized preference difference and the bottom plot shows the mse.

In figure 4.11 and figure 4.12 the individual runs are shown together with corresponding means from figure 4.10, for the ALP and the ALM criteria, respectively. These two figures give an expression of the data resulting in the average performance from figure 4.10. The reason that the ALM criterion results in a proposed best setting with higher preference value than the ALP criterion early in the run (between the 5'th and the 10'th iteration) is due to the shape of the underlying preference function. The ALM method will use observations at the locations with highest variance, hence ideally for fixed hyper-parameters in the GP, this results in equally distributed observations in input space. Consequently, the GP will smooth the observations before any of the real peaks are observed. Apparently, this smoothing often gives rise to a weak maximum around the middle of input space early in the run (see the normalized distance in the top plot of figure 4.12). For the objective function used in these simulations, suggesting a point around the middle as an optimal setting results in a setting with relatively high preference. Thereby, the ALM criterion seems to perform better very early in the run compared to the ALP criterion, but this is a result of an incorrect estimate of the underlying function, which results in a good proposed optimal

setting for this particular underlying preference function. In comparison, the proposed optimal setting early in the run with the ALP criterion corresponds to one of the local maxima as seen from the normalized distance plot in the top of figure 4.11.

4.4.3 Noisy Observations Results

For each noise level 100 simulations are performed and the average performance for each noise level is plotted in figure 4.13 together with the noise-free average performance from figure 4.11.

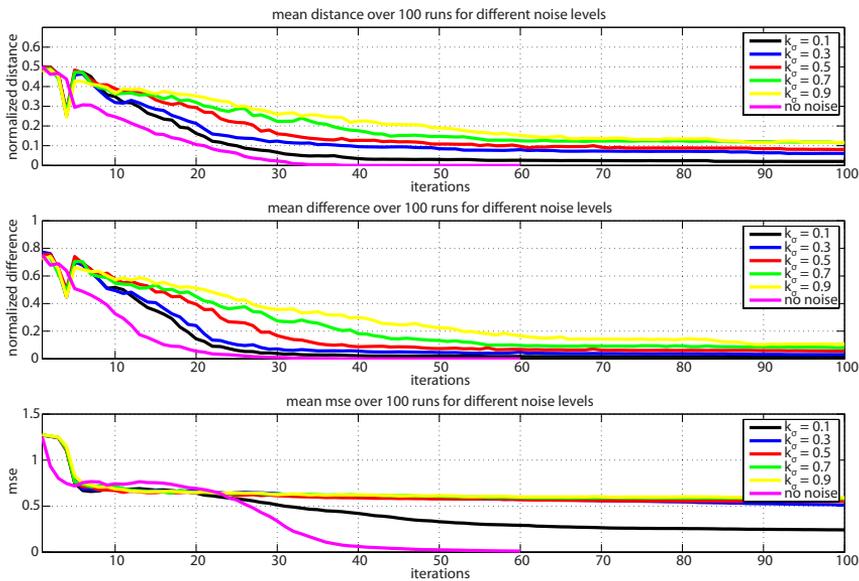


Figure 4.13: Mean over 100 runs for each of the five noise levels together with the noise-free simulation. The top plot shows the normalized distance, the middle plot shows the normalized preference difference and the bottom plot shows the mse.

As expected the convergence time increases as the noise level increases. It can also be noticed that on average the solution for which the search converges gets increasingly worse with increasing noise level. With the relatively small number of observations located in assumed high preference regions, the uncertainty caused by the observation noise will be difficult for the GP to handle. When the strength of the observation noise is comparable to the fluctuations in the

underlying preference function, the required number of repeated observation in a given points must be increased to reduce the noise. Therefore, with limited number of observations the precision of the provided preference function estimate has a limit, which decreases with increasing noise level. To avoid this behavior numerous observations should be made at a lot of input points, but this would obviously be infeasible in practice and consequently it would also be meaningless to apply active learning. Conclusively, there is a limit in how good on average the proposed optimal setting can get with very noisy observations. However, $k_\sigma = 0.9$ is an extremely high amount of noise (ref. equation (4.4.4)), and for medium noise contributions the algorithm shows acceptable average performance, although there is still room for improvements. At this point, no more effort is put into optimizing the algorithm, but instead a couple of general observations will be shown.

In figure 4.14 the 19'th iteration for one particular noisy simulation with $k_\sigma = 0.3$ is shown. In this iteration the GP starts to fit the data with something else than just a plane. Until the 19'th iteration the active learning has only picked observation points randomly. In the noise-free simulations this happened earlier (in the provided example at the 4'th iteration), because the absence of noise made the problem of over-fitting less dominating. In the noisy case, the covariance function additionally contains the independent noise variance. Apparently, the embedded regularization in the marginal likelihood optimization of the hyperparameters, restricts the GP not to over-fit the underlying functions, before the probability of the data dominates the model complexity (ref. equation (3.2.40)). This obviously increases the convergence time.

In figure 4.15 the 46'th iteration is shown. In this iteration the search has converged - not to the global maximum, but to the largest of the local maxima. For the remaining 14 iterations the algorithm keeps suggesting observations in the setting corresponding to the prominent peak observed in the max probabilities. This is because there is no limit in how many observations that can be performed at a given setting. Even though the search cannot move away from the local maximum, unexplored information about the maximum of the underlying preference function is obviously still captured in the max probabilities. The peak located next to the current proposed optimal setting prevents the search in continuing into other regions observed also to have considerably high max probabilities. In these simulations no constraints have been applied for detecting if the search has converged to a certain maximum and does not explore the input space any further. These constraints should not be specific stopping criteria, but for instance criteria that detect if either the proposed optimal setting has not moved or the new observation point has not moved throughout several iterations.

The first simple constraint would be to set a limit of the number of repeated

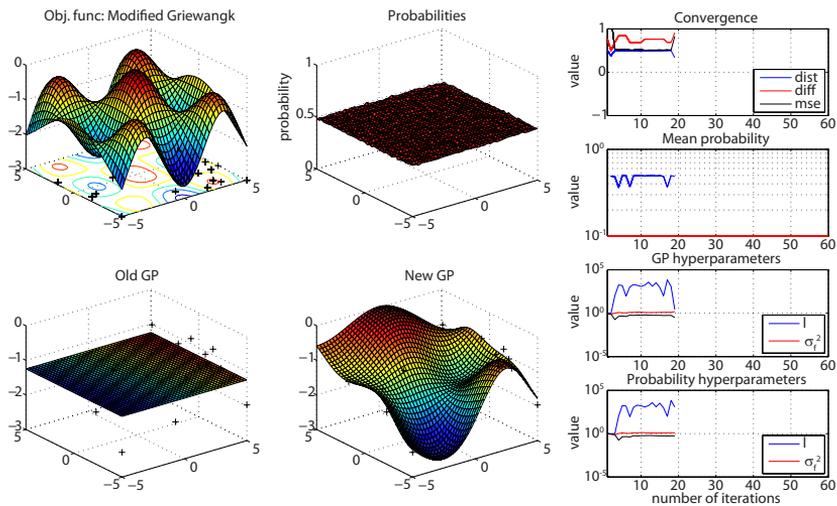


Figure 4.14: Noisy 2D simulation plot with $k_\sigma = 0.3$ for the 19'th iteration with a modified Griewangk function as the unknown underlying preference function.

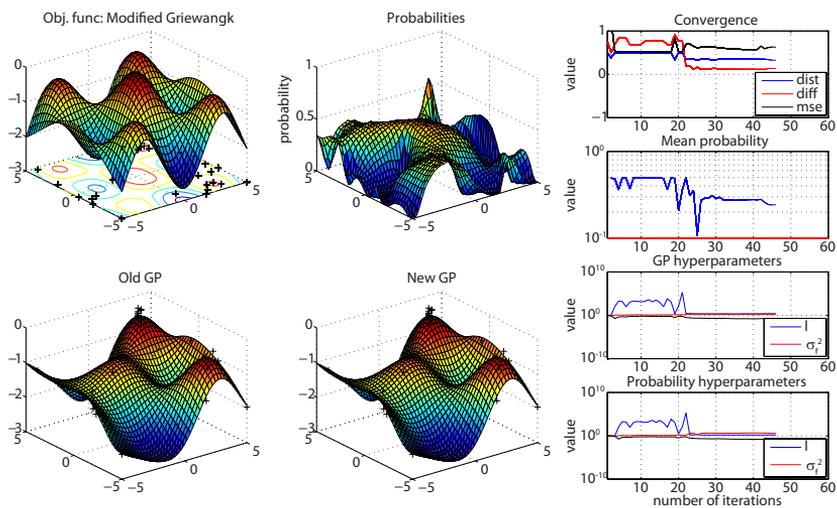


Figure 4.15: Noisy 2D simulation plot with $k_\sigma = 0.3$ for the 46'th iteration with a modified Griewangk function as the unknown underlying preference function.

observations. If this limit is reached the setting with the second highest max probability is suggested. This strategy is similar to the strategy used in the noise-free setting, but it might not be applicable for the noisy case. If a peak occurred in the noise-free setting it would be narrow, meaning that the max probability was only high exactly at the center of the peak. In this noisy example it is seen from figure 4.15 that the peak does not decrease as fast as in the noise-free simulations, hence around the maximum of the the peak the max probabilities are also relatively large. Consequently, the second highest max probability might correspond to a setting just next to the setting corresponding to the center of the peak. An inevitable result of this will be that a lot of redundant observations are used to move away from a local maximum. Naturally, this behavior will be influenced by the resolution of the setting step-size. In the presented example the resolution is relatively large, hence for smaller resolutions the settings close to the center of the peak might not get a high max probability. Nevertheless, it is inconvenient to have the performance influenced by the setting resolution.

Forcing the search into other important regions is the motivation for suggesting the second highest max probability, but instead it can result in redundant experiments in already explored regions. To actually force the algorithm into other regions, only maximum points of the max probabilities should be considered. If a particular setting has been suggested numerous times the second largest maximum point of the max probabilities should be suggested. In this way, the concept of suggesting experiments that have the highest probability of having larger preference than the current maximum is respected within particular regions. This would yield a difference. Obviously, to actually be able to find maximum points of the max probabilities, the sampling noise/error should be filtered out by smoothing the max probability, with e.g. low-pass filtering. In the work presented in this thesis such a strategy has not been investigated.

Conclusively, it has been shown that addition of noise does slow down the convergence time. On average, the preference value for the found optimal setting is reduced with increasing noise level as well. However, it is interesting to observe that the max probabilities contain a lot of useful information - how to optimally make use of this information must be further investigated in the future (see section 7.3).

4.4.4 Summary

The simulations have shown good performance by the novel active learning algorithm together with the GP for noise-free observations compared with Mackay's method and random sampling. The algorithm performance the way that is

desired, i.e., a useful maximum - local or global - is quickly identified. The algorithm has further the potential to escape from a possible local maximum and identify the global maximum.

With the addition of noise to the observations the performance is as expected considerably worse than in the noise-free case. However, a considerably amount of noise can be tolerated, still yielding acceptable performance, i.e., on average the search converges quickly, although the suggested optimal setting is not necessarily the global maximum. It might be possible to optimize the performance further by using the concept of the max probabilities in a more sophisticated manner, but the scope of the thesis has not allowed for a thorough study of such strategies (see section 7.3).

User-Driven Personalization Framework

In this chapter the baseline framework proposed in this project for capturing personal preferences among a finite set of settings in a simplified HA simulator will be presented.

Initially, in section 5.1 an earlier framework proposed by Heskes and de Vries (2005) will be described, since this probabilistically based framework has been the main inspiration for the framework developed in this thesis. Next, section 5.2 through section 5.4 explain the proposed framework. A simulation study of the full framework is presented in section 5.5 followed by final remarks in section 5.6.

5.1 Bayesian Utility Elicitation

This section will review a Bayesian framework for HA personalization developed by Heskes and de Vries (2005). The basic principle is to model preferences of subject i through a utility function $u_i(y) = U(y; w_i)$. The utility function (preference function) mapping from HA outputs (sound) $y \in Y$ to utility u_i is parameterized by a final set of parameters $w \in W$. However, Heskes and de Vries (2005) do not explicit state any model for the utility function, yet the

problem is addressed in Birlutiu et al. (2010).

The utility parameterization w_i is treated as a probabilistic variable and a prior distribution $p(w|a_i)$ is assigned over w_i . Individual w will be referred to as a given *utility state*. a_i denotes a *patient profile* containing information about subject i , such as for instance type of HL, age etc. It is assumed that such priors over utility states are available based on a collection of HI persons.

Next, the HA processes an input sound $x \in X$ resulting in the output stimuli y from above. Heskes and de Vries (2005) assign a probability function $p(x)$ over the sound input X . The HA contains a final set of tuning parameters $\theta \in \Theta$, which control the sound processing, such that a fixed input sound x and different HA parameters θ result in different output sound stimuli y . The sound processing in the HA will be denoted $y = F(x; \theta)$. Given the utility function $U(y; w_i)$, obviously, the goal is to find the “optimal” HA parameter setting θ_i for subject i that gives the highest utility averaged over the fixed collection of sounds $x \in X$. From this point on the subject index i will be omitted, because all subjects will in practice be considered separately. An overview of the user-hearing-aid system is shown in figure 5.1.

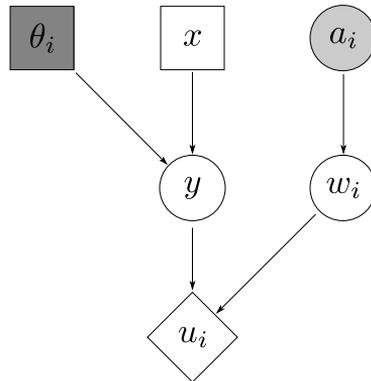


Figure 5.1: Bayesian decision network. $x \in X$ relates to an input, $\theta \in \Theta$ is the HA parameters transforming x to an output $y \in Y : y = F(x; \theta)$. The utility model $u(y) = U(y; w)$ is described by a final set of parameters $w \in W$ having a prior distribution given by the *patient profile* a . Dark gray refers to parameters subject to the optimization and light gray refers to parameters given in advance (Heskes and de Vries, 2005, figure 1(a)).

In practice, a paired comparison paradigm is a robust evaluation method, because the judgments are relative. To infer the paired comparison decision network an experiment e will be defined as picking an input x from X in combina-

tion with two different parameter settings θ_1 and θ_2 from Θ , i.e., $e = \{x, \theta_1, \theta_2\}$. The subject is then asked to choose the preferred stimuli resulting from the two settings, i.e., $y_1 = F(x, \theta_1)$ or $y_2 = F(x, \theta_2)$. The parameter $d \in \{-1, 1\}$ captures the choice from the subject with $d = 1$ indicating that y_1 is preferred over y_2 and similar $d = -1$ indicating that y_2 is preferred over y_1 . With the assumption that this paired-comparison forced-choice paradigm follows a logistic regression model, the decision from the subject is described by (Heskes and de Vries, 2005, equation (1))

$$p(d|e, w) = \frac{1}{1 + \exp\{-d \times [U(x; \theta_1, w) - U(x; \theta_2, w)]\}} \quad (5.1.1)$$

This paired-comparison forced-choice Bayesian experiment is visualized in figure 5.2. The probability density $p(w|D_n)$ over utility states w after having done

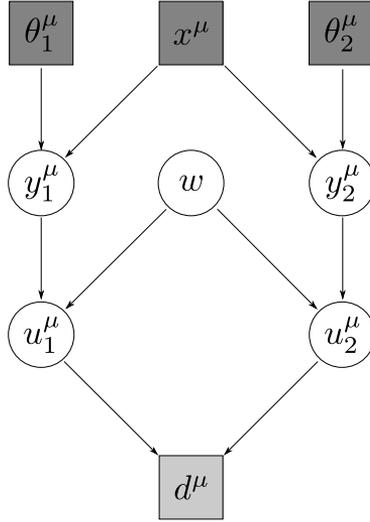


Figure 5.2: Bayesian experimental design visualizing the paired comparison paradigm. $d \in \{-1, 1\}$ captures the choice from the subject where $d = 1$ indicates that y_1 is preferred and $d = -1$ indicates that y_2 is preferred (Heskes and de Vries, 2005, figure 1(b)).

a new experiment $\{d^{n+1}, e^{n+1}\}$ can be obtained through Bayesian updating given by Bayes' theorem equation (3.1.2)

$$p(w|D_{n+1}) = p(w|d^{n+1}, e^{n+1}, D_n) = \frac{p(d^{n+1}|e^{n+1}, w)p(w|D_n)}{p(d^{n+1}|e^{n+1}, D_n)}, \quad (5.1.2)$$

where D_n consist of the tuples $\{e^\mu, d^\mu\}$, $\mu = 1 \dots n$ and μ indicates the experiment number of the n foregoing experiments. $p(d^{n+1}|e^{n+1}, w)$ is given by equa-

tion (5.1.1) and the term in the denominator follows from equation (3.1.3)

$$p(d|e, D_n) = \int_W p(d|e, w)p(w|D_n)dw, \quad (5.1.3)$$

where the superscript $n + 1$ has been omitted, which will be consistent from this point on, i.e., e and d refer to new experiments and D_n contains the n observed experiments.

