

Abstract

Ongoing title:

Probabilistic user modelling methods for improving human-in-the-loop machine learning

Interactive user modelling for human-in-the-loop machine learning

Abstract

En puhu suomea.

Preface

To fill.

Espoo, October 7, 2019,

Pedram Daee

Contents

Preface	3
Contents	5
List of Publications	7
Author's Contribution	9
1. Introduction	11
1.1 Motivation	11
1.2 Research questions and contributions	11
1.3 Organization of the thesis	13
2. Probabilistic modelling of data and user	15
2.1 Preliminaries	15
2.2 Modelling data for prediction	16
2.2.1 Bayesian linear regression	17
2.3 Modelling user interaction for prediction	21
2.3.1 User feedback as observation about model parameters	21
2.3.2 User feedback as outcome of a cognitive process . . .	23
2.3.3 User feedback for intent modelling	25
2.4 Posterior inference	29
2.5 Key components of the joint model	30
3. User interaction with the probabilistic model	31
3.1 Active learning and experimental design	31
3.2 Multi-armed bandits and Bayesian optimization	31
4. Summary of the Contributions	33
4.1 Interactive intent modelling from multiple feedback domains (Publications I and V)	33
4.2 Expert knowledge elicitation for high-dimensional prediction (Publications II and III)	33

Contents

4.3	User modelling for avoiding overfitting in knowledge elicitation (Publication IV)	33
5.	Discussion	35
	References	37
	Publications	41

List of Publications

This thesis consists of an overview and of the following publications which are referred to in the text by their Roman numerals.

- I** Pedram Daei, Joel Pyykkö, Dorota Glowacka, and Samuel Kaski. Interactive Intent Modeling from Multiple Feedback Domains. In *Proceedings of the 21st International Conference on Intelligent User Interfaces*, Sonoma, California, USA, 71–75, March 2016.
- II** Pedram Daei^{*}, Tomi Peltola^{*}, Marta Soare^{*}, and Samuel Kaski. Knowledge elicitation via sequential probabilistic inference for high-dimensional prediction. *Machine Learning*, 106, 9-10, 1599–1620, 2017.
- III** Iris Sundin^{*}, Tomi Peltola^{*}, Luana Micallef, Homayun Afrabandpey, Marta Soare, Muntasir Mamun Majumder, Pedram Daei, Chen He, Baris Serim, Aki Havulinna, Caroline Heckman, Giulio Jacucci, Pekka Marttinen, and Samuel Kaski. Improving genomics-based predictions for precision medicine through active elicitation of expert knowledge. *Bioinformatics*, 34, 13, i395–i403, 2018.
- IV** Pedram Daei^{*}, Tomi Peltola^{*}, Aki Vehtari, and Samuel Kaski. User Modelling for Avoiding Overfitting in Interactive Knowledge Elicitation for Prediction. In *Proceedings of the 23rd International Conference on Intelligent User Interfaces*, Tokyo, Japan, 305–310, March 2018.
- V** Giulio Jacucci, Oswald Barral, Pedram Daei, Markus Wenzel, Baris Serim, Tuukka Ruotsalo, Patrik Pluchino, Jonathan Freeman, Luciano Gamberini, Samuel Kaski, Benjamin Blankertz. Integrating Neurophysiological Relevance Feedback in Intent Modeling for Information Retrieval. *Journal of the Association for Information Science and Technology*, 2019.

Author's Contribution

Publication I: “Interactive Intent Modeling from Multiple Feedback Domains”

The author had the main responsibility in problem formulation and modeling. The author designed and implemented the simulation experiment. Joel Pyykkö and the author built the system for user studies and conducted them together. The author wrote the initial draft of the manuscript, after which all co-authors joined for revisions.

Publication II: “Knowledge elicitation via sequential probabilistic inference for high-dimensional prediction”

The ideas and experiments in this article were designed jointly (the first three authors contributed equally). The author had the main responsibility in the derivation of the sequential experimental design and implementation of the experiments. Dr. Tomi Peltola derived and implemented the posterior approximation. The manuscript was written jointly.

Publication III: “Improving genomics-based predictions for precision medicine through active elicitation of expert knowledge”

The author contributed on formulation of the sequential experimental design and implementation of a portion of the early version of the experiments. The author made comments to the manuscript in preparation.

Publication IV: “User Modelling for Avoiding Overfitting in Interactive Knowledge Elicitation for Prediction”

The ideas and experiments in this article were designed jointly (the first two authors contributed equally). The author designed and implemented the user study. Dr. Tomi Peltola had the main responsibility of the model formulation. The first two authors wrote the initial draft of the manuscript, after which all co-authors joined for revisions.

Publication V: “Integrating Neurophysiological Relevance Feedback in Intent Modeling for Information Retrieval”

The author had the main responsibility in design and implementation of the interactive intent modelling and information retrieval system, and writing of the corresponding sections. All the authors contributed to paper revisions.

1. Introduction

Whether it is an everyday user searching for an application in her mobile phone or a doctor working with a cancer diagnostic system, humans and machines are increasingly interacting with each other [3] [more strong refs]. The goal of the thesis is to improve this interaction by incorporating a probabilistic model of the human user in the system they are interacting with. In particular, the thesis considers the family of problems where the human and machine interact to solve a prediction problem. Such problems can include personalized search activity or medical prediction about the response of a cancer drug. An important common factor in both these scenarios is that the number of labeled data (training data) that the machine can use to make predictions, is usually very few compared to the dimension of search space. This results in ill-posed statistical learning since there are limits in how low in sample size statistical methods can go [12].

User modelling in human–computer interaction aims at improving the usability and usefulness of collaborative human–computer systems and providing personalised user experiences [14]. Machine learning based interactive systems extend user modelling to encompass statistical models interpreting user’s actions.

1.1 Motivation

1.2 Research questions and contributions

This thesis investigates methods to tackle the limited user interaction challenge in interactive machine learning for prediction. The thesis focuses on scenarios where there is few labeled data available compared to the dimension of the problem, or when a human user is provider of the labeled data. The core idea of the thesis is to jointly model the human user with the data as part of a unified probabilistic model and use the model to improve the interaction.

RQ1 – *Can we exploit new sources of interaction as additional learning signals from human user to improve interactive intent modelling?*

Publications I and V contribute to this research question by proposing models to incorporate new types of user feedback to amend the limited feedback in exploratory information seeking tasks. The tasks considered are document search scenarios where a user needs to sequentially provide relevance feedback to suggested keywords in order to find the targeted document. This is modelled as a multi-armed bandit problem with the goal of finding the most relevant document with minimum interaction. In particular, Publication I couples user relevance feedback on both documents and keywords by assuming a shared underlying latent model connected through a probabilistic model of the relationship between keywords and documents. Thompson sampling on the posterior of the latent intent was then used to recommend new documents and keywords in each iteration. Publication V investigates the use of implicit relevance feedback from neurophysiology signals for effortless information seeking. The work contributes by demonstrating how to integrate this inherently noisy and implicit feedback source with scarce explicit interaction. A model for controlling the accuracy of the feedback given its nature (implicit or explicit) was introduced. Similar to Publication I, Thompson sampling was used to control the exploration and exploitation balance of the recommendations. Both publications were evaluated by user studies in realistic information seeking tasks.