In practice, it will not be possible to let a subject evaluate all possible parameter settings. Therefore, a cost function or equivalently a goal function is introduced to enable active data selection for new experiments. Naturally, a specific choice of cost/goal function defines the outcome of the optimization or learning process (ref. chapter 4). At this point it is assumed that a goal function $G(\theta, w, e, d)$ is defined, which encodes consequences of doing another experiment e with result d in a current state w . First of all the posterior density function over utility states w from equation (5.1.2) is used to marginalize the goal with respect to w resulting in the *posterior expected goal*. For HA personalization the task is to achieve the highest utility (preference) over all HA settings θ , therefore Heskes and de Vries (2005) argue that the optimal decision is to maximize the expected goal with respect to θ (Heskes and de Vries, 2005, equation (4))

$$G(e, d) \equiv \max_{\theta} \int_W G(\theta, w, e, d)p(w|d, e, D_n)dw, \quad (5.1.4)$$

Next, taken the expectation with respect to the outcome d of the experiment using equation (5.1.3) yields what Heskes and de Vries (2005) refer to as the *pre-posterior expected goal*

$$G(e) = \sum_{d=\pm 1} G(e, d)p(d|e, D_n). \quad (5.1.5)$$

The optimal experiment becomes the experiment that maximizes the pre-posterior expected goal and a Bayesian solution e^* to the experimental design becomes

$$e^* = \arg \max_e G(e) \quad (5.1.6)$$

$$= \arg \max_e \sum_{d=\pm 1} p(d|e, D_n) \max_{\theta} \int_W G(\theta, w, e, d)p(w|d, e, D_n)dw \quad (5.1.7)$$

$$= \arg \max_e \sum_{d=\pm 1} \max_{\theta} \int_W p(d|e, D_n)G(\theta, w, e, d)p(w|d, e, D_n)dw \quad (5.1.8)$$

Despite the usefulness of these derivation, the question still remains to define a proper and tractable cost/goal function. If the goal would be to infer the utility model (preference function) parameters w one common choice would be the log

difference between the posterior and prior density functions of w

$$G(\theta, w, e, d) = \log p(w|d, e, D_n) - \log p(w|D_n) \quad (5.1.9)$$

$$= \log \frac{p(w|d, e, D_n)}{p(w|D_n)} \quad (5.1.10)$$

Obviously, this goal function does not depend on θ , hence equation (5.1.4) becomes

$$G(e, d) = \max_{\theta} \int_W G(\theta, w, e, d) p(w|d, e, D_n) dw \quad (5.1.11)$$

$$= \int_W p(w|d, e, D_n) \log \frac{p(w|d, e, D_n)}{p(w|D_n)} dw \quad (5.1.12)$$

This equation is the Kullback-Leibler divergence between the posterior and prior distribution over w , thus with this kind of goal function the optimal new experiment would be the one that gives the largest information gain about the utility parameters w averaged over the outcomes of the experiment. This is exactly the ALM criterion from section 4.1.

For the purpose of HA personalization Heskes and de Vries (2005) suggest to use a goal function based on what they refer to as *expected utility*. Basically, this is the expectation with respect to the input sound files x of the modeled utility function

$$EU(\theta, w) = \sum_{x \in X} p(x) U(x; \theta, w) \quad (5.1.13)$$

Notice, that the utility function $U(x; \theta, w)$ models preferences directly on the HA parameter instead of on the output stimuli of the HA. This seems like a convenient and natural approximation, since now preferences among HA settings are modeled directly on the HA parameters. Further, assuming that all sound files within the context of X are equally likely, the expected utility boils down to the modeled utility function. Using this as the overall goal function, equation (5.1.5) becomes

$$G(e) = \sum_{d=\pm 1} G(e, d) p(d|e, D_n) \quad (5.1.14)$$

$$= \sum_{d=\pm 1} p(d|e, D_n) \max_{\theta} \int_W p(w|d, e, D_n) U(\theta, w) dw \quad (5.1.15)$$

$$= \sum_{d=\pm 1} p(d|e, D_n) \max_{\theta} \int_W \frac{p(d|e, w) p(w|D_n)}{p(d|e, D_n)} U(\theta, w) dw \quad (5.1.16)$$

$$= \sum_{d=\pm 1} \max_{\theta} \int_W p(d|e, w) p(w|D_n) U(\theta, w) dw \quad (5.1.17)$$

This is by Heskes and de Vries (2005) referred to as the *expected value given perfect information* and they embed active data selection by performing the experiment that maximizes this term. This criterion realizes the idea of performing new experiments that in the context of hearing aid personalization focus directly on the overall goal of maximum utility for a particular setting. As mentioned previously, this is in nature different from the majority of other active learning problems, where the main interest is to be confident about an underlying function over the entire input space or alternatively in a subspace of interest.

To summarize, the framework by Heskes and de Vries (2005) consists of three parts - a preference function model must be defined, the preference function model should be trained on pair-wise comparisons and finally active learning should be applied to perform the most informative experiment. Such a method is used in this thesis, but the three parts are addressed differently than what is sketched by Heskes and de Vries (2005). The reason for this is that Heskes and de Vries (2005) assume that the preference function is modeled by a parameterized model. It will be preferable to use a non-parameterized model, because these are more flexible than parameterized models. Expected utility is defined for parameterized models and how to apply the concept for non-parameterized models is not evident. Therefore, the three steps in the framework by Heskes and de Vries (2005) must be addressed differently, although the basic concepts in the framework appear to be a promising strategy. How the three parts are addressed in this thesis will be presented in the following three sections.

5.2 Modeling Preference Functions

A different notation will be used in this thesis than the notation used by Heskes and de Vries (2005). A HA consisting of \mathcal{D} adjustable parameters collected in the vector $\mathbf{x} = [x_1, x_2, \dots, x_{\mathcal{D}}]$ is considered. In practice each HA parameter x_k , where $k = 1, 2, 3, \dots, \mathcal{D}$, is bounded to a particular range and can take only a finite number of values m within that range.

Next, it is assumed that there for a particular user exists a preferences function $f_u(\mathbf{x})$ over HA settings, which describes the user preference for a particular HA setting in a given sound environment. The observed values (targets) of the user preference function are labeled y and are assumed to be related to the hidden preference functions f_u through contaminating independent zero-mean Gaussian noise with variance σ_n^2 . The unknown user preference function f_u is modeled by a zero-mean GP with a SE covariance function with an additional independent noise term $f \sim \mathcal{GP}(\mathbf{0}, K_{SE}(\mathbf{X}, \mathbf{X}))$. Predictions of the real user preference function values \mathbf{f}_* at particular HA settings collected in \mathbf{X}_* given N observations

of HA settings/preference values $(\mathbf{X}_N, \mathbf{y}_N)$ follow from equation (3.2.24).

The GP predictions of preference function values over HA settings are updated every time a new preference value y_{N+1} is observed, where N is the number of preference data points measured at a particular iteration. Naturally, the goal is to find the optimal HA setting \mathbf{x}_{opt} that maximizes the user preference function.

The user is always presented to the same sound file processed differently through the HA, hence the preference function $f_u(\mathbf{x})$ is only affected by the setting of the HA. Furthermore, for the sake of simplicity it is assumed that the user preference function over HA settings does not drift over time, which will never be true in a real setup, even with a comprehensive user training phase.

5.3 Transform Pair-Wise Judgments to Preferences

Typically, pair-wise preference judgments are adopted in subjective experiments, since it is easier for listeners to rate or judge two opportunities relative to each other rather than to give an individual absolute rating on a given scale. Transforming the pair-wise preference judgments to an actual preference scale for individual settings is not trivial. Chu and Ghahramani (2005) have derived a method for training a latent preference function modeled through a GP on pair-wise judgments. Although, inferring the distribution over the latent function directly on the pair-wise judgments seems like the optimal strategy, the time limitations did not allow for an implementation of the method. Instead, a less efficient strategy is used where the GP training step is divided into two separate parts. In this section a possible transformation of the pair-wise preference judgments into absolute preference scale values used in the GP model in section 5.2 is described.

The pair-wise preference judgments experiments are presented as modified *A/B* experiments, where the user is *forced* to decide which of the presented two options the user prefers. Further, the user has to express to what extend he or she prefers the dominating option on a 3 step scale ranging from *little*, *somewhat* to *clearly*. To summarize, an experiment consists of two HA settings, $\mathbf{x}_q^{(1)}$ and $\mathbf{x}_g^{(2)}$, where $q = 1, 2, 3, \dots, n$ and $g = 1, 2, 3, \dots, n$ are two different HA settings among all possible settings n . Next, the user rates and a label $d = \{-3, -2, -1, 1, 2, 3\}$ is assigned to each experiment, where $d < 0$ denotes the case where option 1 is preferred over option 2 and vice versa. The absolute value of d indicates to what extend the dominating option is preferred on the scale mentioned above.

To transform the pair-wise judgments described above into preference values which can be used to train the GP, it is assumed that the probability $P_j(\mathbf{x}_g^{(2j)} \succ \mathbf{x}_q^{(1j)})$ of option 2 to be preferred over option 1 from the j 'th experiment can be modeled with a sigmoid function of the form

$$P_j(\mathbf{x}_g^{(2j)} \succ \mathbf{x}_q^{(1j)}) = \frac{1}{1 + \exp[-(y_g - y_q) + b]}, \quad (5.3.1)$$

where y_g and y_q is the user preference function value for the g 'th and q 'th HA settings \mathbf{x}_g and \mathbf{x}_q , respectively, and b is a bias favoring either option 1 or option 2. Notice, that in this thesis it is assumed that there is no bias favoring either option 1 or option 2, e.i. $b = 0$.

The transformation of the pair-wise preference judgments to actual observed preference function values y is performed by minimizing the sum of squared errors function between the preference label d and the prediction from the sigmoid function with respect to the preference function values \mathbf{y}_N (used to train the GP) corresponding to the HA settings in \mathbf{X}_N . \mathbf{X}_N is the set of settings that occurs in the pair-wise experiments, i.e., no settings occur more than once in \mathbf{X}_N even if they have been used in several experiments. To do the minimization, the preference label d is mapped to six discrete probabilities P_d spread equally over the interval from 0 to 1. In this way the label $d = -3$ corresponds to a 1/12 chance for option 2 to be preferred over option 1, $d = -2$ corresponds to a 3/12 chance, $d = -1$ to a 5/12 chance, $d = 1$ to a 7/12 chance, $d = 2$ to a 9/12 chance and finally $d = 3$ corresponds to a 11/12 chance of option 2 to be preferred over option 1. The sum of squared errors function e_{MS} becomes

$$e_{MS} = \frac{1}{M} \sum_{j=1}^M \left| P_d^{(j)} - P_j(\mathbf{x}_g^{(2j)} \succ \mathbf{x}_q^{(1j)}) \right|^2 \quad (5.3.2)$$

and is minimized with respect to the preference function values \mathbf{y}_N . M is the number of pair-wise experiments. The method is sketched in figure 5.3.

The presented sum of squared errors cost function is continuous which makes it easier to minimize, yet it might not be possible to obtain a minimum equal to zero. Consider the example sketched in figure 5.4, where three settings \mathbf{x}_q , \mathbf{x}_g and \mathbf{x}_h for which the preference values y_q and y_g for the q 'th and the g 'th setting, respectively are much larger than the preference value y_h for the h 'th setting. Additionally, the preference value y_g are slightly larger than y_q . If three experiments are now performed assuming no bias, where h is presented versus q , h versus g and q versus g . Ideally, the results from these three experiments would be $d = 3$, $d = 3$ and $d = 1$. When the preference values are obtained by minimizing the described sum of squared errors cost function from equation (5.3.2), even the correct preferences values will result in a small error, because the distance

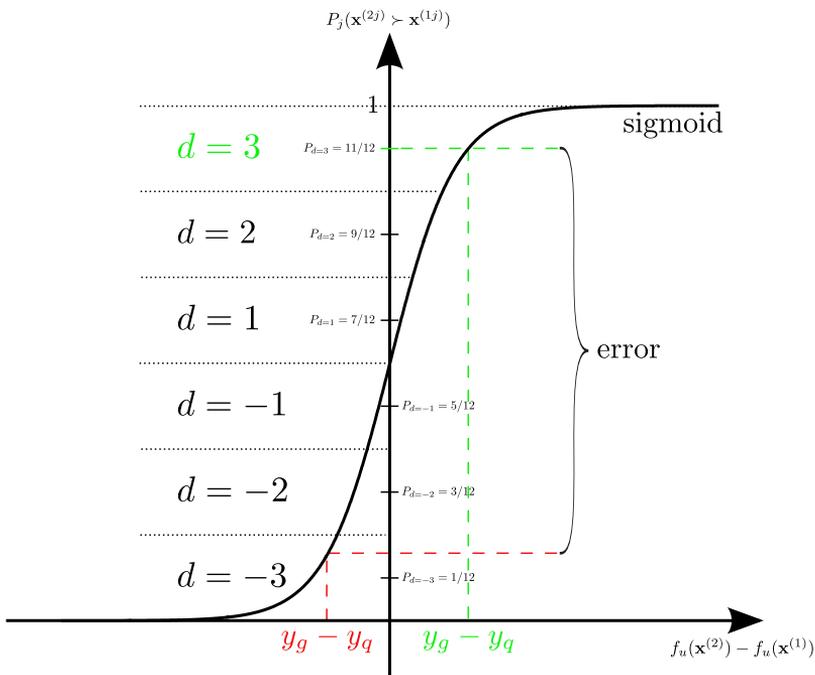


Figure 5.3: Sketch of the pair-wise transformation given an experiment j with HA settings \mathbf{x}_g and \mathbf{x}_q . From the subjective rating, the label $d = 3$ has been assigned to the particular experiment. The error is minimized with respect to the preference values y_g and y_q . The red configuration is an example of a starting guess in the minimization resulting in the shown error. The green configuration has minimized the error, i.e., an error of zero.

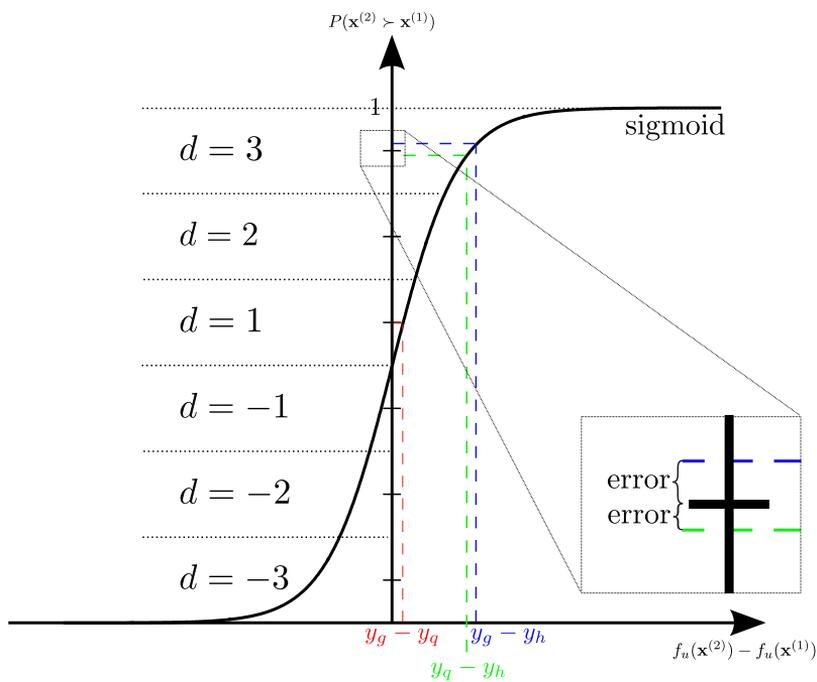


Figure 5.4: Given three pair-wise experiments, \mathbf{x}_g vs. \mathbf{x}_q with label $d = 1$, \mathbf{x}_q vs. \mathbf{x}_h with label $d = 3$ and \mathbf{x}_g vs. \mathbf{x}_h with label $d = 3$, then the $N = 3$ obtained transformed preference values y_g , y_q and y_h from the minimization of equation (5.3.2) results in the error shown in the zoomed box.

between the discrete probabilities $P_d^{(j)}$ and the probabilities from the sigmoid function (equation (5.3.1)) will not coincide. The reason for this is that to satisfy the experiment between q and g , the preference value for the q 'th setting must be (and is) larger than the preference value for g . On the other hand to obtain an error of zero from the two other experiments the preference values for the q 'th and the g 'th settings have to be equal. Consequently, the sum of squared errors cannot be zero, hence the obtained preference values from the minimization will be a trade-off between not introducing too large an error when considering any of the experiments. Furthermore, the preference values y_q and y_g resulting from the mean square error minimization in the example depicted in figure 5.4 can be significantly different from the real preference values. Increasing the preference values y_q and y_g results in the same output from the three experimental, but the error will increase. Therefore, the relative difference between the obtained preference values will in general not be correct. If the experiment between the q 'th and g 'th settings was not performed in the example depicted in figure 5.4, the cost function would be zero, resulting in identical preference values for the two settings, which is neither the right result. The previous considerations indicate, that the transformation itself probably gives rise to additional noise even in cases where the observer is noise free.