RQ2 – *Can expert knowledge about high dimensional data models be elicited to improve the prediction performance?*

Publications II and III contribute to this research question. Publication II proposes a framework for user knowledge elicitation as a probabilistic inference process, where the user knowledge is sequentially queried to improve predictions. In particular, sparse linear regression is considered as the data model with access to only few high-dimensional training data. It is assumed that there are experts who have knowledge about the relevance of the covariates, or of values of the regression coefficients and can provide this information to the data model if queried. The work contributes by an algorithm and computational approximation for fast and efficient interaction, which sequentially identifies the most informative queries to ask from the user. Publication III, builds on Publication II by adding user knowledge about direction of relevance of covariates and applying the method in important applications of precision medicine with the goal of predicting the effects of different treatments using high-dimensional genomic measurements. Both publications were evaluated by extensive simulations and user studies. Source codes for methods presented in Publications II and III, and user study data from Publication II are available at <https://github.com/HIIT/knowledge-elicitation-for-linear-regression> and <https://github.com/AaltoPML/knowledge-elicitation-for-precision-medicine>.

RQ3 – *Is it enough to incorporate human knowledge directly in the data model as explained in RQ2, or could it be beneficial to account for rational knowledge updates that humans may undergo during the interaction?*

Publication IV contributes to this research question by modelling the knowledge provider, here the human user, as a rational agent that updates its knowledge about the underlying prediction task during the interaction. In particular, certain aspects of training data may be revealed to the user during knowledge elicitation. The design of the system is then critical, since the elicited user knowledge cannot be assumed to be independent from the data model knowledge coming from the training data. If not accounted properly, knowledge elicitation can lead to double use of data and overfitting, if the user reinforces noisy patterns in the data. We propose a user modelling methodology, by assuming simple rational behaviour, to correct the problem and evaluate the method in a user study. Source code and user study data are available at <https://github.com/HIIT/human-overfitting-in-IML>.

1.3 Organization of the thesis

The organization of the thesis is as follows. Chapter 2 provides an overview of probabilistic modelling and introduces our approach of modelling the user and data as a joint probabilistic model. Chapter three investigates the purpose of interactions and reviews different utility functions designed to minimize the effort of the user. The fourth chapter summarizes Publications I-IV. Chapter five concludes the thesis and provides discussions for future works.

2. Probabilistic modelling of data and user

This chapter provides a brief introduction to probabilistic modelling as the main statistical framework that is used through the thesis. After some preliminaries, Section 2.2 briefly reviews the type of linear models that are used in Publication I-IV for prediction. Section 2.3 introduces different types of user interaction with the linear model and explains how user knowledge about the model can be incorporated as observational feedback. The computational solutions for Bayesian inference are reviewed in 2.4. Finally, the key components for modelling observational data and user feedback as a joint probabilistic model is introduced in Section 2.5.

2.1 Preliminaries

The core idea of probabilistic modelling is to describe all the unobserved parameters and observed data as random variables from probability distributions. The unobserved parameters include the unknown quantity of interest or other parameters that affect the data or the quantity of interest. Bayesian inference provides a powerful framework to fit the described probabilistic model to observational data [16]. A core feature of Bayesian inference is that it provides probability distributions as the solution, compared to deterministic methods which provide a single outcome. This uncertainty quantification is of particularly high interest in cases where few observational data are available or when the data acquisition scheme is controlled by the model. Both of these constraints are prominently present in the tasks investigated by this thesis.

We follow the notation of [16] and use $p(\cdot)$ to denote a probability distribution and $p(\cdot | \cdot)$ a conditional distribution. Consider the case where there are a set of observations $\mathcal{D} = \{y_1, \dots, y_n\}$ generated from a probabilistic model with an unobserved parameter of interest θ . The observational model for an observation y describes the conditional density of y given the parameter θ and is denoted as $y \sim p(y | \theta)$. It is usually assumed that the observations are conditionally independent given θ , enabling us to write the model for all observations as $p(\mathcal{D} | \theta) = \prod_{i=1}^n p(y_i | \theta)$. The observational model is called likelihood function if

perceived as a function of θ with fixed observation y . One of the core questions in statistical inference is to estimate the parameter of interest θ based on observations \mathcal{D} . Bayesian inference answers this question by computing the conditional distribution of θ given \mathcal{D} following the Bayes rule

$$p(\theta | \mathcal{D}) = \frac{p(\mathcal{D} | \theta)p(\theta)}{p(\mathcal{D})}. \quad (2.1)$$

Where $p(\theta)$ represents the prior belief about θ and $p(\mathcal{D})$ is called the marginal likelihood and acts as a normalization factor as it does not depend on θ . The marginal likelihood can be computed using the marginalization rule, i.e., $p(\mathcal{D}) = \int_{\theta} p(\mathcal{D}, \theta) d\theta = \int_{\theta} p(\mathcal{D} | \theta)p(\theta) d\theta$ ¹. $p(\theta | \mathcal{D})$ is called the posterior and it expresses the uncertainty surrounding the true value of θ , after updating the prior assumptions about θ (i.e., the prior distribution) with the knowledge coming from the observations through the likelihood.

In many cases, we may be more interested to make a prediction about an unknown observable data point \hat{y} rather than the parameter θ . Bayesian inference allows us to compute the conditional distribution of this unknown observable data given the observed data point as

$$p(\hat{y} | \mathcal{D}) = \int_{\theta} p(\hat{y} | \theta)p(\theta | \mathcal{D}) d\theta. \quad (2.2)$$

$p(\hat{y} | \mathcal{D})$ is known as the posterior predictive distribution and represents the uncertainty about a potential new observation. An example usage of this distribution could be when we want to predict the value of a test data. Since test data is unobserved, we can use posterior predictive distribution as our best guess. However, in many applications, only a value (and not a distribution) is required as the prediction. This can be handled by using some statistics of the distribution (for example mean or median) as the prediction. Still, the quantified uncertainty in the distribution can be useful as it provides knowledge about how certain we are about our estimate. Particularly, this uncertainty can help us to design more efficient interaction with a user, as will be discussed in Section 3.

2.2 Modelling data for prediction

Prediction is one of the core problems in statistical analysis and supervised machine learning. Given a set of n input and output pairs, called training data, denoted by $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$, the goal is to find a mapping from inputs to outputs. Here, $\mathbf{x}_i = [x_i^1, \dots, x_i^d]^T \in \mathbb{R}^d$ is a d -dimensional column vector representing the values of i^{th} data point². The dimensions are commonly called feature, covariates, or attribute. y_i is the corresponding response (or target) variable which can be anything depending on the underlying problem. In this thesis, we consider

¹The integral turns to summation for discrete θ .

²We use subscripts to index an specific item (e.g, one particular observation out of several) and superscripts to refer to dimensions of the variable.

the regression tasks, meaning that we model response variables by real values, i.e., $y_i \in \mathbb{R}$. The problem is called classification in supervised learning if the response variable is restricted to categorical values (called classes).

2.2.1 Bayesian linear regression

A well-studied and widely practical type of regression, known as linear regression, assumes that the relationship between all inputs and their corresponding response variables is linear. This relationship can be described as

$$y_i = \sum_{j=1}^d x_i^j w^j + \epsilon_i = \mathbf{x}_i^\top \mathbf{w} + \epsilon_i, \quad i = 1, \dots, n,$$

where $\mathbf{w} \in \mathbb{R}^d$ is the regression coefficients or the model's weights and ϵ_i is the residual error between linear prediction $\mathbf{x}_i^\top \mathbf{w}$ and the response value y_i . Given the labeled data \mathcal{D} , a commonly used error function is the sum of squared of residual errors $\sum_{i=1}^n (\mathbf{x}_i^\top \mathbf{w} - y_i)^2$. By stacking the inputs in $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n]^\top \in \mathbb{R}^{n \times d}$ and outputs in $\mathbf{y} = [y_1, \dots, y_n]^\top \in \mathbb{R}^n$ the error can be expressed in vector format as $(\mathbf{X}\mathbf{w} - \mathbf{y})^\top (\mathbf{X}\mathbf{w} - \mathbf{y})$. The frequentist solution directly finds the point estimate for \mathbf{w} that minimizes this error (also known as least squares solution). It is straightforward to show that $\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$ would be the point estimate solution given that $(\mathbf{X}^\top \mathbf{X})^{-1}$ is invertible.

However, as mentioned, we are interested in directly quantifying the uncertainty of the solution. The probabilistic way to model this problem is to explicitly describe the model assumptions (likelihood and priors) as probability distributions. In linear regression, the residual errors ϵ_i and the weights \mathbf{w} are modelled as random variables and the inputs \mathbf{x}_i as vector of values that are given. A customary assumption is to model the residual errors as independent zero-mean Gaussian random variables $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$, where σ^2 is variance of the Gaussian distribution which indicates the model tolerance about residual errors. It is common to also model σ^2 as another random variable with its own distribution assumption, however, we consider it to be a fixed hyperparameter for simplicity. The quantity of interest in the linear model is the regression coefficients. To complete the Bayesian inference loop, we need to consider a prior distribution on \mathbf{w} . There are many ways to do this depending on the underlying task. One simple prior could be to assume that coefficients are independent and each come from a zero-mean Gaussian distribution, encouraging the weights to be close to zero. This simple Bayesian linear regression can be described by

$$\begin{aligned} \mathbf{y} &\sim \mathcal{N}(\mathbf{X}\mathbf{w}, \sigma^2 \mathbf{I}), \\ \mathbf{w} &\sim \mathcal{N}(0, \tau^2 \mathbf{I}), \end{aligned} \tag{2.3}$$

where τ^2 is the variance of the weights and \mathbf{I} is the identity matrix. For simplicity we assume that τ^2 is also a fixed hyperparameter. Therefore, The only

unobserved parameters of this model is \mathbf{w} . The Bayesian rule allows us to derive the posterior for \mathbf{w} as³

$$p(\mathbf{w} | \mathcal{D}) = \frac{p(\mathcal{D} | \mathbf{w})p(\mathbf{w})}{p(\mathcal{D})} = \frac{N(\mathbf{y} | \mathbf{X}\mathbf{w}, \sigma^2 \mathbf{I})N(\mathbf{w} | 0, \tau^2 \mathbf{I})}{p(\mathcal{D})}. \quad (2.4)$$

Generally, the posterior in many problems cannot be analytically derived (we will discuss this issue more in Section 2.4). For this simple model, however, an analytical solution is available. We will go through the steps as an exercise with posterior inference. Before starting, it would be useful to note that a multivariate Gaussian distribution can be expressed as

$$\begin{aligned} N(\mathbf{w} | \mu, \Sigma) &\propto \exp(\mathbf{w} - \mu)^\top \Sigma^{-1} (\mathbf{w} - \mu) \\ &\propto \exp(\mathbf{w}^\top \Sigma^{-1} \mathbf{w} - 2\mathbf{w}^\top \Sigma^{-1} \mu), \end{aligned} \quad (2.5)$$

after dropping all the terms that are constant with respect to \mathbf{w} .

To derive the posterior, we follow Equation 2.4

$$\begin{aligned} p(\mathbf{w} | \mathcal{D}) &\propto \exp\left(\frac{1}{2\sigma^2}(\mathbf{y} - \mathbf{X}\mathbf{w})^\top (\mathbf{y} - \mathbf{X}\mathbf{w})\right) \exp\left(\frac{1}{2\tau^2} \mathbf{w}^\top \mathbf{w}\right) \\ &\propto \exp\left(\sigma^{-2}(\mathbf{y}^\top \mathbf{y} - 2\mathbf{w}^\top \mathbf{X}^\top \mathbf{y} + \mathbf{w}^\top \mathbf{X}^\top \mathbf{X} \mathbf{w}) + \tau^{-2} \mathbf{w}^\top \mathbf{w}\right) \\ &\propto \exp\left(-2\sigma^{-2} \mathbf{w}^\top \mathbf{X}^\top \mathbf{y} + \mathbf{w}^\top (\sigma^{-2} \mathbf{X}^\top \mathbf{X} + \tau^{-2} \mathbf{I}) \mathbf{w}\right) \end{aligned} \quad (2.6)$$

where we dropped $p(\mathcal{D})$ and all the other terms which were constant with respect to \mathbf{w} . Equation 2.6 has the same form to Equation 2.5 (with respect to \mathbf{w}) and therefore is proportional to a multivariate Gaussian distribution. This relation shows that the posterior should be multivariate Gaussian, as both equations should integrate to one, and thus its parameters can be found by matching the terms of the two equations as $\Sigma^{-1} = (\sigma^{-2} \mathbf{X}^\top \mathbf{X} + \tau^{-2} \mathbf{I})$ and $\Sigma^{-1} \mu = \sigma^{-2} \mathbf{X}^\top \mathbf{y}$. Finally, the posterior of our simple linear regression can be described as

$$\begin{aligned} p(\mathbf{w} | \mathcal{D}) &= N(\mathbf{w} | \mu, \Sigma), \text{ where} \\ \Sigma^{-1} &= (\sigma^{-2} \mathbf{X}^\top \mathbf{X} + \tau^{-2} \mathbf{I}), \\ \mu &= \Sigma \sigma^{-2} \mathbf{X}^\top \mathbf{y}. \end{aligned} \quad (2.7)$$

This simple Bayesian regression is used as the underlying data model in Publication I. It would be interesting to compare a point estimate of the posterior distribution with the frequentist solution $\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$. Maximum a posteriori (MAP) is a point estimate which represents the value in the posterior that has