Another cost function that would not result in an error in the previously explained example, would be a cost function, which divides the probabilities predicted by the sigmoid into six equidistant intervals corresponding to the preference label d as depicted in figure 5.5. If the predicted probability for the outcome of a given experiment falls in the correct interval the error is zero, one if it falls in the first adjacent interval, two if it falls in the second adjacent interval and so forth. For correctly obtained preference values, this cost function will have an error of zero in a noise free case. The problem with this cost function is that it is a staircase function. Hence, in the minimization with respect to preference values two sets of slightly different preference values can lead to the same error. Therefore, the minimization is intractable with standard minimization tools.

The sum of squared errors function is proposed for the transformation, despite the disadvantages that it might lead to. Additionally, it must be mentioned that with this pair-wise transformation it is advisable to have all experiments "connected", meaning that if the first experiment consists of HA setting \mathbf{x}_g and \mathbf{x}_q , then the next should only contain one new setting, while either \mathbf{x}_g or \mathbf{x}_q is used once again. In this way all mapped preference function values are "connected" when minimizing equation (5.3.2). Hereby, no sets of preference function values can change arbitrary, since one preference function value influence the value of another through the sum of squared errors function. Notice, that a drawback from this constraint is that even though a new experiment consists of two HA settings, only one of them can result in a new GP observation point, although

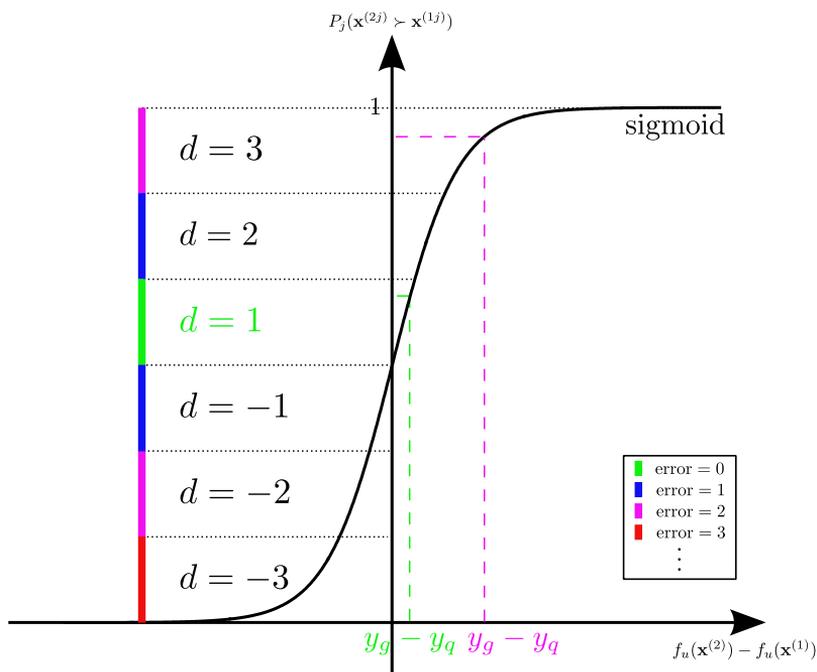


Figure 5.5: Sketch of a pair-wise transformation with a staircase cost function given an experiment j with HA settings \mathbf{x}_g and \mathbf{x}_q . From the subjective rating, the label $d = 1$ has been assigned to the particular experiment. The error is minimized with respect to the preference values y_g and y_q . The purple configuration is an example of a starting guess in the minimization resulting in a error of 2 (see legend box). The green configuration has minimized the error, i.e., an error of zero. The largest obtainable error is 5, which can only occur when an outcome of an experiment results in a label d equal to either -3 or 3.

it might alter the previous ones. The transformation method is investigated through simulations in section 5.5.

5.4 Active Learning

The last part of the baseline framework consists of applying active learning to select the optimal experiment. In this work optimal experiments refer to experiments that optimize user preference among HA settings. In section 4.3 a novel active learning criterion ALP for the case of optimizing user preference has been proposed. The active learning part in this thesis will be based on this concept. Naturally, it would be very interesting to study if the ALP criterion for the case of optimal user preference outperforms the two criteria ALM and ALC by Mackay (ref. section 4.1) and Cohn (ref. section 4.2), respectively. However, due to time limitations such a study have not been done, unfortunately.

As touched upon in section 4.3 the ALP criterion seems to have the potential to avoid getting stuck in a local maximum, thus the abilities exist, yet how to exploit them will be dependent on the actual preference function. Different proposals have been proposed during this thesis, only the reader should notice that there might exist other strategies in how to alter the parameters and in particular the hyper-parameters in the covariance function (see section 7.2).

5.5 2D Baseline Simulation

In this section the baseline algorithm including the modeling of the underlying preference function, the pair-wise transformation of the observation and the active learning part is investigated by a simulation. The simulation conditions are presented in section 5.5.1 and results and comments are given in section 5.5.2.

5.5.1 Simulation Conditions

The simulation setup is similar to the one used in section 4.4, but instead of direct elicitation of the preference values y , the observations consist of noise-free pair-wise preference judgments d . The preference judgments are given by mapping the difference in preference values resulting from the modified Griewangk function between the two options through a sigmoid P_{decision} given by equation (5.3.1) with no bias. Next, the interval between zero and one is divided

into six equidistant intervals each corresponding to a particular decision label, i.e., the interval from 5/6 to 1 corresponds to $d = 3$, the interval from 4/6 to 5/6 corresponds to $d = 2$ and so forth. The simulated decision label for a particular experiments is given by assigning the decision label d corresponding to the interval that the output from the sigmoid P_{decision} falls into. Additionally, a slope constant a has been introduced in the equation for the sigmoid, hence the decision sigmoid is given by

$$P_{\text{decision}} = \frac{1}{1 + \exp[-a \cdot (y_2 - y_1)]}, \quad (5.5.1)$$

where y_1 and y_2 is the preference value given by the modified Griewangk function for option 1 and option 2, respectively. The slope constant is adjusted in order for the standard deviation of the output from the sigmoid given all possible experiment to correspond to the standard deviation of the differences in preference values from all possible experiments normalized with respect to the maximum range of preference differences. This was done trying to simulate an optimal user.

Next, to ensure that all experiments are “connected” as described in section 5.3 the next proposed observation by the active learning algorithm is always judged versus the observations with the highest preference value. Alternatively, if the current best setting of the GP is not an observation, then this setting and the best setting among the already observed settings are judged versus each other. In this way a similar strategy as used for the noisy simulations from section 4.4.3 is adopted here as well.

80 iterations are performed. The mean of the max probabilities are given by $m_{\text{maxthres}} = 10^{-1}$ and infinitely many repeated observations are allowed. The GP covariance function is a SE with an independent noise contribution, where the hyper-parameters are learned from data by marginal likelihood maximization. The altered hype-parameter for the max probability calculations is the signal variance $\sigma_{f, \text{maxprob}}^2$ given by equation (4.4.2). The simulation conditions are summarized in table 5.1.

5.5.2 Results

Numerous simulations have been run and the overall picture from the simulations is that it is almost impossible for the GP to fit the pair-wise data mapped back to preference values by the sigmoid transformation described in section 5.3, even without any noise applied. Generally, it seems as if the transformation corrupts the data such that the marginal likelihood estimate of the hyper-parameters fits the data as being merely noise. As already mentioned in

| | |
|--------------------------------------|---|
| Underlying preference function | Modified Griewangk |
| Input space dimensions | 2D |
| x range | -5;5 |
| y range | -5;5 |
| Resolution in each dimension | 41 uniformly spaced points |
| Number of iterations | 80 |
| Covariance function | SE and independent noise |
| Covariance function hyper-parameters | l, σ_f^2, σ_n |
| m_{maxthres} | 10^{-1} |
| Altered hyper-parameter | $\sigma_{f, \text{maxprob}}^2 = \sigma_f^2(1+2 \cdot k \cdot \sigma_f^2)$ |
| Repeat observations | yes (infinitely) |
| Sigmoid slope constant a | 0.81 |

Table 5.1: Simulation conditions for noise-free pair-wise preference judgments simulations

section 5.3 this might be a results of the imperfections of the cost function used for backward mapping the pair-wise judgments to actual preference values. As a result of this, the GP fitted preference function becomes extremely unstable, hence the robustness of the method is very limited. This behavior will be illustrated by examples from the simulations, instead of showing curves for the performance as this is essentially very bad, because of the fragile estimate of the underlying preference function.

The data from three consecutive iterations (67'th to the 69'th) from the only simulation that turned out to be somewhat satisfactory is depicted in figure 5.6 to figure 5.8. These figures illustrate the unstable behavior of the preference function estimate. In figure 5.6 the previous GP state has captured the underlying preference function around the correctly identified global maximum. However, the addition of one new observation changes the preference function estimate significantly. In this example the maximum of the preference function is still located around the global maximum of the underlying preference function, although this is simply a coincidence. In figure 5.7 the addition of a second observation results in yet another preference function shape, although this is more similar to the actual underlying preference function. Finally, in figure 5.8 the fitted preference function has returned to the shape seen in the old GP from figure 5.6.

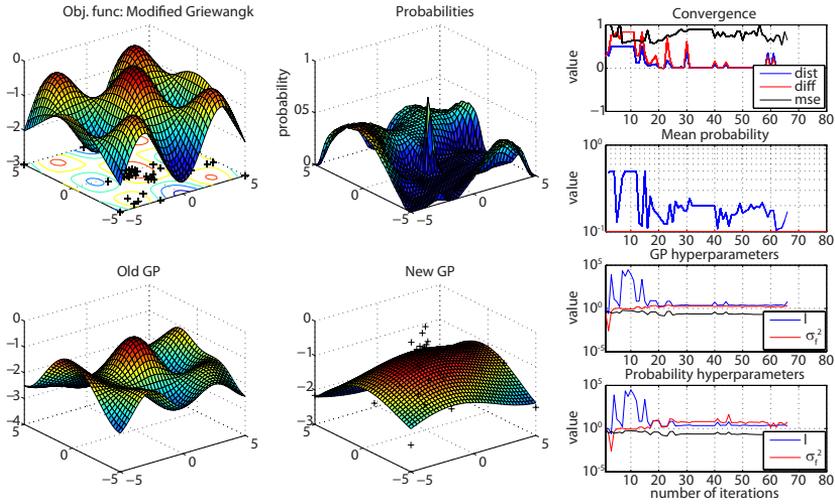


Figure 5.6: Noise-free 2D simulation with pair-wise transformation from preference judgments to preference values for the 66'th iteration with a modified Griewangk function as the unknown underlying preference function.

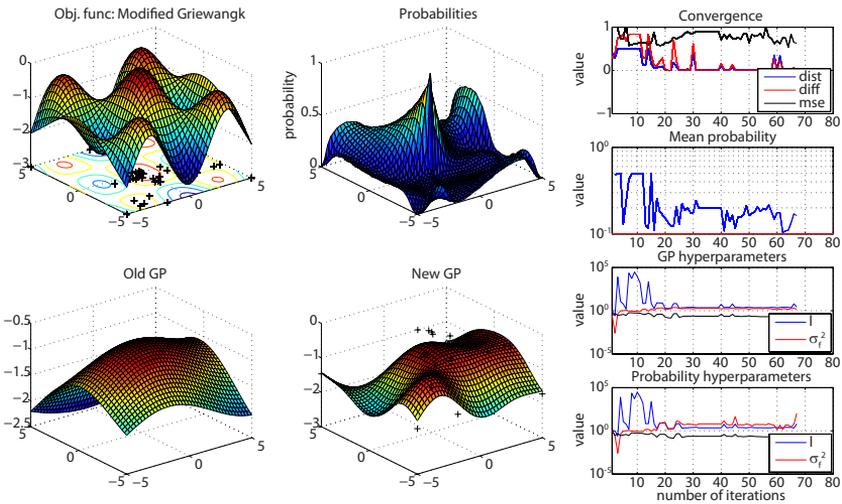


Figure 5.7: Noise-free 2D simulation with pair-wise transformation from preference judgments to preference values for the 67'th iteration with a modified Griewangk function as the unknown underlying preference function.

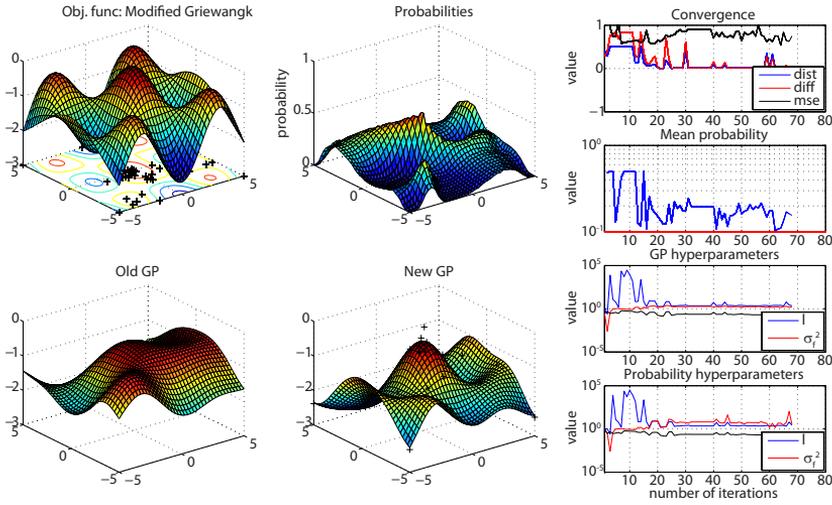


Figure 5.8: Noise-free 2D simulation with pair-wise transformation from preference judgments to preference values for the 68th iteration with a modified Griewangk function as the unknown underlying preference function.

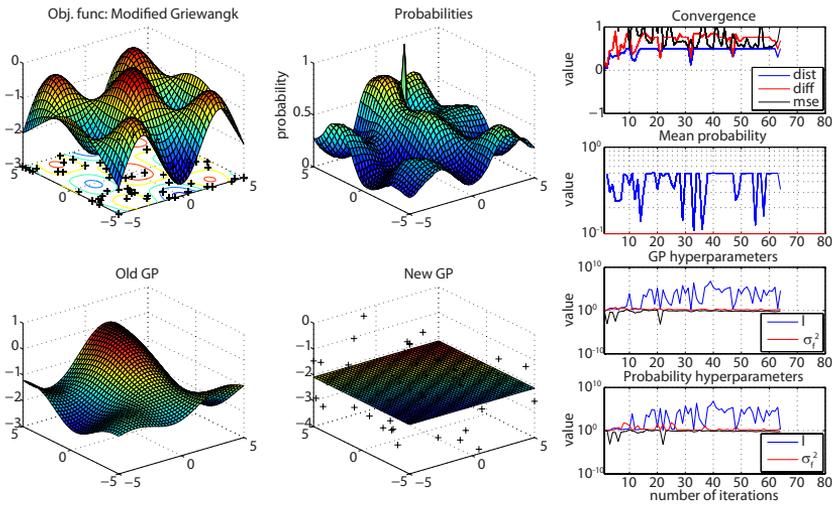


Figure 5.9: Noise-free 2D simulation with pair-wise transformation from preference judgments to preference values for the 64th iteration with a modified Griewangk function as the unknown underlying preference function.