³Note that we use $\mathbf{w} \sim N(0, \tau^2 \mathbf{I})$ to denote the random variable \mathbf{w} and $N(\mathbf{w} | 0, \tau^2 \mathbf{I})$ to refer to the density function.

the highest density. In our case, this would be equal to the posterior mean, which after some rearrangement, can be described as $\hat{\mathbf{w}}_{MAP} = (\mathbf{X}^\top \mathbf{X} + \frac{\sigma^2}{\tau^2} \mathbf{I})^{-1} (\mathbf{X}^\top \mathbf{y})$. Comparing the mean of the posterior with the frequentist solution indicates that the two estimates are very similar with the only difference that the MAP estimate has the additive term $\frac{\sigma^2}{\tau^2} \mathbf{I}$ in the covariance matrix. This added term is the direct outcome of our assumptions in the likelihood and prior of the model which were not present in our simple frequentist model. This term can also be interpreted as a regularization parameter that controls the variance of weights. The designer of the model may want to consider different types of regularization or constraints in the model, which in probabilistic modelling is usually done by incorporating them in the prior distribution. Such problems may not have an analytical posterior available. We will consider two such models in the following subsections.

Sparsity-inducing priors

In many prediction problems, in particular those that require human intervention in a form, the number of available training data is smaller than dimensions of the problem. If not regularized properly, a model trained in this setting may overfit to the training data (achieving very low training error) while not being able to generalize properly to unobserved data (high test data error). One way to tackle this challenge, is to directly regularize the model parameters so that they would not overfit. This regularization can be done by adding penalty terms, for example l_1 norms of the weights, to the error function in the non-Bayesian models with the general idea of pushing weights that are not useful toward zero (see for example Lasso [38]). In probabilistic modelling, we can achieve the same goal by selecting sparsity-inducing priors.

There are different priors that encourage sparsity but they can be categorized to two general groups of mixture priors, such as spike-and-slab prior [17], and the continuous shrinkage priors, such as horseshoe [27] and Laplace prior [34]. The core idea of these priors is that their densities for coefficients peak at zero but at the same time have good amount of probability mass in non-zero values. In this thesis we consider the spike-and-slab prior as it introduces a binary latent variable that explicitly indicates whether a coefficient should be zero or non-zero (relevant and not-relevant groups). This explicit explanation could be very useful as it is compatible with human opinion about inclusion/exclusion of variables in a model (we will discuss this link in detail in Section 2.3).

The linear regression model with sparsity-inducing spike-and-slab prior on coefficients \mathbf{w} can be described as

$$\begin{aligned} \mathbf{y} &\sim \mathcal{N}(\mathbf{X}\mathbf{w}, \sigma^2 \mathbf{I}), \\ \sigma^{-2} &\sim \text{Gamma}(\alpha_\sigma, \beta_\sigma), \\ w^j &\sim \gamma^j \mathcal{N}(0, \tau^2) + (1 - \gamma^j) \delta_0, \quad j = 1, \dots, d, \end{aligned} \tag{2.8}$$

$$\gamma^j \sim \text{Bernoulli}(\rho), \quad j = 1, \dots, d,$$

where δ_0 is a Dirac delta point mass at zero, and γ^j is the binary variable that indicates whether the corresponding coefficient w^j should be excluded from the model (if $\gamma^j = 0$, then $w^j = 0$) or should it be addressed as a normal variable (if $\gamma^j = 1$, then $w^j \sim \text{N}(0, \tau^2)$). As we are interested in the behavior of γ^j s, we considered a Bernouli prior distribution on all of them with the prior inclusion probability ρ as a fixed hyperparameter that controls the expected number of non-zero covariates. Unlike the simple model introduced before, here we consider σ^2 as an unknown parameter and assume a prior distribution for it with α_σ and β_σ as fixed hyperparameters. for the three unobserved parameters \mathbf{w} , $\boldsymbol{\gamma}$, and σ^2 the posterior can be derived following the Bayes rule

$$p(\mathbf{w}, \boldsymbol{\gamma}, \sigma^2 | \mathcal{D}) = \frac{p(\mathcal{D} | \mathbf{w}, \sigma^2) p(\mathbf{w} | \boldsymbol{\gamma}) p(\sigma^2) p(\boldsymbol{\gamma})}{p(\mathcal{D})}. \quad (2.9)$$

The posterior does not have an analytical solution. We will discuss different methods to compute the posterior of these types of problems in Section 2.4. The introduced spike-and-slab Sparsity-inducing model is used as the underlying data model in Publication II, III, and IV.

Detecting outliers

In regression, there may be observations that have response values substantially different from all other data. These highly noisy observations, called outliers, can considerably affect the results if not accounted properly. A Bayesian way to handle this is to consider different variance for residual error of each observation [39]. This can be implemented by changing the observational model from $y_i \sim \text{N}(\mathbf{x}_i^\top \mathbf{w}, \sigma^2)$, where there was a global parameter σ^2 for the variance of all observations, to $y_i \sim \text{N}(\mathbf{x}_i^\top, \frac{\sigma^2}{v_i})$, where the parameter v_i controls the noise variance per observation. As v_i is an unobserved parameter, we consider a prior distribution for it to complete the Bayesian loop

$$\begin{aligned} y_i &\sim \text{N}(\mathbf{x}_i^\top \mathbf{w}, \frac{\sigma^2}{v_i}), & i = 1, \dots, n, \\ v_i &\sim \text{Gamma}(\alpha_v, \beta_v), & i = 1, \dots, n, \\ \sigma^{-2} &\sim \text{Gamma}(\alpha_\sigma, \beta_\sigma), \\ \mathbf{w} &\sim \text{N}(0, \tau^2 \mathbf{I}), \end{aligned} \quad (2.10)$$

where α_v , β_v , α_σ , β_σ , and τ^2 are fixed hyperparameters and n is the number of observations. For the unobserved parameters \mathbf{w} , σ^2 , and \mathbf{v} (as a vector of all v_i s), the posterior does not have an analytical solution and can be written as

$$p(\mathbf{w}, \mathbf{v}, \sigma^2 | \mathcal{D}) = \frac{p(\mathcal{D} | \mathbf{w}, \sigma^2, \mathbf{v}) p(\mathbf{w}) p(\sigma^2) p(\mathbf{v})}{p(\mathcal{D})}. \quad (2.11)$$

Noisy observations can particularly happen if the data provider is a noisy source. For example in Publication V, we considered the setting where some of the response variables were generated from noisy physiological signals of a human user. This model was employed to handle the potential outliers in the data.

2.3 Modelling user interaction for prediction

A core idea in the thesis is to model user interaction with the probabilistic model as observational feedback. These feedbacks can be tied to the model parameters via conditional distributions (feedback models). The posterior inference can then learn the unobserved parameters given data observations (as introduced in the previous section) and the feedback observations (which will be discussed in this section). The feedback models are used to incorporate the user knowledge into the regression models introduced in the previous section. The following subsections explain the feedback models used in Publication I-V, and their relations to the probabilistic models.

2.3.1 User feedback as observation about model parameters

Publication II and III target the case where the user has knowledge about the regression parameters and can provide feedback if queried. As mentioned, we consider high dimensional scenarios where few data points are available. An example task could be a drug response prediction given genomic features of patients [4]. The genomic feature size may exceed thousands of variables, but only a small number of patient data might be available and it may not be possible (or too expensive) to add new data, which make the prediction task challenging. Fortunately, experts in the field may have experience or knowledge from literature about which features, and in what ways, are relevant for this prediction task. If accounted properly, the expert knowledge can help the prediction performance. Another potential scenario is the sentiment prediction problem [22] when only few texts (e.g., textual reviews of items) with known sentiments (e.g., score of the review) are available. A classical representation of textual data known as bag-of-words, considers each data as a vector of distinct words where the feature value indicates the number of appearance of each word in the text⁴. Humans have intuitions about how individual words may be related to the sentiment (e.g., appearance of the word "awful" in a review may indicate low review score). Our goal is to design probabilistic models that can receive these types of expert knowledge, alongside the training data, to boost the prediction performance.

We consider the linear regression model with sparsity-inducing spike-and-slab

⁴It is also common to represent textual data in dense forms such as [10]. We use bag-of-words due to its simplicity and interpretability of the features.

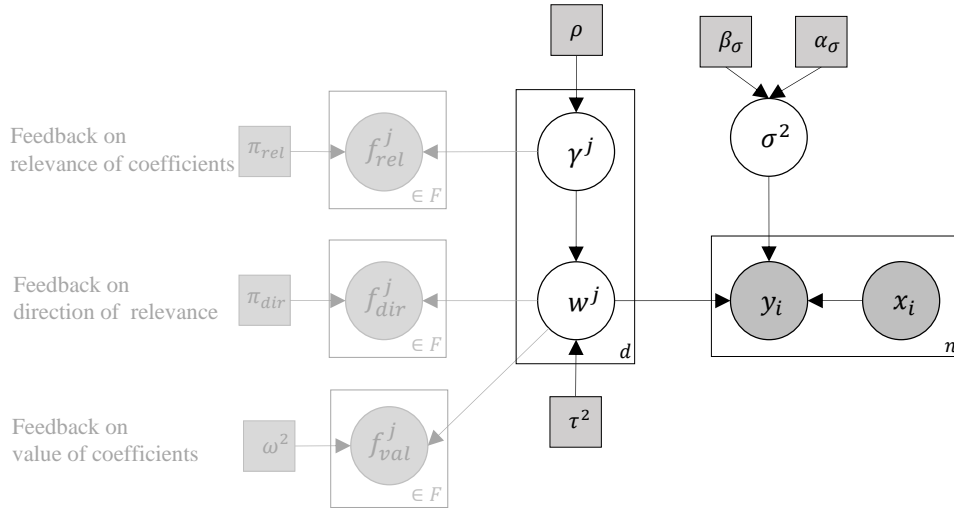


Figure 2.1. Plate notation of the prediction model (right; see Equation 2.8) and user feedback observations (left; see Equations 2.12, 2.13, and 2.14). User feedback influences the data model through observations about model parameters. Circles represent variables (observed variables are shaded), and shaded squares are the fixed hyperparameters.

prior on coefficients introduced in Equation 2.8 as the underlying data model. The type of feedback naturally depends on the task and availability of user knowledge. We consider three simple and natural types of user interaction with the model.

- With some noise, the user can provide feedback about the value of coefficients

$$f_{val}^j \sim \mathcal{N}(w^j, \omega^2), \quad (2.12)$$

where ω^2 models the variance of error in user feedback. Knowledge about value of coefficients is very powerful, however, only available in some specific applications (e.g., exploiting similar trained models or reported values in related literature).

- With some probability, the user can provide feedback about whether a coefficient should be included or excluded from the model

$$f_{rel}^j \sim \gamma^j \text{Bernoulli}(\pi_{rel}) + (1 - \gamma^j) \text{Bernoulli}(1 - \pi_{rel}). \quad (2.13)$$

where π_{rel} indicates the probability that user is correct about the feedback. This relevant/not-relevant feedback is the simplest form of knowledge that a user may have about a regression model.

- With some probability, the user can provide feedback about direction of relevance of a coefficient, i.e., whether a feature is positively or negatively correlated with the response variable

$$f_{dir}^j \sim I(w^j \geq 0) \text{Bernoulli}(\pi_{dir}) + I(w^j < 0) \text{Bernoulli}(1 - \pi_{dir}), \quad (2.14)$$

where π_{dir} indicates the probability that user is correct about the feedback.

The connection of the three mentioned feedback with the data model (Equation 2.8) is shown as a plate diagram in Figure 2.1. The full posterior of the unknown parameters given data observations and the user feedback can be described as

$$p(\mathbf{w}, \gamma, \sigma^2 | \mathcal{D}, \mathcal{F}) = \frac{p(\mathcal{D} | \mathbf{w}, \sigma^2) p(F_{val} | \mathbf{w}) p(F_{rel} | \gamma) p(F_{dir} | \mathbf{w}) p(\mathbf{w} | \gamma) p(\sigma^2) p(\gamma)}{p(\mathcal{D}, \mathcal{F})}, \quad (2.15)$$

where F_{val} , F_{rel} , and F_{dir} are the sets of collected feedback corresponding to the three considered feedback types and $\mathcal{F} = (F_{val}, F_{rel}, F_{dir})$. The posterior computation is discussed in Section 2.4.

Related works

The proposed feedback models provide an intuitive and effortless way for the user to directly influence the prediction model without caring about the complications of the data model. Classical prior elicitation (see for example [26, 15]) aims at eliciting a distribution to represent the expert's knowledge by asking about summary information such as quantiles of the parameters. This is done through iterations between the expert in the related field and statisticians who design the model. Our work goes beyond pure elicitation as it directly connects the expert to the model and also exploits the training data to facilitate the interaction as will be discussed more in Section 3.

We have considered three intuitive types of feedback models. Studies have shown that the type of domain knowledge in prediction tasks can be summarize in a small set [8]. A large body of works have studied the exclusion/inclusion of features (also known as feature selection [9]) in different contexts, for classification [28, 13, 35], and regression [24]. These methods are different to ours from the modelling point as we consider sparse models and also the type of interaction. A different type of feedback is the case where user has knowledge about pairwise similarity of features with respect to the response variable (i.e., asking the user about which pairs of features affect the prediction output similarly) which has been investigated in [1, 2].

2.3.2 User feedback as outcome of a cognitive process

The feedback models proposed in the previous section consider the user as a passive data provider with some noise models. However, humans are more complicated than that. Publication IV investigates a similar knowledge elicitation task to the previous section, but aims at accounting for rational process that humans may undergo to produce the feedback during the interaction. In other words, we will define a more complex feedback model that accounts for the cognitive process of the user, with the idea that performance can be improved if we also model the thought process behind the feedback.