Obviously, the preference function fit is extremely unstable, which is as mentioned probably a result of the imperfect backwards transformation of the preference values. With the behavior seen from these three figures, it is the backwards transformation that determines how the new preference function looks and therefore which observation that is performed next, rather than being the active learning algorithm that determines it. The active learning algorithm proposed in this thesis assumes that the GP can fit the data properly - especially around the current maximum - and this is seen to be extremely difficult with the pair-wise transformation. Recall, that there is not added any noise to the observations. Also, the reader should be aware that this particular simulation is the only example showing just acceptable performance.

In figure 5.9 the 64'th iteration of another simulation is shown. In contrast to the previous example, this simulation shows the typical behavior of the algorithm. Occasionally, the GP provides an estimate different from a plane. Sometimes these fits are somewhat similar to the underlying preference function and sometimes they are not - as the case in figure 5.9. However, the GP typically provides an estimate being simply a plane and thereby treats the fluctuations in the data as noise. Since, no noise is added to any of these simulations, the problem must occur from the pair-wise transformation. Therefore, it is concluded that the algorithm is not robust with pair-wise data, because the data cannot be fitted correctly. Hence, it is not sensible to investigate the behavior with the presence of noise in the observations. Instead, the GP must be trained directly on the pair-wise data as described in Chu and Ghahramani (2005), if the preference elicitation is done using a pair-wise experimental setup. Recently, Groot et al. (2010) have successfully applied this method for preference learning with normal hearing (NH) and HI listeners.

5.6 Summary

The pair-wise transformation has been shown to destroy the data in the majority of situations and it will therefore not be sensible to progress any further with the method. Instead, since a considerable amount of observation noise can be tolerated in the simulations shown in section 4.4 with direct observations of the preference values, the work continues with an experimental setup where subjects rate only one particular setting in each iteration on a virtual scale. However, keep in mind that such observations will be very noisy.

Pilot Experiment

To investigate the behavior and performance of the algorithm in a practical setup, a pilot experiment with four test subjects is performed. The experimental setup does not reflect a realistic automatic HA fitting situation in the sense that it consists of finding an optimal HA setting for a NH subject. Despite this, the setup is suitable to verify that the developed method can discover a preferred setting for individual users and estimate real preference functions as well. Further, it is possible to compare the functionality and performance of the algorithm with the findings in the simulations.

The experimental method is presented in section [6.1](#). Following this, the results are shown in section [6.2](#), and finally, the results are discussed in section [6.3](#).

6.1 Experimental Method

The experimental method for the pilot experiments is presented in this section, including stimuli, apparatus, subjects and procedure.

6.1.1 Stimuli

The stimuli in the pilot experiments are defined by two parts - the sound file fed to a HA simulator and the processing taking place in the particular HA simulator. The output from the HA simulator is the stimulus presented to the subjects in each iteration.

Sound file

The same 15 seconds looped sound file is used for all experiments. The sound file is a real recording of part of a conversation where a danish female speaker tells about an experience at a restaurant. On the recording there is background noise consisting of other conversations which are so weak that it is impossible to understand the context. Also, there is clear music in the background. At the end of some of the words the speaker makes a smacking sound with her lips and tongue.

HA simulator

The HA simulator consists of a simple compressor. The compressor is a 15 band filterbank compressor, where each band consists of one knee-point (K3 [dB]) also referred to as compression threshold, a compression ratio (CR) above the knee-point and a constant gain (G3 [dB]) below the knee-point as shown in figure 6.1. Additionally, the compressor has an attack speed (Att [dB/sec]) and a release speed (Rel [dB/sec]), which are not band dependent. The attack and release speeds are equivalent to the more common formulation using attack and release times. The speed defines how much the level at the output can increase or decrease per second, whereas time refers to how fast the level at the output settles around the resulting steady state output when exposed to an instant change in input level, e.g. ANSI/ASA S3.22-2009.

The knee-points K3 and gains G3 in the 15 bands are defined by

$$K3 = [38, 47, 50, 51, 53, 55, 53, 52, 52, 51, 51, 52, 51, 45, 45] \text{ dB} \quad \text{and} \quad (6.1.1)$$

$$G3 = [10, 14, 18, 18, 17, 18, 22, 22, 21, 19, 18, 18, 17, 16, 16] \text{ dB}, \quad (6.1.2)$$

where the first element in the vectors defines the value in the first band of the compressor, the second element defines the value in the second band and so forth. These settings are chosen in order to obtain a compressor setting typical for a

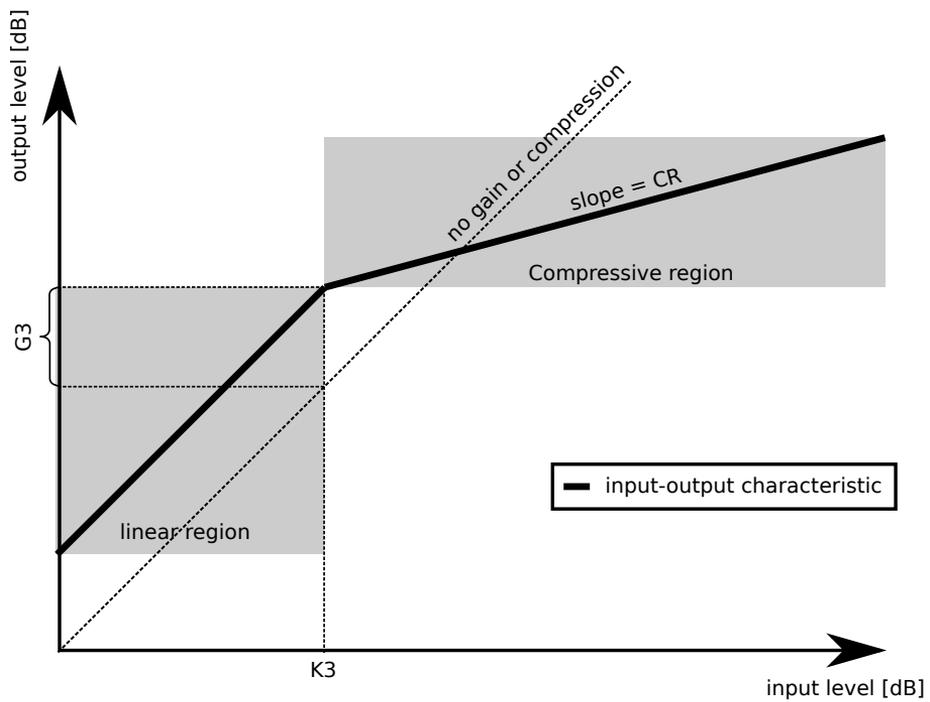


Figure 6.1: Schematic of the input-output characteristic of the compressor in one particular band, where $K3$ is the knee-point/compression threshold, $G3$ is the constant gain applied in the linear region and CR is the compression ratio.

mild to moderate HL. Identical compression ratios are set in each band. Thus, three free parameters define the input-output characteristic of the compressor.

Finally, the output of the compressor is always normalized to -20 dB FS¹ in MATLAB. This is done to prevent clipping.

6.1.2 Apparatus

The experiment is performed on a Windows XP PC in MATLAB. A Creative Sound Blaster Extigy is used as external sound card and Beyer Dynamic DT 770 PRO 250 Ohm headphones are used. The SPL in the headphones is not calibrated. Instead, subjects adjust the output to a comfortable level initially, and this level is kept fixed during a particular experiment. The experiments are conducted in an open office, i.e., no sound booth or particular listening room is used.

6.1.3 Subjects

Four male subjects did the pilot experiment, referred to as JBN, GST, OHA, and AWE. No audiometry was performed, but all test subjects reported to be NH. GST is not native danish, but understands danish, hence this was specified to be non-significant. JBN is the author of this thesis and the other subjects are employees of the research department at Widex A/S. Therefore, all test subjects have previous experience in listening experiments and are familiar with HA stimuli. All test subjects were instructed to rate the stimuli in relation to their overall preference, i.e., they were not instructed to focus on particular attributes.

6.1.4 Procedure

The procedure used in the pilot experiment is explained in this section. First the implemented GUI and subject task are presented followed by the experiment conditions.

¹FS refers to *full-scale*, hence 0 dB full-scale corresponds to a square-wave being fully spanned between -1 and 1 in MATLAB

GUI and Task

At the beginning of this project it was the outline that the experimental setup should use paired comparisons to elicit the preference for particular settings. However, the proposed method for handling the pair-wise data was seen to perform poorly in section 5.5 and therefore the method was discarded. Further, the time scope of the project did not allow for an implementation of a method using a GP trained directly on pair-wise judgment. Instead, it was observed that the algorithm can handle a considerable amount of noise on the direct observed preference values (see section 4.4.3). Therefore, it was decided to use an experimental setup where the user rates only one particular setting in each iteration, although such experiments are difficult for subjects to perform consistently.

To accommodate the latter, subjects do a training phase to get familiar with the range of the preference scale for a particular HA parameter space. After the training phase, i.e., before starting the actual experiment, subjects are instructed to consider their intention based on the given sound file. Further, a reference is included throughout the experiment, in order to prevent significant drift in the subjective ratings. The reference is defined as the sound file processed by the HA simulator (compressor) without any compression. Thereby, the only processing affecting the reference stimuli is the filterbank in the compressor.

The MATLAB GUI used for the experiment is depicted in figure 6.2 (see caption for an explanation).

Experiment Conditions

The conditions for the pilot experiment including real subjects, the HA compressor and the GUI are presented in this section. The conditions for the active learning algorithm are presented in this section as well.

The justification of the rather complicated algorithm developed in this thesis is that preference functions are believed not to have a simple shape, i.e., contain local maxima and minima. To confirm this, the underlying preference function defined by the settings in the HA simulator should be as complicated as possible. Addition of a simply noise reduction (NR) algorithm in the HA simulator was considered. However, due to the poor quality of the considered NR algorithm it was not believed to complicate the underlying preference function significantly. A fitted preference function over the three parameters in the used HA simulator can be visualized. Furthermore, it is believed that the three parameters do not

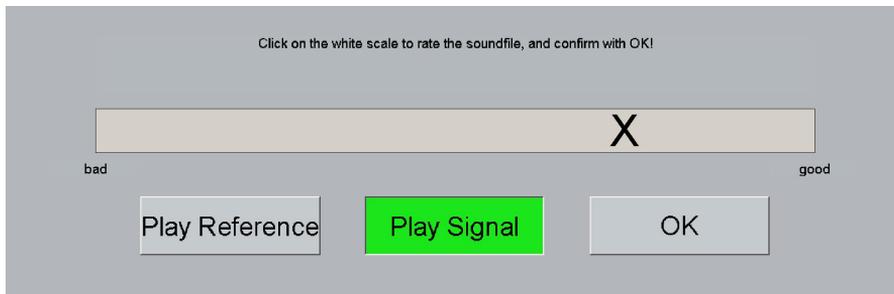


Figure 6.2: The MATLAB GUI used for the experiments. The cross on the white scale indicates the user rating of the processed sound file. The user can move the cross as many times as he or she wants by pressing elsewhere on the scale with the mouse cursor - first when the “OK” button is pressed the rating is confirmed moving on to the next experiment. The full range of the scale is mapped to a preference value between -10 and 0, where -10 corresponds to “bad” and 0 to “good”. When the “Play Signal” button is green the processed signal is played, while pressing the “Play Reference” button this turns green instead and the reference signal is played. The user can cross-fade between the processed signal and the reference as many times as he or she wants to. The switches are made real-time, so that the signal continues where the other one has gone to.

necessarily induce a simple preference function. This is the motivation for the use of the HA simulator described in section 6.1.1.

A grid is used for each of the three parameters restricting the parameters only to change in specified steps and in a given range. The discrete values for each of the three parameters are given by

$$\text{Att} \in \{30.0; 85.7; 244.9; 699.9; 2000.0\} \quad [\text{dB/sec}] \quad (6.1.3)$$

$$\text{Rel} \in \{30.0; 85.7; 244.9; 699.9; 2000.0\} \quad [\text{dB/sec}] \quad (6.1.4)$$

$$\text{CR} \in \{1 : 2; 1 : 4; 1 : 6; 1 : 8\} \quad (6.1.5)$$

The grid for the attack and release speeds are five log-spaced points from 30.0 dB/sec to 2000.0 dB/sec, i.e., equivalent to very long attack/release times to very short attack/release times. With these grids, the experiments consist of 100 different settings and naturally the optimal one should be detected as fast as possible.

The sequence of settings included in the training phase is fixed and thereby the same for all subjects. The sequence consists of 18 settings for all the combina-

tions resulting from the grid given by

$$\text{Att}_{\text{train}} \in \{30.0; 244.9; 2000.0\} \quad [\text{dB/sec}] \quad (6.1.6)$$

$$\text{Rel}_{\text{train}} \in \{30.0; 244.9; 2000.0\} \quad [\text{dB/sec}] \quad (6.1.7)$$

$$\text{CR}_{\text{train}} \in \{1 : 2; 1 : 8\} \quad (6.1.8)$$

Notice, that this corresponds to the ‘‘corner’’ points and points in between for the attack and release speeds. This sequence is chosen to present extreme settings for the subjects, in order to span the virtual preference scale as much as possible.

The covariance function in the GP providing the preference function estimate is a SE with an independent noise term. The active learning algorithm has the constraint, that only three identical observations are allowed. When the limit is reach, the algorithm suggests the setting with the second highest max probability and so forth. The threshold of the mean of the max probabilities is defined by $m_{\text{maxthres}} = 5 \cdot 10^{-2}$. When the mean of the max probabilities is lower than the threshold, the signal variance $\sigma_{f,\text{maxprob}}^2$ in the max probability calculations is altered according to equation (4.4.2). The search consists of 50 observations, i.e., 50 iterations are run, even if it the search seems to have converged earlier. The conditions for the algorithm is shown in table 6.1.

| | |
|--------------------------------------|--|
| Number of iterations | 50 |
| Covariance function | SE and independent noise |
| Covariance function hyper-parameters | l, σ_f^2, σ_n |
| m_{maxthres} | $5 \cdot 10^{-2}$ |
| Altered hyper-parameter | $\sigma_{f,\text{maxprob}}^2 = \sigma_f^{2(1+2 \cdot k \cdot \sigma_f^2)}$ |
| Repeat observations | yes (3) |

Table 6.1: Algorithm conditions in the pilot experiment

6.2 Results

The results from the experiments with the active learning algorithm in the practical setup including real subjects, the HA compressor and the GUI will be presented in this section. The section is divided into two parts. In the first part (section 6.2.1) the convergence points with each of the four subjects are determined by inspection of the predicted optimal setting during each search. In the

second part (section 6.2.2) the preference function estimates during the search are compared with the estimated preference function fit in the final iteration with each of the four subjects.

6.2.1 Sequence of Proposed Optimal Setting

The optimal setting is defined as the optimum among the possible observations defined by the discrete grid (see section 6.1.4).

Three subjects (JBN, GST and OHA) end up in the same optimal setting given by the smallest possible attack speed of 30 dB/sec, the largest possible release speed of 2000 dB/sec and the smallest possible compression ratio of 1:2 (see table A.1, table A.4 and table A.7 in appendix A). The three searches have found the final predicted optimum after 9, 6 and 8 iterations, although JBN and OHA have one iteration where a slightly different optimum is predicted. Some might argue that the search for OHA seems to have converged even earlier than the 8'th iteration, but as will be clear later the optimum predicted in iteration 7 is found to be a very bad setting of the HA, hence it will definitely be wrong to conclude that the search has converged before this iteration.

The search for test subject AWE seems to be doubtful about what the optimal setting is (see table A.10 in appendix A). Obviously, two settings seem to compete, both correspond to attack and release speeds given by 30 dB/sec and 2000 dB/sec, whereas the compression ratio should either be 1:6 or 1:8. These two settings are competing from iteration 6. This behavior could indicate that the fit is not completely reliable.

The converging iterations defined by inspecting of the proposed optimal settings during the searches for the four test subjects are summarized in table 6.2. The entire sequences of predicted optimal settings for each of the test subject during the experiments can be found in appendix A (table A.1, table A.4, table A.7 and table A.10).

| Subject | JBN | GST | OHA | AWE |
|----------------------|-----|-----|-----|-----|
| Converging iteration | 9 | 6 | 8 | 6 |

Table 6.2: Number of iterations until convergence determined by inspection of the proposed optimal settings during the search - table A.1, table A.4, table A.7, table A.10.

6.2.2 Preference Function Estimates

To gain further insight about at what point the different searches have converged, the final estimated preference function is compared with the individual fits during the search for each of the four test subjects, individually. The estimates will be depicted with a higher resolution than the resolution defined by the observation grid (see section 6.1.4), although the optimal settings are still defined as the optimum among the settings defined by the observation grid. A GP will estimate the interpolation between observations. Therefore, it is sensible to depict the preference function with a higher resolution in order to gain an additional detailed impression of the preference function shapes. The resolution is 21 uniformly spaced points in each parameter direction.

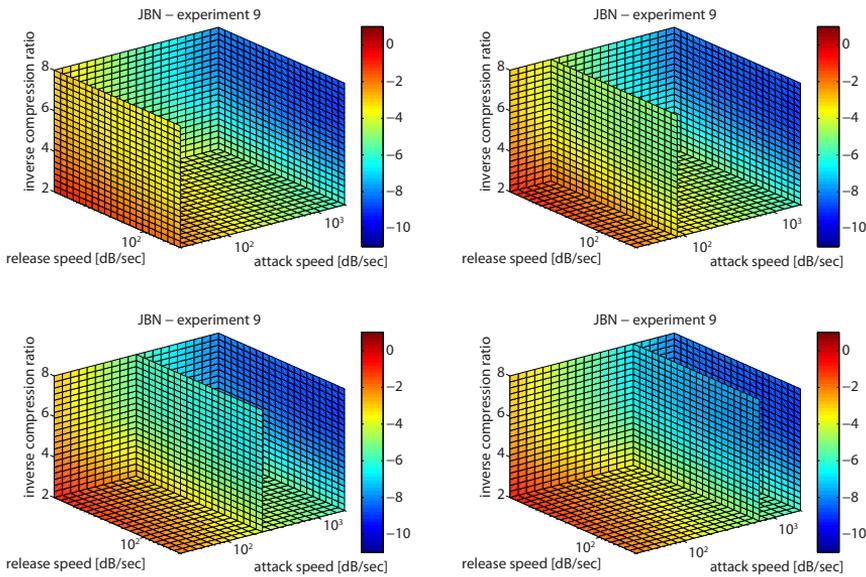


Figure 6.3: GP provided estimate of the preference function as a function of the attack speed (Att), release speed (Rel) and inverse compression ratio ($1/CR$) after the 9th observation for test subject JBN. The color scale indicates the preference. The four plots correspond to same preference function estimate, but the plane in the direction of the release speed and inverse compression ratio are moved along the attack speed axis. The resolution of the fitted preference function contains 21 uniformly spaced points in each direction as opposed to the rougher resolution of the discrete observation grid.

In figure 6.3 and figure 6.4 the fits after the 9th iteration and the final iteration,

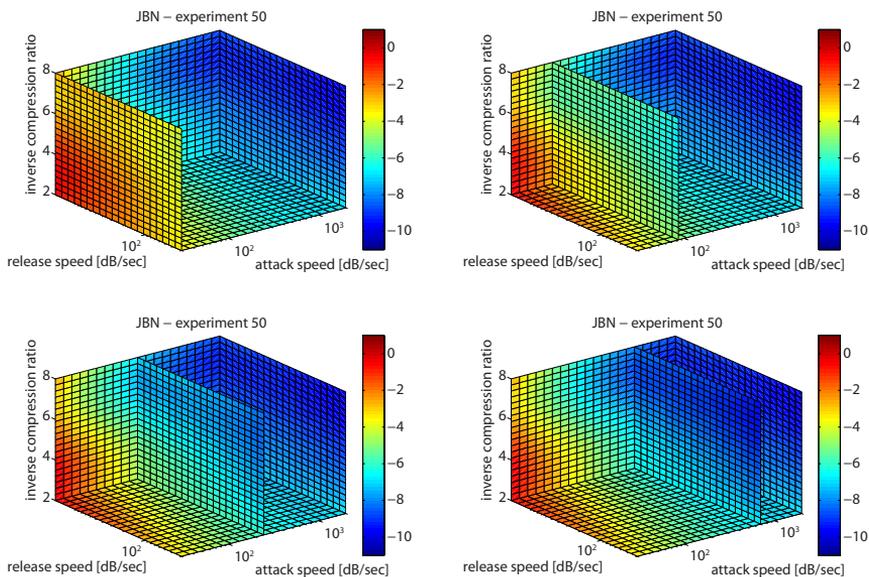


Figure 6.4: Final estimated preference function after the 50th observation for test subject JBN. The four plots are the same preference fit only the plane in the direction of the release speed and inverse compression ratio axis are moved along the attack speed axis. The resolution of the fitted preference function contains 21 uniformly spaced points in each direction as opposed to the rougher resolution of the discrete observation grid.

i.e., 50th iteration, for test subject JBN are depicted, respectively. The 9th iteration is the first iteration where the information caught by the GP fit is similar to the final fit. This point does actually correspond to the presumed convergence point for the search with JBN defined in section 6.2.1. From the preference function fits it is observed that the remaining iterations do reveal some further information about the assumed underlying preference function. For test subject JBN the preference in the final plot is almost unchanged in a relatively large region with compression ratios from 1:2 to 1:4, fast release speeds and slow attack speeds.

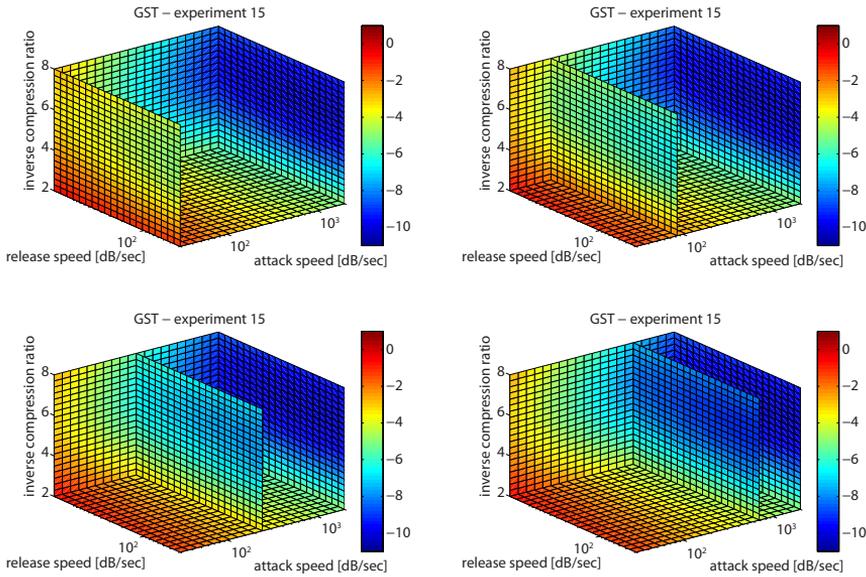


Figure 6.5: Estimated preference function after the 15th observation for test subject GST. The four plots are the same preference fit only the plane in the direction of the release speed and inverse compression ratio axis are moved along the attack speed axis. The resolution of the fitted preference function contains 21 uniformly spaced points in each direction as opposed to the rougher resolution of the discrete observation grid.

The 15th and final preference function fits for test subject GST are depicted in figure 6.5 and figure 6.6, respectively. The 15th iteration is the first iteration, where there seems to be similarity with the final preference function, hence this is a quite different convergence point than the one defined in section 6.2.1. Subject GST has an increased preference from compression ratios around 1:5 to higher compression ratios while the attack speed is slow. Thereby, a significant “valley” located around a compression ratio of 1:5 occurs in the preference function

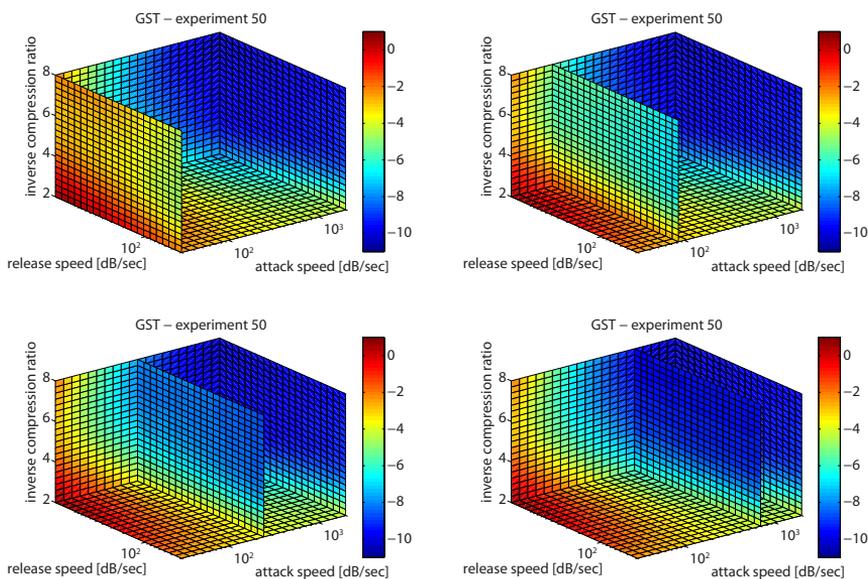


Figure 6.6: Final estimated preference function after the 50th observation for test subject GST. The four plots are the same preference fit only the plane in the direction of the release speed and inverse compression ratio axis are moved along the attack speed axis. The resolution of the fitted preference function contains 21 uniformly spaced points in each direction as opposed to the rougher resolution of the discrete observation grid.

estimate.

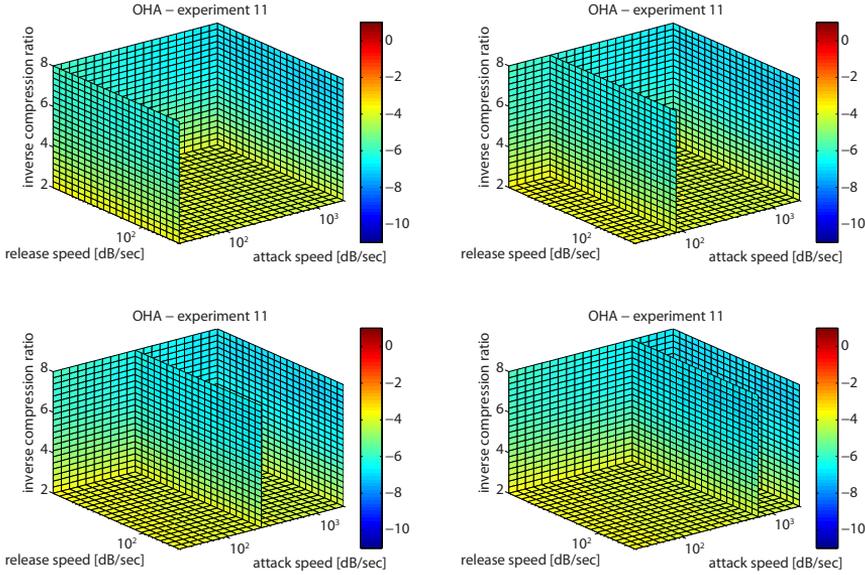


Figure 6.7: Estimated preference function after the 11'th observation for test subject OHA. The four plots are the same preference fit only the plane in the direction of the release speed and inverse compression ratio axis are moved along the attack speed axis. The resolution of the fitted preference function contains 21 uniformly spaced points in each direction as opposed to the rougher resolution of the discrete observation grid.

The comparison with the first obtained preference function showing similar information about the assumed underlying preference function as the final preference function for test subject OHA is depicted in figure 6.7 and figure 6.8. In section 6.2.1 it was proposed that the search has converged in the 8'th iteration. However, by inspection of the preference function fits, the convergence point seems to be the 11'th iteration. The tendency of large compression ratios obtaining good acceptance is not significantly present for subject OHA, yet the not preferred part of the parameter space is similarly detected as for JBN and GST.

The preference function from the 17'th iteration and the final iteration for test subject AWE are shown in figure 6.9 and figure 6.10, respectively. The 17'th iteration is the first fit showing similar tendencies as the final preference function. In section 6.2.1 the optimal setting for AWE did not seemed evident. This tendency also emerges in the fitted preference functions. The range of the fitted

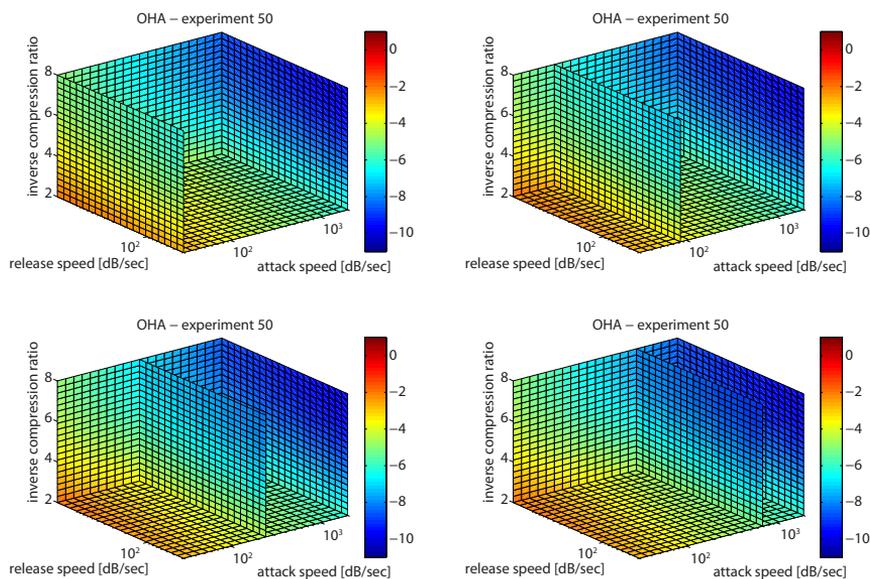


Figure 6.8: Final estimated preference function after the 50th observation for test subject OHA. The four plots are the same preference fit only the plane in the direction of the release speed and inverse compression ratio axis are moved along the attack speed axis. The resolution of the fitted preference function contains 21 uniformly spaced points in each direction as opposed to the rougher resolution of the discrete observation grid.

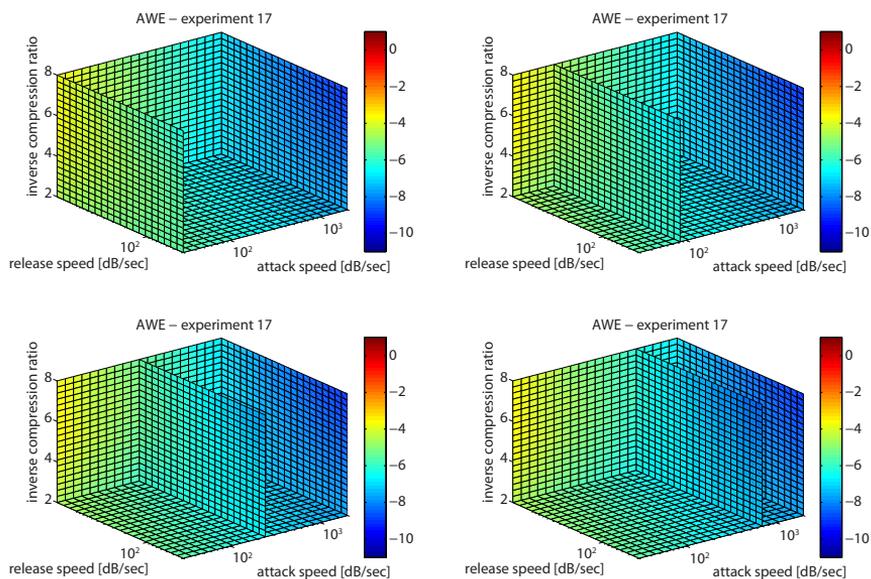


Figure 6.9: Estimated preference function after the 17'th observation for test subject AWE. The four plots are the same preference fit only the plane in the direction of the release speed and inverse compression ratio axis are moved along the attack speed axis. The resolution of the fitted preference function contains 21 uniformly spaced points in each direction as opposed to the rougher resolution of the discrete observation grid.