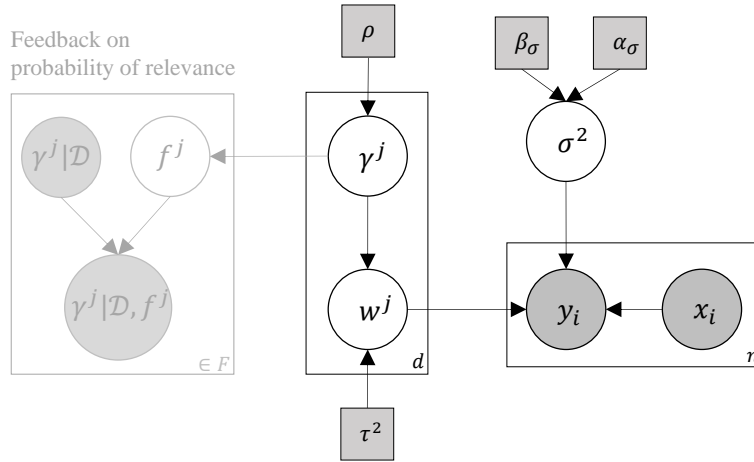


Figure 2.2. Plate notation of the prediction model (right; see Equation 2.8) and user model for feedback (left). The observed user feedback about the probability of relevance of coefficient, $p(\gamma^j | \mathcal{D}, f^j)$, is biased as it is generated from combination of latent user knowledge about the coefficient, i.e., $p(f^j | \gamma^j)$, and information visualized from the data model about probability of relevance of the coefficient, i.e., $p(\gamma^j | \mathcal{D})$. We are interested in inferring the unbiased, latent, version of the feedback to update the parameters of interest in the data model.

In many interaction scenarios, such as visual analytics or human in the loop machine learning systems, certain aspects of the training data, such as statistics or outputs of the machine learning method, may be revealed to the user [11, 31, 1, 25, 36, 41, 20, 21, 32, 24]. This is mainly done to provide information about important characteristic of the data to improve decision making or to guide the user to facilitate the interaction. For example, in the mentioned knowledge elicitation tasks, the system may first show what it has learned about the coefficient of the linear model from the training data and then ask for the user feedback about the same coefficient. Though very common, this type of interaction is prone to overfitting, as the user feedback may not be independent of the knowledge in the training data, while the machine learning methods commonly assume independence between data and user input. Studies in cognitive science have shown that humans are unintentionally biased toward the pieces of information provided for them [40, 15]. We propose a user model that accounts for such bias. In other words, the user model assumes that the user is rational [18] and combines the information provided to her with her knowledge and then provides feedback as the outcome of such cognitive procedure. The user model can then undo the bias in the feedback by performing the inverse of the same rational update.

In particular, we consider the case where the user provides feedback about the probability of relevance of a coefficient. The belief of the model (based on data alone) about the probability of relevance of the coefficient, i.e., marginal posterior probability $p(\gamma^j | \mathcal{D})$, is also visualized for the user. We make the assumption that the user is rational in first combining the visualized information from data with her latent knowledge. The reported feedback is the updated knowledge which is

naturally biased as it is partially coming from information in the training data. The model should consider this while inferring the posterior of the parameters of interest given the user feedback. Figure 2.2 depicts a schematic of the data and user feedback model for this problem.

We assume that the user is rational and combines her latent knowledge about the probability of relevance of the j^{th} coefficient, i.e., $p(f^j | \gamma^j = 1)$, with the revealed information from the training data, i.e., $p(\gamma^j = 1 | \mathcal{D})$, following a Bayesian update

$$p(\gamma^j = 1 | \mathcal{D}, f^j) \propto p(f^j | \gamma^j = 1)p(\gamma^j = 1 | \mathcal{D}), \quad (2.16)$$

and provides as feedback the resulting posterior $p(\gamma^j = 1 | \mathcal{D}, f^j)$. The latent feedback likelihood can be represented as $p(f^j | \gamma^j) = \pi^j \gamma^j + (1 - \pi^j)(1 - \gamma^j)$, with π^j being the likelihood for latent feedback when $\gamma^j = 1$. By doing the reverse of the Bayesian update in 2.16, we can infer π^j as:

$$\pi^j \propto \frac{p(\gamma^j = 1 | \mathcal{D}, f^j)}{p(\gamma^j = 1 | \mathcal{D})}. \quad (2.17)$$

We can then use the inferred user feedback $p(f^j | \gamma^j)$, instead of the observed one, to update the posterior for the parameters of interest. Publication IV shows in a simple user study that the prediction performance in a simple sentiment analysis task can be improved if the user model accounts for the rational updates that the user may undergo.

Related works

Human bias in interactive tools has been discussed in prior elicitation [15], and visual analytics [42]. In particular, [15] reviews the common human biases from psychological literature and provides guidelines for how to reliably extract expert knowledge about uncertain quantities. [42] proposed a framework for measurement of such biases in visual analytics tools and investigated how to overcome it through informing the user of potential biases. Our work is fundamentally different as we are not limiting or interrupting the interaction (by just following a guideline or showing a warning), but rather, acknowledge the presence of such biases and allow the user model, as part of the system, to account for that.

2.3.3 User feedback for intent modelling

Intent modelling describes the task of learning the hidden intent of the user from user feedbacks. Unlike the previous mentioned scenarios where the user feedback was used as complementary information to the training data, in Publication I and V the task is to predict the user intent based on user feedback alone. An important instance of this scenario is the interactive personalized search systems where in each iteration the system recommends items and the user provides feedback regarding relevance of recommendations. The goal is

not just to better predict relevance of items for the user but rather to find the most relevant item with minimal interaction. In this section, we discuss the probabilistic modelling of this task and later in Section 3 discuss how to design the interaction so that the most relevant item can be learned as fast as possible.

Consider the case where a user is searching for a scientific article by forming a query with keywords. The user is uncertain about the exact description of the article (e.g., the user may not know the title), however, if the article is recommended, the user can assess the relevance of it. This information seeking task is usually done in several iterations starting with an initial query, assessing the results recommended by the system, and modifying the query with the goal of getting better results. This type of information seeking with the user being uncertain about her information need, is known as exploratory search [23, 43] and covers around half of search behaviours of users [37] (see [5] for more in depth review for exploratory and standard lookup search). The main bottle necks of such search systems are that (i) the amount of user feedback that can be gathered is very limited compared to the size of the information space, and (ii) the users are mostly sometimes reluctant to give more than few explicit feedback after assessing the recommended results. To tackle these challenges, Publication I, exploits the relationship between articles and their keywords and defines feedback likelihoods on both keywords and articles, providing more flexibility for the user to express its intent. Furthermore, the modelling of noisy implicit feedback, in particular feedback generated from brain activity measured through electroencephalography (EEG) and eye movements, is investigated in Publication V.