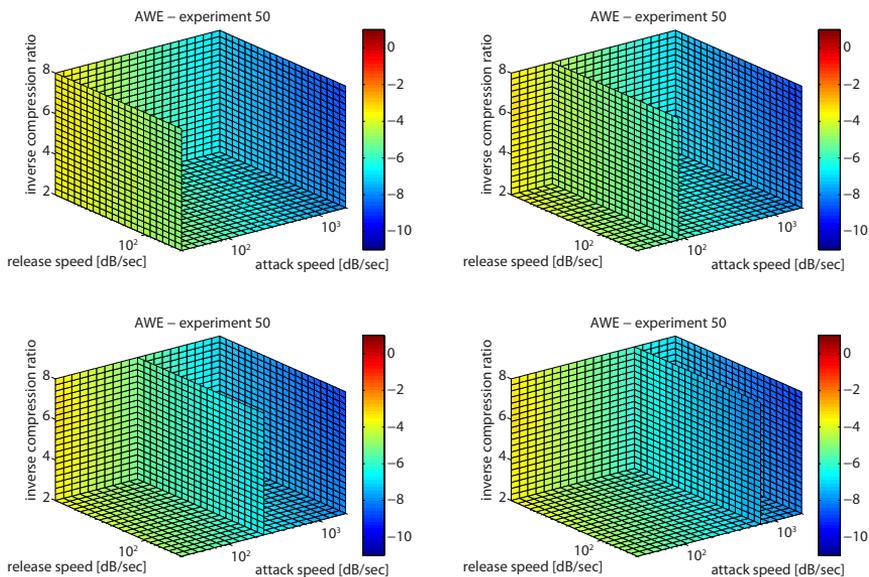


Figure 6.10: Final estimated preference function after the 50'th observation for test subject AWE. The four plots are the same preference fit only the plane in the direction of the release speed and inverse compression ratio axis are moved along the attack speed axis. The resolution of the fitted preference function contains 21 uniformly spaced points in each direction as opposed to the rougher resolution of the discrete observation grid.

preference values is very narrow and the preference values are merely concentrated around moderate values. It is only the slow and fast attack speeds that are confidently separated in relation to preference, whereas the variations in preference values along the two other directions - release speed and compression ratio - are not significant.

The converging iterations defined by inspecting of the fitted preference functions for the four test subjects are summarized in table 6.3.

| Subject | JBN | GST | OHA | AWE |
|----------------------|-----|-----|-----|-----|
| Converging iteration | 9 | 15 | 11 | 17 |

Table 6.3: Number of iterations until convergence determined by comparing the final preference functions with the preference functions during the search.

6.3 Discussion

Based on the sequence of proposed optima (section 6.2.1), specific convergence points for the individual searches are difficult to determine. The sequences do not express if a correctly proposed optimum is due to coincidence or due to sufficient confidence in the estimated preference function. When the search has converged, the estimated preference function should express a certain confidence about the fit - in particular around the estimated optimum - but also about the possible shape of the rest of the preference function. Therefore, only the boundaries for the convergence points of the searches can be expressed by presumed lower and upper limits. These can reasonably be specified by the convergence points from table 6.2 and table 6.3, respectively. It is not appropriate to estimate an actual mean convergence performance based on the results, because the amount of data are not sufficient. However, the results do give the impression that the baseline performance is satisfactory in relation to convergence time. The found optima for the subjects seem realistic, since NH subjects will generally not prefer compression.

By comparison of the final preference functions for the four test subjects it is seen that slightly different shapes are found, even for the three test subjects with the same predicted optimal setting. It would be wrong to over-interpret the results, but the structures fitted by the GP - especially for JBN and GST - are seen to be fairly complicated. High compression ratios and concurrent slow attack speeds (long attack times) will reduce high impact sounds, which might be preferred by some subjects. This tendency emerges in the estimated preference function for JBN and GST. The only significant high impact sound on the sound file used

in the experiment is the smacking tongue and lips sounds made by the female speaker. Probably, this smacking sound is experienced to be annoying by JBN and GST. Contrarily, reduction of the smacking sounds do not seem to have any preference by OHA and AWE, probably due to other side effects occurring from high compression. In comparison, high compression ratios with concurrent fast attack speeds generally induce a lot of unpleasant and annoying artifact. The low preference for high compression ratios and fast attack speeds by the four subjects is presumably a results of the emerging artifacts. The important conclusion to draw from these consideration is that the method seems to have the potential to discover individual preferences among subjects.

Despite the potential of the method, the results have revealed potential issues as well. If a subject assigns the top rating to a particular setting, the subject cannot in a later iteration rate another setting as being better than the previously top rated candidate, although the subject prefers the other setting over the previously top rated candidate. This problem may influence the results more dramatically than the well known - and modeled - observation noise, because the entire scale for the previous observations should be rescaled somehow. From figure 6.7 and figure 6.8 it can be seen that for OHA no settings are estimated to have the highest possible preference value (deep red) given the virtual preference scale. This either indicates that OHA did not use the top of the scale or that the virtual scale was not used consistently. Inconsistent ratings will result in a smoothing effect of the ratings. After the experiment, test subject OHA expressed that he did not use the top part of the scale, since he did not think that any of the given settings were good. Hence, the compressed preference value behavior seen in the fitted preference functions for OHA (figure 6.7 and figure 6.8) is most likely a result of this. In comparison, significant compression of the estimated preference values for AWE can be observed in figure 6.9 and figure 6.10. AWE reported that he did indeed use the entire range of the scale. Since, this does not emerge in the preference function estimates, the compressed behavior is probably caused by inconsistent ratings. As a results, the GP smooths sudden variations in the observations. Consequently, the range of the fitted preference values are compressed. The training phase was introduced to reduce inconsistent ratings, but obviously they have not been significantly reduced. In addition to this, it is rather doubtful whether or not it has been a good idea to apply the reference sound. Without the reference some subjects complained that it was difficult to rate a new setting and they believed that they only rated the setting relatively to the previous setting, whereas with the addition of the reference others complained that it was difficult to rate a new setting relatively to the reference, since it was not necessarily the best candidate. Nevertheless, the reference possible fixed the scale slightly for some subjects, while others did not used the reference at all or found it misleading. A setup with additional robustness must inevitable be invented to accommodate inconsistent ratings. Suggestions regarding this will be discussed in section 7.1.

The observations for each of the three test subjects is found in table A.2, table A.3, table A.5, table A.6, table A.8, table A.9, table A.11 and table A.12. The observation locations are plotted in figure A.1, figure A.2, figure A.3 and figure A.4.² This information shows that the algorithm suggests experiments more densely in regions with high preference and only few in regions which turn out to have low preference. In the simulations, the observation points are more densely distributed in high preference regions than in low preference regions. Hence, this desired behavior is maintained in the experiments.

The pilot experiment does not give reason to believe that the proposed method including the developed active learning algorithm cannot be used to address these types of problems. Instead the algorithm shows to be a proper basis. The algorithm quickly tracks a good setting and afterwards explores other regions to gain additional information about the assumed underlying preference function. However, the method adapted in this experiment does prove to lack a robust interaction with the user.

²The author is aware, that it is difficult to spot the locations of each observation in the figures from appendix A. The exact locations do not appear before the figures have been rotated in MATLAB

Future Work

This thesis has investigated possibilities for an automatic user-driven personalization procedure of HAs. A developed novel active learning criterion has been used in a framework, which forms a proper basis for further development. However, this thesis has also revealed a vast number of issues with the framework - and numerous potential solutions. This section presents future research topics that can further improve the framework used in this thesis.

7.1 Experimental Paradigm

The baseline proposed in this thesis lacks a robust experimental paradigm, that effectively limits the influence from noisy and inconsistent feed-back by subjects (see section 6.3).

A typical experimental paradigm used for perceptual listening experiments are relative judgments between instances, such as two-alternative forced choice, three-alternatives forced choice etc. The two-alternatives forced choice paradigm, i.e., paired comparisons, have been applied in the field of preference learning (Groot et al., 2010) and shown to be applicable. The method assumes - as in this thesis - a latent preference function modeled by a GP learned directly from pair-wise data. The method appears to be a promising basis for modeling

individual preference functions among subjects. The active learning criterion developed in this thesis is in principle directly applicable for determining new experiments with that type of GP formulation. The paired comparison experimental paradigm is presumably a significant improvement of the baseline framework. Other transformation methods might also be adapted to transform paired comparison data into scales, for instance Wickelmaier and Schmid (2004), but intuitively the optimal strategy is to train the GP directly on the pair-wise data. Conclusively, the method would be interesting to include in a future version of the algorithm.

Another possibility of an experimental setup, which has not been suggested previously in the field of preference learning, is to make a modified-MUSHRA¹ setup with an adaptive scale compensating for inconsistent ratings. How exactly an adaptive scale can be applied has to be studied. A modified MUSHRA test setup could consist of presenting the current lowest and highest preference points, together with the previous new data point and the new data point suggested by the active learning criterion.

7.2 Altering the Hyper-Parameters

With the novel active learning criterion it is possible to alter the hyper-parameters in the covariance function to obtain a more global search (see section 4.3 and section 4.4). In the noise-free simulations a good performance was obtained by only adjusting the signal variance σ_f^2 , whereas it was found not to be sufficient in the noisy simulations, where the length scale was altered additionally to obtain a better performance. It is yet unclear, which hyper-parameters to adjust, when a more global search is desired, hence this needs further study.

7.3 Induce a Global Search and Stop Criterion

In the pilot experiment presented in chapter 6 three identical observations were allowed to be repeated, otherwise the setting with the second highest max probability was suggested and so forth until the constraint was not violated. As mentioned during this thesis (see section 4.4), this might not be the optimal strategy, especially when the amount of possible settings is increased in the future. Therefore, more intelligent criteria should be investigated. In this thesis the mean over the max probabilities was used to alter the hyper-parameters (see

¹MUSHRA tests are described in Bech and Zacharov (2007)

section 4.4). This appears to be a natural procedure, because if the mean over the max probabilities is low given a current GP state, the preference gain in the next iteration is presumably minimal. Thereby, it could be interpreted as a convergence detection criterion. However, the mean over the max probabilities was also observed to be a sufficient tool to embed a global search after a good local maximum was obtained (see section 4.4.2). Therefore, the mean of the max probabilities might not be optimal to use as a stop criterion additionally.

The maximum points of the max probabilities are probably useful information, since the maximum points express the max probability in different regions in HA parameter space. Thereby, the search will possibly be further global, without violating the local fine adjustment of the setting. Probably, the maximum points of the max probabilities avoid the use of redundant observations at points next to a correctly detected local maximum (as seen in section 4.4.3). It would be naturally to study how this alternative use of the max probabilities would perform.

It might turn out that the optimal strategy is to limit the number of repeated observations, but a thorough study would be favorable. Correspondingly, the optimal stopping criterion might simply be when the predicted optimum setting has not changed throughout several iterations.

7.4 Preference Elicitation Strategy

In this thesis the subjects were instructed to rate individual stimuli according to preference in a completely general sense (see section 2.2). Possibly, this is not the optimal strategy for elicitation of user preference. A comprehensive study in how to elicit user preference is therefore required. Knowledge about user intentions in different sound environments would serve as crucial information, in order to provide proper instructions to the users about their intentions during an experiment. Consequently, the experiments would be easier for the users to conduct resulting in more reliable results.

7.5 Test-Retest Experiments

In this study it has not been verified that the found optimal settings, i.e., preference, for the four subjects could be recreated in a retest experiment. Thorough test-retest experiments should therefore be performed in the future, to verify the findings (see section 6.2) in this thesis.

7.6 Use the Algorithm in a Realistic Setup with Hearing Impaired Subjects

The sound-user environment (see chapter 6) imitated in this thesis is not reflecting a realistic situation. When a suitable performance of the algorithm has been verified (see section 7.5), the algorithm performance should be tested for realistic HA fitting using HI subjects. In such a study, it should be verified if a HI subject does in general prefer the personalized setting proposed by the interactive user-driven framework among the optimal settings discovered for other HI subjects.

7.7 Cluster Hearing Impaired Subjects based on the Obtained Preference Function

Assume that individual optimal settings of a HA have been discovered for numerous HI subjects by the use of an initial version of the algorithm. The algorithm has thereby found an estimated preference function for each individual HI subject as well. Additionally, a *subject profile* of each individual HI subject has been established, including features such as type of HL, age, mental condition, daily environment, level and type of education etc. It would be useful to study if it would be possible to cluster the individual estimated preference functions into different groups and train a machine learning classifier to map from a particular HI subject profile to one of the groups.

From a HA research perspective this would potentially uncover unknown information about how peoples daily environment influences the optimal setting of a HA and about which personal features that determine the preferences among HA settings.

7.8 Use Cluster Preference Function as Prior over Functions in the GP

A method to include a prior preference function and corresponding uncertainty in a GP was derived in section 3.2.3. The method becomes useful if it is possible to define a proper mapping from subject profiles to particular groups of subjects with similar properties in relation to preferences (see section 7.7). Consequently,

the mean and variance over preference function values for a particular group of subjects can be used as an initial prior estimate of the preference functions. This will potentially improve the convergence performance, because from the beginning the search will focus on regions with high max probabilities given the prior mean and variance. If a particular subject is not similar to the group that he or she is initially assigned to, the algorithm should ideally have the same capability to find an optimal setting as with a traditional zero-mean GP. These scenarios should naturally be investigated.

Conclusion

In this thesis a probabilistic inter-active user-driven personalization framework to discover the optimal setting of HAs for individual users based on their preferences has been proposed. The framework is based on modeling a latent preference function over HA settings. The latent preference function has been modeled by a GP with a SE covariance function trained directly on elicited preference values. A novel active learning criterion has been developed, which expresses the probability for a given setting to obtain higher preference than the current best setting.

Noise-free simulations with the novel active learning criterion have shown that it is suitable for preference learning. An algorithm consisting of a GP and the active learning criterion discovers a potential local maximum and iterates towards the global maximum. By comparing three sets of 100 simulations, the criterion was compared with random selection of new observations and with Mackay's criterion, which strives to maximize the total information gain. Random selection of new observations was as expected not an efficient strategy. The observed behavior of the novel and Mackay's criterion was essentially different, even though the two methods on average converged to the global maximum in the same number of iterations. On average, the novel criterion detects a maximum which has a 10 percent lower preference value than the global maximum after 17 iterations. In comparison, Mackay's criterion uses 24 iterations for a similar achievement.

Subsequently, addition of noise to the simulations showed that the convergence performance of the algorithm was reduced with increasing noise level and the preference value of the discovered optimal setting was correspondingly worsened. However, it was concluded that for realistic noise contributions the performance was satisfactory.

Besides a GP modeling a latent preference function and the novel active learning criterion, the baseline framework consisted of a transformation between pairwise preference judgment observations and actual preference values. Simulations with the baseline framework revealed that the transformation corrupted the data resulting in unstable preference function fits. Instead, a pilot experiment was conducted with direct elicitations of preference values.

Overall, the results from the pilot experiments showed similar tendencies as in the simulations. The experiment focused on the optimal setting of attack and release speeds together with identical compression ratios in each band of a 15 band filterbank compressor. After having discovered a good setting of the three parameters quickly, the algorithm exploits additional information about the assumed underlying preference function. The discovered preference function for each of the four test subjects indicated that personal preferences in the used compressor setup do exist, even for the three test subjects that ended up in the same optimal setting.

Finally, the framework investigated in this thesis appears to form a proper basis for preference learning in relation to optimize personalization of HAs, although a robust experimental setup must be invented.