The intent model is defined as function that maps all the keywords (used to express the search intent) and articles to real values, indicating the relevance of each item for the user. As the number of user feedback is few, it is reasonable to use a linear model to define the relation of user feedback on keywords to the hidden intent (denoted by \mathbf{w})

$$f_{kw}^i \sim \mathcal{N}(\mathbf{x}_i^\top \mathbf{w}, \sigma_{kw}^2), \quad (2.18)$$

where σ_{kw}^2 models the noise of the keyword feedback. The keywords are represented in the keyword-document matrix $X \in \mathbb{R}^{k \times d}$ where the element (i, j) describes the tf-idf weighting of keyword i (out of k total keywords) in document j (out of d total documents), and \mathbf{x}_i^\top indicates the i^{th} row of the matrix. As mentioned, we are interested to also model potential user feedback on the documents. We make the simplifying assumption that the expected relevance of a document can be represented as a weighted sum of the expected relevance of keywords that appear in it

$$\mathbb{E}[f_{doc}^j] = \sum_{i=1}^k p_{(i|j)} \mathbb{E}[f_{kw}^i], \quad (2.19)$$

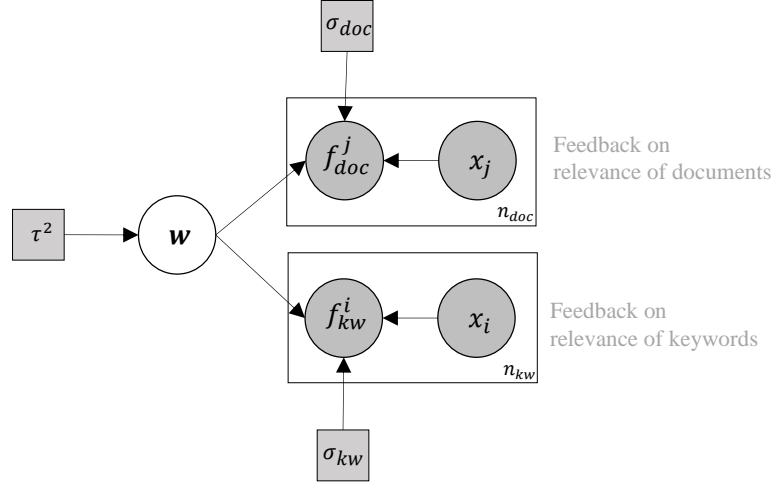


Figure 2.3. Plate notation of the intent modelling from feedback on recommended documents and keywords. The feature transformation are now shown. The feature for the i^{th} keyword (shown as x_i) is the i^{th} row of keyword document matrix X , and for the j^{th} document (shown as x_j) is the j^{th} column of $\hat{X}^\top X$.

where f_{doc}^j indicates the relevance of the j^{th} documents, $\mathbb{E}[\cdot]$ denotes the expected value and $p_{(i|j)}$ is the likelihood of the i^{th} keyword being present in the j^{th} document. This likelihood is not available but can be approximated by normalizing X such that its columns sum up to one. Denoting the resulting matrix as \hat{X} and following following 2.18, the feedback for relevance of documents can be described as

$$F_{doc} \sim N(\hat{X}^\top X w, \sigma_{doc}^2 \mathbf{I}), \quad (2.20)$$

where $F_{doc} = [f_{doc}^1, \dots, f_{doc}^d]$ is the vector representation of relevance feedback for all documents, and σ_{doc}^2 models the noise of the document feedback.

Given user feedback on keywords and documents the posterior of the hidden intent can be learned by assuming a Gaussian prior on the intent weights, i.e., $w \sim N(0, \tau^2 \mathbf{I})$. As the both likelihoods follow a Gaussian distribution, the posterior has a closed form solution and can be derived following the same derivations as the simple linear model in 2.7. Figure 2.3 illustrates the schematic of user feedback models and its connection to the latent intent.

Implicit feedback model

The model proposed model is very simple as the noise of feedback is assumed to be known and the only unknown parameter is the intent vector w . As mentioned, we are interested to consider feedbacks from implicit sources such as neurophysiologic signals gathered from brain activities⁵. The implicit feedback

⁵Note that here we assume that the implicit feedbacks, EEG signals, are already converted to a continuous relevance value (see Publication V for details about this conversion).

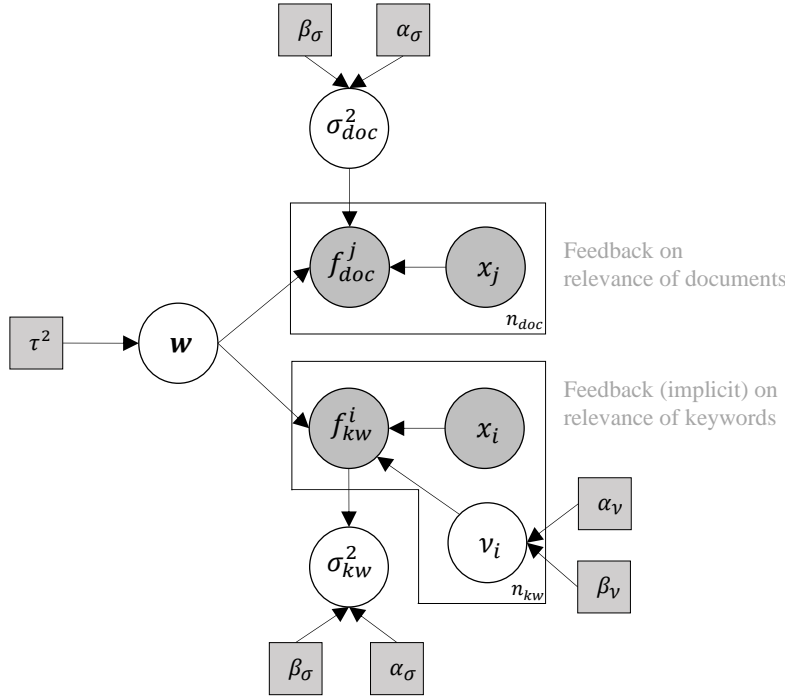


Figure 2.4. Plate notation of the intent modelling from feedback on recommended documents and noisy, implicit feedback on keywords. The data model is based on 2.10 with the feedbacks as the observational data. The feature vector of documents follows the transformation introduced in equation 2.20

is extremely cheap, as the it requires no physical effort from the user, but at the same time extremely noisy. Our modelling solution is to use the automatic relevance detection model proposed in 2.10 to endow the feedback likelihood handle the inherent noise in the implicit feedback when used in combination of the explicit feedbacks. For simplicity, let's assume that only the keyword feedback is generated from an implicit source (it is straightforward to consider a similar feedback source for documents or to consider both explicit and implicit feedback for keywords or documents). The implicit feedback likelihood for keywords can be expressed as

$$f_{kw}^i \sim \mathcal{N}(\mathbf{x}_i^\top \mathbf{w}, \frac{\sigma_{kw}^2}{v_i}), \quad (2.21)$$

where the parameters follow the model 2.10. The model can then infer the accuracy of feedback v_i based on the observed feedback and handle the high uncertainty in implicit feedback. The resulting model for a case of availability of explicit feedback on documents and implicit feedback on keywords is depicted in Figure 2.4. The resulting posterior is analytically intractable.