APPENDIX A

Relevant Experimental Data from the Pilot Experiment

A.1 JBN

| Predicted optimal setting for JBN | | | |
|-----------------------------------|---------------|---------------|-----|
| Iteration | Att | Rel | CR |
| 1 | 2000.0 dB/sec | 30.0 dB/sec | 1:2 |
| 2 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 |
| 3 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 |
| 4 | 30.0 dB/sec | 30.0 dB/sec | 1:8 |
| 5 | 30.0 dB/sec | 30.0 dB/sec | 1:8 |
| 6 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 |
| 7 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 |
| 8 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 |
| 9 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 |
| ⋮ | ⋮ | ⋮ | ⋮ |
| 25 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 |
| 26 | 30.0 dB/sec | 2000.0 dB/sec | 1:4 |
| 27 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 |
| ⋮ | ⋮ | ⋮ | ⋮ |
| 50 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 |

Table A.1: The predicted optimal setting during the search for JBN

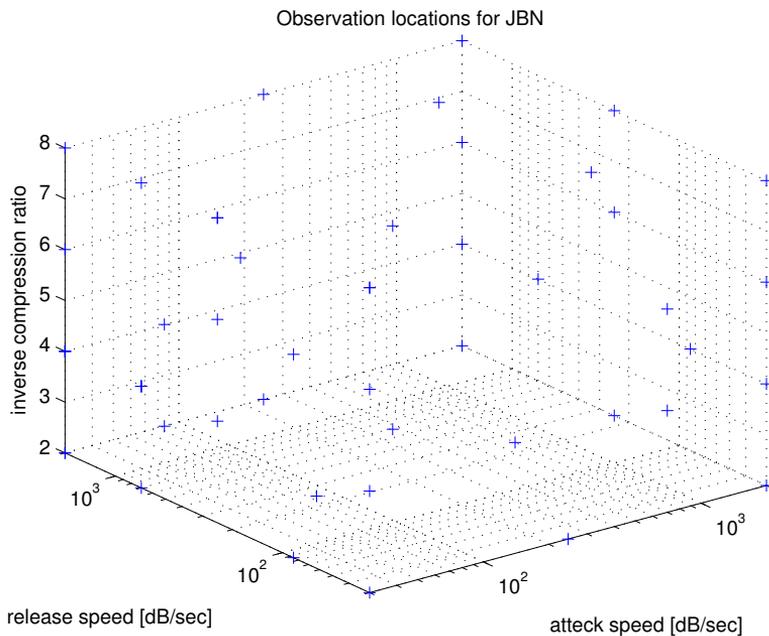


Figure A.1: Observation locations for test subject JBN.

| Observations with JBN | | | | |
|-----------------------|---------------|---------------|-----|-------------|
| Iteration no. | Att | Rel | CR | Scale value |
| 1 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 | -2.7 |
| 2 | 2000.0 dB/sec | 30.0 dB/sec | 1:2 | -6.3 |
| 3 | 2000.0 dB/sec | 2000.0 dB/sec | 1:8 | -8.4 |
| 4 | 30.0 dB/sec | 244.9 dB/sec | 1:8 | -1.1 |
| 5 | 30.0 dB/sec | 30.0 dB/sec | 1:8 | -4.0 |
| 6 | 699.9 dB/sec | 699.9 dB/sec | 1:8 | -7.9 |
| 7 | 30.0 dB/sec | 2000.0 dB/sec | 1:6 | -3.4 |
| 8 | 30.0 dB/sec | 30.0 dB/sec | 1:8 | -4.2 |
| 9 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 | -0.5 |
| 10 | 30.0 dB/sec | 699.9 dB/sec | 1:2 | -2.8 |
| 11 | 2000.0 dB/sec | 2000.0 dB/sec | 1:2 | -7.5 |
| 12 | 30.0 dB/sec | 2000.0 dB/sec | 1:4 | -0.9 |
| 13 | 2000.0 dB/sec | 85.7 dB/sec | 1:4 | -6.9 |
| 14 | 2000.0 dB/sec | 30.0 dB/sec | 1:8 | -9.6 |
| 15 | 30.0 dB/sec | 2000.0 dB/sec | 1:4 | -0.1 |
| 16 | 30.0 dB/sec | 2000.0 dB/sec | 1:4 | -1.9 |
| 17 | 30.0 dB/sec | 699.9 dB/sec | 1:4 | -1.2 |
| 18 | 30.0 dB/sec | 30.0 dB/sec | 1:4 | -4.1 |
| 19 | 30.0 dB/sec | 244.9 dB/sec | 1:8 | -2.9 |
| 20 | 85.7 dB/sec | 2000.0 dB/sec | 1:2 | -1.7 |
| 21 | 30.0 dB/sec | 699.9 dB/sec | 1:4 | -0.5 |
| 22 | 30.0 dB/sec | 85.7 dB/sec | 1:6 | -3.1 |
| 23 | 30.0 dB/sec | 699.9 dB/sec | 1:4 | -1.3 |
| 24 | 85.7 dB/sec | 85.7 dB/sec | 1:4 | -5.1 |
| 25 | 699.9 dB/sec | 30.0 dB/sec | 1:6 | -9.3 |

Table A.2: The observations made with test subject JBN for the 1st to the 25th iteration

| Observations with JBN | | | | |
|-----------------------|---------------|---------------|-----|-------------|
| Iteration no. | Att | Rel | CR | Scale value |
| 26 | 85.7 dB/sec | 699.9 dB/sec | 1:6 | -6.0 |
| 27 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 | -0.4 |
| 28 | 2000.0 dB/sec | 2000.0 dB/sec | 1:4 | -8.3 |
| 29 | 244.9 dB/sec | 30.0 dB/sec | 1:2 | -6.8 |
| 30 | 2000.0 dB/sec | 244.9 dB/sec | 1:6 | -9.3 |
| 31 | 699.9 dB/sec | 244.9 dB/sec | 1:2 | -6.8 |
| 32 | 2000.0 dB/sec | 2000.0 dB/sec | 1:6 | -8.8 |
| 33 | 30.0 dB/sec | 244.9 dB/sec | 1:6 | -2.8 |
| 34 | 85.7 dB/sec | 85.7 dB/sec | 1:8 | -6.6 |
| 35 | 85.7 dB/sec | 2000.0 dB/sec | 1:4 | -3.3 |
| 36 | 30.0 dB/sec | 244.9 dB/sec | 1:4 | -2.4 |
| 37 | 699.9 dB/sec | 30.0 dB/sec | 1:4 | -9.2 |
| 38 | 244.9 dB/sec | 2000.0 dB/sec | 1:8 | -7.6 |
| 39 | 2000.0 dB/sec | 699.9 dB/sec | 1:4 | -8.6 |
| 40 | 30.0 dB/sec | 699.9 dB/sec | 1:8 | -2.3 |
| 41 | 2000.0 dB/sec | 244.9 dB/sec | 1:8 | -8.8 |
| 42 | 30.0 dB/sec | 30.0 dB/sec | 1:2 | -3.2 |
| 43 | 244.9 dB/sec | 2000.0 dB/sec | 1:2 | -5.3 |
| 44 | 30.0 dB/sec | 30.0 dB/sec | 1:6 | -4.0 |
| 45 | 2000.0 dB/sec | 30.0 dB/sec | 1:6 | -9.1 |
| 46 | 2000.0 dB/sec | 244.9 dB/sec | 1:2 | -7.5 |
| 47 | 30.0 dB/sec | 85.7 dB/sec | 1:2 | -2.4 |
| 48 | 2000.0 dB/sec | 30.0 dB/sec | 1:4 | -8.7 |
| 49 | 85.7 dB/sec | 244.9 dB/sec | 1:2 | -4.0 |
| 50 | 699.9 dB/sec | 85.7 dB/sec | 1:8 | -8.9 |

Table A.3: The observations made with test subject JBN for the 26th to the 50th iteration

A.2 GST

| Predicted optimal setting for GST | | | |
|-----------------------------------|---------------|---------------|-----|
| Iteration | Att | Rel | CR |
| 1 | 2000.0 dB/sec | 30.0 dB/sec | 1:8 |
| 2 | 2000.0 dB/sec | 2000.0 dB/sec | 1:4 |
| 3 | 30.0 dB/sec | 30.0 dB/sec | 1:2 |
| 4 | 30.0 dB/sec | 30.0 dB/sec | 1:2 |
| 5 | 30.0 dB/sec | 30.0 dB/sec | 1:2 |
| 6 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 |
| ⋮ | ⋮ | ⋮ | ⋮ |
| 50 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 |

Table A.4: The predicted optimal setting during the search for GST

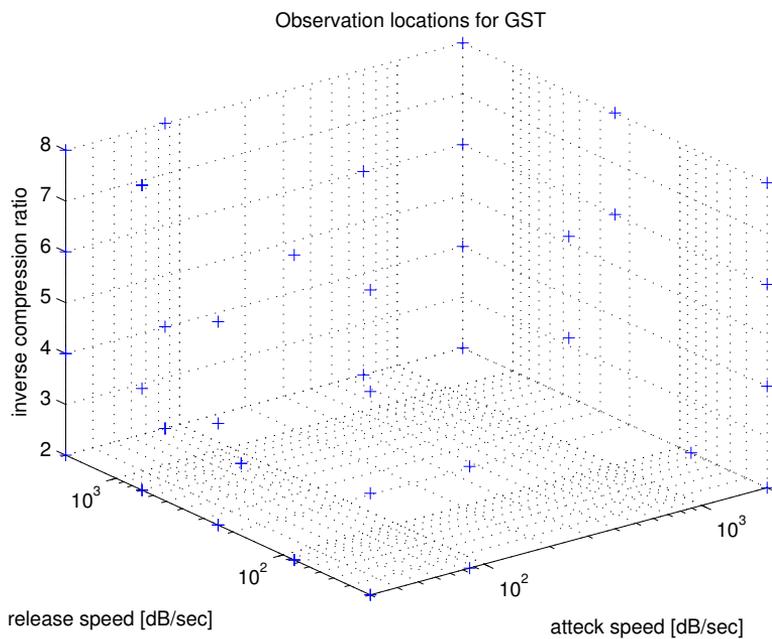


Figure A.2: Observation locations for test subject GST.

| Observations with GST | | | | |
|-----------------------|---------------|---------------|-----|-------------|
| Iteration no. | Att | Rel | CR | Scale value |
| 1 | 30.0 dB/sec | 244.9 dB/sec | 1:2 | -0.8 |
| 2 | 2000.0 dB/sec | 30.0 dB/sec | 1:8 | -9.0 |
| 3 | 2000.0 dB/sec | 2000.0 dB/sec | 1:4 | -9.2 |
| 4 | 30.0 dB/sec | 30.0 dB/sec | 1:2 | -0.5 |
| 5 | 2000.0 dB/sec | 30.0 dB/sec | 1:2 | -2.6 |
| 6 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 | -0.4 |
| 7 | 85.7 dB/sec | 2000.0 dB/sec | 1:2 | -1.2 |
| 8 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 | -1.7 |
| 9 | 30.0 dB/sec | 85.7 dB/sec | 1:8 | -3.7 |
| 10 | 85.7 dB/sec | 30.0 dB/sec | 1:4 | -7.5 |
| 11 | 85.7 dB/sec | 30.0 dB/sec | 1:2 | -4.1 |
| 12 | 2000.0 dB/sec | 2000.0 dB/sec | 1:2 | -8.3 |
| 13 | 30.0 dB/sec | 2000.0 dB/sec | 1:6 | -2.8 |
| 14 | 30.0 dB/sec | 2000.0 dB/sec | 1:4 | -0.8 |
| 15 | 2000.0 dB/sec | 2000.0 dB/sec | 1:8 | -9.8 |
| 16 | 30.0 dB/sec | 699.9 dB/sec | 1:2 | -0.2 |
| 17 | 30.0 dB/sec | 699.9 dB/sec | 1:2 | -0.2 |
| 18 | 30.0 dB/sec | 699.9 dB/sec | 1:2 | -0.5 |
| 19 | 30.0 dB/sec | 699.9 dB/sec | 1:4 | -1.6 |
| 20 | 30.0 dB/sec | 30.0 dB/sec | 1:6 | -4.2 |
| 21 | 30.0 dB/sec | 30.0 dB/sec | 1:2 | -4.1 |
| 22 | 2000.0 dB/sec | 30.0 dB/sec | 1:6 | -9.0 |
| 23 | 30.0 dB/sec | 244.9 dB/sec | 1:2 | -0.3 |
| 24 | 30.0 dB/sec | 244.9 dB/sec | 1:2 | -0.4 |
| 25 | 85.7 dB/sec | 2000.0 dB/sec | 1:4 | -5.8 |

Table A.5: The observations made with test subject GST for the 1st to the 25th iteration

| Observations with GST | | | | |
|-----------------------|---------------|---------------|-----|-------------|
| Iteration no. | Att | Rel | CR | Scale value |
| 26 | 2000.0 dB/sec | 30.0 dB/sec | 1:4 | -9.2 |
| 27 | 30.0 dB/sec | 30.0 dB/sec | 1:4 | -4.2 |
| 28 | 30.0 dB/sec | 30.0 dB/sec | 1:8 | -4.9 |
| 29 | 2000.0 dB/sec | 85.7 dB/sec | 1:2 | -6.2 |
| 30 | 30.0 dB/sec | 2000.0 dB/sec | 1:4 | -3.1 |
| 31 | 30.0 dB/sec | 85.7 dB/sec | 1:2 | -0.4 |
| 32 | 30.0 dB/sec | 85.7 dB/sec | 1:2 | -0.9 |
| 33 | 30.0 dB/sec | 85.7 dB/sec | 1:2 | -2.4 |
| 34 | 85.7 dB/sec | 2000.0 dB/sec | 1:8 | -8.4 |
| 35 | 30.0 dB/sec | 244.9 dB/sec | 1:6 | -3.7 |
| 36 | 244.9 dB/sec | 30.0 dB/sec | 1:8 | -9.4 |
| 37 | 2000.0 dB/sec | 244.9 dB/sec | 1:6 | -9.4 |
| 38 | 30.0 dB/sec | 699.9 dB/sec | 1:8 | -1.3 |
| 39 | 699.9 dB/sec | 2000.0 dB/sec | 1:2 | -8.5 |
| 40 | 30.0 dB/sec | 699.9 dB/sec | 1:8 | -0.6 |
| 41 | 30.0 dB/sec | 699.9 dB/sec | 1:8 | -1.0 |
| 42 | 699.9 dB/sec | 2000.0 dB/sec | 1:6 | -9.8 |
| 43 | 30.0 dB/sec | 244.9 dB/sec | 1:4 | -2.1 |
| 44 | 2000.0 dB/sec | 244.9 dB/sec | 1:8 | -9.9 |
| 45 | 85.7 dB/sec | 699.9 dB/sec | 1:2 | -1.6 |
| 46 | 85.7 dB/sec | 2000.0 dB/sec | 1:2 | -1.2 |
| 47 | 85.7 dB/sec | 2000.0 dB/sec | 1:2 | -0.5 |
| 48 | 244.9 dB/sec | 30.0 dB/sec | 1:6 | -8.6 |
| 49 | 2000.0 dB/sec | 2000.0 dB/sec | 1:6 | -9.8 |
| 50 | 85.7 dB/sec | 699.9 dB/sec | 1:2 | -0.1 |

Table A.6: The observations made with test subject GST for the 26th to the 50th iteration

A.3 OHA

| Predicted optimal setting for OHA | | | |
|-----------------------------------|---------------|---------------|-----|
| Iteration | Att | Rel | CR |
| 1 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 |
| ⋮ | ⋮ | ⋮ | ⋮ |
| 6 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 |
| 7 | 2000.0 dB/sec | 2000.0 dB/sec | 1:8 |
| 8 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 |
| ⋮ | ⋮ | ⋮ | ⋮ |
| 12 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 |
| 13 | 30.0 dB/sec | 30.0 dB/sec | 1:2 |
| 14 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 |
| ⋮ | ⋮ | ⋮ | ⋮ |
| 50 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 |

Table A.7: The predicted optimal setting during the search for OHA

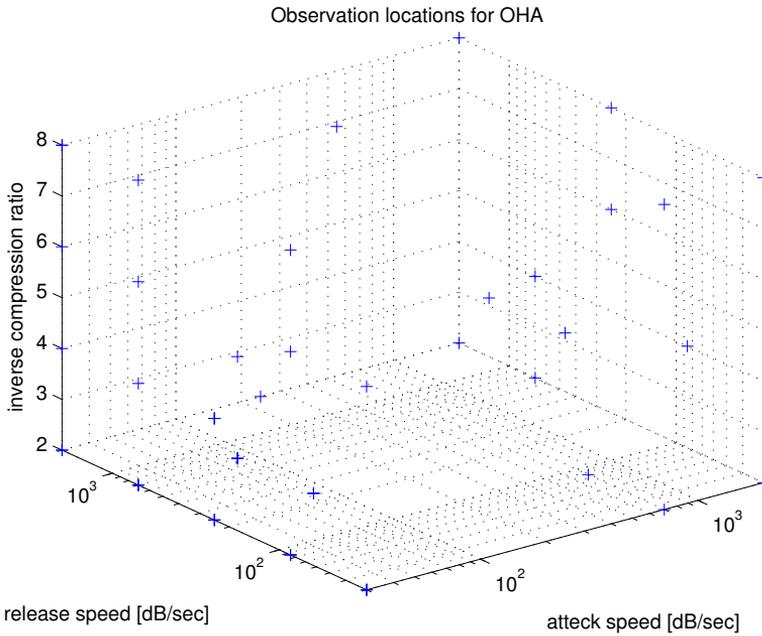


Figure A.3: Observation locations for test subject OHA.

| Observations with OHA | | | | |
|-----------------------|---------------|---------------|-----|-------------|
| Iteration no. | Att | Rel | CR | Scale value |
| 1 | 244.9 dB/sec | 85.7 dB/sec | 1:6 | -6.8 |
| 2 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 | -3.0 |
| 3 | 244.9 dB/sec | 2000.0 dB/sec | 1:2 | -4.1 |
| 4 | 30.0 dB/sec | 30.0 dB/sec | 1:2 | -5.7 |
| 5 | 2000.0 dB/sec | 2000.0 dB/sec | 1:2 | -4.4 |
| 6 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 | -3.3 |
| 7 | 30.0 dB/sec | 30.0 dB/sec | 1:6 | -7.6 |
| 8 | 2000.0 dB/sec | 2000.0 dB/sec | 1:8 | -8.7 |
| 9 | 2000.0 dB/sec | 699.9 dB/sec | 1:4 | -7.4 |
| 10 | 2000.0 dB/sec | 30.0 dB/sec | 1:2 | -2.3 |
| 11 | 30.0 dB/sec | 85.7 dB/sec | 1:2 | -2.2 |
| 12 | 30.0 dB/sec | 30.0 dB/sec | 1:2 | -2.9 |
| 13 | 30.0 dB/sec | 30.0 dB/sec | 1:2 | -3.0 |
| 14 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 | -1.6 |
| 15 | 30.0 dB/sec | 85.7 dB/sec | 1:2 | -0.9 |
| 16 | 30.0 dB/sec | 85.7 dB/sec | 1:2 | -0.7 |
| 17 | 30.0 dB/sec | 244.9 dB/sec | 1:2 | -2.4 |
| 18 | 30.0 dB/sec | 244.9 dB/sec | 1:2 | -1.2 |
| 19 | 85.7 dB/sec | 244.9 dB/sec | 1:2 | -1.6 |
| 20 | 30.0 dB/sec | 699.9 dB/sec | 1:2 | -0.8 |
| 21 | 30.0 dB/sec | 699.9 dB/sec | 1:2 | -3.3 |
| 22 | 30.0 dB/sec | 699.9 dB/sec | 1:2 | -1.5 |
| 23 | 85.7 dB/sec | 244.9 dB/sec | 1:2 | -1.4 |
| 24 | 30.0 dB/sec | 244.9 dB/sec | 1:2 | -2.8 |
| 25 | 699.9 dB/sec | 85.7 dB/sec | 1:2 | -7.4 |

Table A.8: The observations made with test subject OHA for the 1st to the 25th iteration

| Observations with OHA | | | | |
|-----------------------|---------------|---------------|-----|-------------|
| Iteration no. | Att | Rel | CR | Scale value |
| 26 | 30.0 dB/sec | 85.7 dB/sec | 1:8 | -6.7 |
| 27 | 30.0 dB/sec | 699.9 dB/sec | 1:6 | -3.6 |
| 28 | 30.0 dB/sec | 2000.0 dB/sec | 1:4 | -3.2 |
| 29 | 30.0 dB/sec | 244.9 dB/sec | 1:4 | -2.4 |
| 30 | 30.0 dB/sec | 244.9 dB/sec | 1:4 | -4.3 |
| 31 | 30.0 dB/sec | 2000.0 dB/sec | 1:6 | -4.5 |
| 32 | 2000.0 dB/sec | 30.0 dB/sec | 1:8 | -8.7 |
| 33 | 85.7 dB/sec | 699.9 dB/sec | 1:2 | -3.1 |
| 34 | 85.7 dB/sec | 699.9 dB/sec | 1:2 | -2.9 |
| 35 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 | -7.1 |
| 36 | 85.7 dB/sec | 699.9 dB/sec | 1:2 | -5.2 |
| 37 | 30.0 dB/sec | 699.9 dB/sec | 1:4 | -2.0 |
| 38 | 30.0 dB/sec | 244.9 dB/sec | 1:4 | -3.0 |
| 39 | 85.7 dB/sec | 244.9 dB/sec | 1:2 | -4.3 |
| 40 | 699.9 dB/sec | 30.0 dB/sec | 1:2 | -6.9 |
| 41 | 85.7 dB/sec | 699.9 dB/sec | 1:4 | -9.0 |
| 42 | 2000.0 dB/sec | 244.9 dB/sec | 1:6 | -9.3 |
| 43 | 2000.0 dB/sec | 85.7 dB/sec | 1:4 | -8.4 |
| 44 | 699.9 dB/sec | 30.0 dB/sec | 1:8 | -9.5 |
| 45 | 30.0 dB/sec | 85.7 dB/sec | 1:6 | -5.8 |
| 46 | 244.9 dB/sec | 30.0 dB/sec | 1:6 | -8.3 |
| 47 | 30.0 dB/sec | 699.9 dB/sec | 1:8 | -1.0 |
| 48 | 2000.0 dB/sec | 699.9 dB/sec | 1:2 | -7.8 |
| 49 | 2000.0 dB/sec | 244.9 dB/sec | 1:8 | -9.7 |
| 50 | 244.9 dB/sec | 699.9 dB/sec | 1:8 | -9.7 |

Table A.9: The observations made with test subject OHA for the 26th to the 50th iteration

A.4 AWE

| Predicted optimal setting for AWE | | | |
|-----------------------------------|---------------|---------------|-----|
| Iteration | Att | Rel | CR |
| 1 | 2000.0 dB/sec | 30.0 dB/sec | 1:2 |
| 2 | 2000.0 dB/sec | 30.0 dB/sec | 1:6 |
| 3 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 |
| 4 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 |
| 5 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 |
| 6 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 |
| ⋮ | ⋮ | ⋮ | ⋮ |
| 10 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 |
| 11 | 30.0 dB/sec | 2000.0 dB/sec | 1:6 |
| 12 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 |
| 13 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 |
| 14 | 30.0 dB/sec | 2000.0 dB/sec | 1:6 |
| ⋮ | ⋮ | ⋮ | ⋮ |
| 26 | 30.0 dB/sec | 2000.0 dB/sec | 1:6 |
| 27 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 |
| 28 | 30.0 dB/sec | 2000.0 dB/sec | 1:6 |
| ⋮ | ⋮ | ⋮ | ⋮ |
| 32 | 30.0 dB/sec | 2000.0 dB/sec | 1:6 |
| 33 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 |
| 34 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 |
| 35 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 |
| 36 | 30.0 dB/sec | 2000.0 dB/sec | 1:6 |
| 37 | 30.0 dB/sec | 2000.0 dB/sec | 1:6 |
| 38 | 30.0 dB/sec | 2000.0 dB/sec | 1:6 |
| 39 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 |
| 40 | 30.0 dB/sec | 2000.0 dB/sec | 1:6 |
| ⋮ | ⋮ | ⋮ | ⋮ |
| 50 | 30.0 dB/sec | 2000.0 dB/sec | 1:6 |

Table A.10: The predicted optimal setting during the search for AWE

| Observations with AWE | | | | |
|-----------------------|---------------|---------------|-----|-------------|
| Iteration no. | Att | Rel | CR | Scale value |
| 1 | 85.7 dB/sec | 2000.0 dB/sec | 1:8 | -2.7 |
| 2 | 2000.0 dB/sec | 30.0 dB/sec | 1:2 | -6.1 |
| 3 | 2000.0 dB/sec | 30.0 dB/sec | 1:6 | -7.8 |
| 4 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 | -2.0 |
| 5 | 30.0 dB/sec | 2000.0 dB/sec | 1:4 | -2.6 |
| 6 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 | -4.5 |
| 7 | 30.0 dB/sec | 2000.0 dB/sec | 1:6 | -2.7 |
| 8 | 30.0 dB/sec | 30.0 dB/sec | 1:8 | -8.8 |
| 9 | 699.9 dB/sec | 2000.0 dB/sec | 1:2 | -8.6 |
| 10 | 2000.0 dB/sec | 699.9 dB/sec | 1:8 | -9.5 |
| 11 | 30.0 dB/sec | 30.0 dB/sec | 1:4 | -5.3 |
| 12 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 | -1.2 |
| 13 | 30.0 dB/sec | 30.0 dB/sec | 1:2 | -6.4 |
| 14 | 2000.0 dB/sec | 244.9 dB/sec | 1:4 | -9.0 |
| 15 | 30.0 dB/sec | 2000.0 dB/sec | 1:8 | -3.1 |
| 16 | 30.0 dB/sec | 2000.0 dB/sec | 1:4 | -3.2 |
| 17 | 85.7 dB/sec | 2000.0 dB/sec | 1:8 | -8.7 |
| 18 | 30.0 dB/sec | 2000.0 dB/sec | 1:4 | -3.3 |
| 19 | 30.0 dB/sec | 699.9 dB/sec | 1:8 | -1.4 |
| 20 | 30.0 dB/sec | 699.9 dB/sec | 1:4 | -1.4 |
| 21 | 30.0 dB/sec | 244.9 dB/sec | 1:4 | -1.1 |
| 22 | 30.0 dB/sec | 699.9 dB/sec | 1:4 | -6.1 |
| 23 | 30.0 dB/sec | 244.9 dB/sec | 1:6 | -5.6 |
| 24 | 30.0 dB/sec | 699.9 dB/sec | 1:4 | -3.2 |
| 25 | 30.0 dB/sec | 244.9 dB/sec | 1:4 | -3.3 |

Table A.11: The observations made with test subject AWE for the 1st to the 25th iteration

| Observations with AWE | | | | |
|-----------------------|---------------|---------------|-----|-------------|
| Iteration no. | Att | Rel | CR | Scale value |
| 26 | 30.0 dB/sec | 244.9 dB/sec | 1:4 | -6.8 |
| 27 | 30.0 dB/sec | 244.9 dB/sec | 1:8 | -0.6 |
| 28 | 30.0 dB/sec | 699.9 dB/sec | 1:8 | -7.9 |
| 29 | 30.0 dB/sec | 2000.0 dB/sec | 1:2 | -5.9 |
| 30 | 30.0 dB/sec | 699.9 dB/sec | 1:8 | -4.3 |
| 31 | 2000.0 dB/sec | 30.0 dB/sec | 1:8 | -9.5 |
| 32 | 30.0 dB/sec | 244.9 dB/sec | 1:8 | -1.9 |
| 33 | 30.0 dB/sec | 244.9 dB/sec | 1:8 | -3.3 |
| 34 | 30.0 dB/sec | 2000.0 dB/sec | 1:6 | -5.6 |
| 35 | 244.9 dB/sec | 30.0 dB/sec | 1:2 | -7.5 |
| 36 | 30.0 dB/sec | 699.9 dB/sec | 1:6 | -1.1 |
| 37 | 30.0 dB/sec | 699.9 dB/sec | 1:6 | -3.8 |
| 38 | 30.0 dB/sec | 699.9 dB/sec | 1:6 | -2.6 |
| 39 | 30.0 dB/sec | 244.9 dB/sec | 1:6 | -5.9 |
| 40 | 30.0 dB/sec | 2000.0 dB/sec | 1:6 | -1.3 |
| 41 | 30.0 dB/sec | 244.9 dB/sec | 1:6 | -0.8 |
| 42 | 30.0 dB/sec | 85.7 dB/sec | 1:4 | -6.3 |
| 43 | 85.7 dB/sec | 244.9 dB/sec | 1:8 | -8.1 |
| 44 | 30.0 dB/sec | 85.7 dB/sec | 1:6 | -6.6 |
| 45 | 30.0 dB/sec | 244.9 dB/sec | 1:2 | -1.7 |
| 46 | 30.0 dB/sec | 244.9 dB/sec | 1:2 | -1.0 |
| 47 | 30.0 dB/sec | 699.9 dB/sec | 1:2 | -4.7 |
| 48 | 30.0 dB/sec | 244.9 dB/sec | 1:2 | -6.7 |
| 49 | 30.0 dB/sec | 699.9 dB/sec | 1:2 | -0.3 |
| 50 | 30.0 dB/sec | 699.9 dB/sec | 1:2 | -5.8 |

Table A.12: The observations made with test subject AWE for the 26th to the 50th iteration

Bibliography

- Ulf Andersson and Björn Lyxell. Phonological deterioration in adults with an acquired severe hearing impairment: A deterioration in long-term memory or working memory? Scandinavia Audiology, 28:241–247, october 1999.
- ANSI/ASA S3.22-2009. Specification of hearing aid characteristics. Standard.
- Deniz Başkent, Cheryl L. Eiler, and Brent Edwards. Using genetic algorithms with subjective input from human subjects: Implications for fitting hearing aids and cochlear implants. Ear & Hearing, 28(3):371–380, June 2007.
- Alan Baddeley. Working memory. Science, New Series, 255(5044):556–559, january 1992.
- S Bech and N Zacharov. Perceptual Audio Evaluation - Theory, Method and Application. Wiley, Chichester, 2007.
- Adriana Birlutiu, Perry Groot, and Tom Heskes. Multi-task preference learning with an application to hearing aid personalization. Neurocomput., 73(7-9): 1177–1185, 2010. ISSN 0925-2312.
- Christopher M. Bishop. Pattern Recognition and Machine Learning. Springer Science+Business Media, New York, 2006. ISBN 0-387-31073-8.
- Craig Boutilier. A pomdp formulation of preference elicitation problems. In Eighteenth national conference on Artificial intelligence, pages 239–246, Menlo Park, CA, USA, 2002. American Association for Artificial Intelligence. ISBN 0-262-51129-0.
- Wei Chu and Zoubin Ghahramani. Extension of gaussian processes for ranking: Semi-supervised and active learning. In Proceedings of the NIPS 2005

- Workshop on Learning to Rank, pages 29–35. Neural Information Processing Systems Foundation, 2005.
- Harvey Dillon. Hearing Aids. Boomerang Press, Sydney & Thieme, New York & Stuttgart, 2001. ISBN 1-58890-052-5.
- Eric A. Durant, Gregory H. Wakefield, Dianne J. Van Tasell, and Martin E. Rickert. Efficient perceptual tuning of hearing aids with genetic algorithms. IEEE Transactions on Speech and Audio Processing, 12(2):144–155, March 2004. ISSN 1063-6676. URL <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1284342>.
- Perry P. C. Groot, Tom Heskes, Tjeerd M. H. Dijkstra, and James M. Kates. Predicting preference judgments of individual normal and hearing-impaired listeners with gaussian processes. IEEE Transactions on Audio, Sound, and Language Processing, 2010.
- Tom Heskes and Bert de Vries. Incremental utility elicitation for adaptive personalization. In Proceedings of the 17th Belgium-Netherlands Conference on Artificial Intelligence, BNAIC; 17, pages 127–134. Information Retrieval and Information Systems, S.I. : s.n., march 2005.
- ITU-T P.835. Subjective test methodology for evaluating speech communication systems that include noise suppression algorithm, november 2003. Recommendation.
- ITU-T P.862. Perceptual evaluation of speech quality (pesq): An objective method for end-to-end speech quality assessment of narrow-band telephone network and speech codecs, february 2001. Recommendation.
- David John Cameron Mackay. Bayesian methods for adaptive models. PhD thesis, Pasadena, CA, USA, 1992.
- Carl E. Rasmussen and Christopher K. I. Williams. Gaussian Processes for Machine Learning. The MIT Press, 2006. ISBN ISBN 0-262-18253-X.
- Todd A. Ricketts and Benjamin W. Y. Hornsby. Sound quality measures for speech in noise through a commercial hearing aid implementation “digital noise reduction”. Journal of the American Academy of Audiology, 16(5): 270–277, 2005.
- Sambu Seo, Marko Wallat, Thore Graepel, and Klaus Obermayer. Gaussian process regression: Active data selection and test point rejection. In In Proceedings of the International Joint Conference on Neural Networks (IJCNN), pages 241–246. IEEE, 2000.
- F. Wickelmaier and C. Schmid. A Matlab function to estimate choice model parameters from paired-comparison data. Behavior Research Methods, Instruments, & Computers, 36:29–40, 2004.