Related works

Intent modelling for article search has been studied in [19, 29, 30]. These works are different to ours as they only define the intent on keywords and then use the

learned intent as an input to a separate retrieval model (based on a standard language model) to retrieve ranked documents, while document retrieval is a natural outcome of the document feedback likelihood in our work. Furthermore, the previous methods did not consider the option of user providing feedback on documents. We borrowed the same user interface for providing feedback on keywords which is based on the idea of visualizing the top relevant keywords in a radar-like interface where the distance to center for each keyword is proportional to the predicted relevance from the intent model. The user then can provide feedback by dragging a keyword toward the center (indicating positive relevance) or further away from the center (indicating negative feedback). We added document feedback through clicks or bookmarking.

2.4 Posterior inference

Other than simple models (like the linear regression introduced in 2.7) the models proposed in this chapter do not have a closed-form posterior and posterior predictive distributions. For low dimensional posteriors, it is still practical to approximate the posterior distribution by simulation methods such as numerical computation on grids or Monte Carlo simulations (see [16, Chapter 10] for a review). However, as mentioned, we are interested on high-dimensional problems (e.g., sentiment analysis or search with textual data or drug prediction with genomic data), where the simple methods cannot scale well due to the curse of dimensionality (the number of evaluations grows exponentially with respect to the dimension). There are two general family of methods to approach such intractable posterior.

Sampling based methods

MCMC.. Recent advances in sampling based methods has resulted in probabilistic programming languages that such as Stan [7] and Pyro [6] that made the computation of Bayesian inference straightforward for end users. The user of such software needs to declare the inference problem, and the probabilistic programming language performs the inference (for example using and provides posterior samples that can be used for next steps (for example see [33] for a step-by-step guides to do Bayesian modelling and inference using Stan).

Deterministic posterior approximation

- normal (Laplace) approximation where the posterior is approximated with multivariate normal distribution (see [16, Chapter 4] for a review), MORE IN chapter 13

2.5 Key components of the joint model

Let y and x denote the outputs (target variables) and inputs (covariates), and θ and ϕ_y the model parameters. Let f encode input (*feedback*) from the user, presumably a domain expert, and ϕ_f be model parameters related to the user input. We identify the following key components:

1. An observation model $p(y | x, \theta, \phi_y)$ for y .
2. A feedback model $p(f | \theta, \phi_f)$ for the expert's knowledge.
3. A prior model $p(\theta, \phi_y, \phi_f)$ completing the hierarchical model description.
4. A query algorithm and user interface that facilitate gathering f iteratively from the expert.
5. Update process of the model after user interaction.

The observation model can be any appropriate probability model. It is assumed that there is some parameter θ , possibly high-dimensional, that the expert has knowledge about. The expert's knowledge is encoded as (possibly partial) feedback f that is transformed into information about θ via the feedback model. Of course, there could be a more complex hierarchy tying the observation and feedback models, and the feedback model can also be used to model more user-centric issues, such as the quality of or uncertainty in the knowledge or user's interests.

TO EXPLAIN: Why we use linear models? because of the risk for overfitting if the model is not regularized enough... and also mentioning what model was used in each publication?

3. User interaction with the probabilistic model

3.1 Active learning and experimental design

3.2 Multi-armed bandits and Bayesian optimization

4. Summary of the Contributions

This chapter briefly summarizes the contributions of Publications I-V with emphasize on answering the research questions of the thesis.

4.1 Interactive intent modelling from multiple feedback domains (Publications I and V)

4.2 Expert knowledge elicitation for high-dimensional prediction (Publications II and III)

4.3 User modelling for avoiding overfitting in knowledge elicitation (Publication IV)

5. Discussion

References

- [1] Homayun Afrabandpey, Tomi Peltola, and Samuel Kaski. Interactive prior elicitation of feature similarities for small sample size prediction. In *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization, UMAP '17*, pages 265–269, New York, NY, USA, 2017. ACM.
- [2] Homayun Afrabandpey, Tomi Peltola, and Samuel Kaski. Human-in-the-loop active covariance learning for improving prediction in small data sets. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 1959–1966. International Joint Conferences on Artificial Intelligence Organization, 7 2019.
- [3] Saleema Amershi, Maya Cakmak, William Bradley Knox, and Todd Kulesza. Power to the people: The role of humans in interactive machine learning. *AI Magazine*, 35(4):105–120, 2014.
- [4] Muhammad Ammad-ud din, Suleiman A. Khan, Disha Malani, Astrid Murumägi, Olli Kallioniemi, Tero Aittokallio, and Samuel Kaski. Drug response prediction by inferring pathway-response associations with kernelized Bayesian matrix factorization. *Bioinformatics*, 32(17):i455–i463, 08 2016.
- [5] Kumaripaba Athukorala, Dorota Glowacka, Giulio Jacucci, Antti Oulasvirta, and Jilles Vreeken. Is exploratory search different? a comparison of information search behavior for exploratory and lookup tasks. *Journal of the Association for Information Science and Technology*, 67(11):2635–2651, 2016.
- [6] Eli Bingham, Jonathan P. Chen, Martin Jankowiak, Fritz Obermeyer, Neeraj Pradhan, Theofanis Karaletsos, Rohit Singh, Paul Szerlip, Paul Horsfall, and Noah D. Goodman. Pyro: Deep Universal Probabilistic Programming. *Journal of Machine Learning Research*, 2018.
- [7] Bob Carpenter, Andrew Gelman, Matthew Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. Stan: A probabilistic programming language. *Journal of Statistical Software, Articles*, 76(1):1–32, 2017.
- [8] In Kwon Choi, Taylor Childers, Nirmal Kumar Raveendranath, Swati Mishra, Kyle Harris, and Khairi Reda. Concept-driven visual analytics: An exploratory study of model- and hypothesis-based reasoning with visualizations. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, CHI '19*, pages 68:1–68:14, New York, NY, USA, 2019. ACM.
- [9] Alvaro H. C. Correia and Freddy Lécué. Human-in-the-loop feature selection. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence*, 2019.
- [10] Andrew M Dai and Quoc V Le. Semi-supervised sequence learning. In *Advances in neural information processing systems*, pages 3079–3087, 2015.

- [11] S. Das, D. Cashman, R. Chang, and A. Endert. Beames: Interactive multimodel steering, selection, and inspection for regression tasks. *IEEE Computer Graphics and Applications*, 39(5):20–32, Sep. 2019.
- [12] David Donoho and Jared Tanner. Observed universality of phase transitions in high-dimensional geometry, with implications for modern data analysis and signal processing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 367(1906):4273–4293, 2009.
- [13] Gregory Druck, Burr Settles, and Andrew McCallum. Active learning by labeling features. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 81–90, 2009.
- [14] Gerhard Fischer. User modeling in human–computer interaction. *User Modeling and User-Adapted Interaction*, 11(1-2):65–86, 2001.
- [15] Paul H Garthwaite, Joseph B Kadane, and Anthony O’Hagan. Statistical methods for eliciting probability distributions. *Journal of the American Statistical Association*, 100(470):680–701, 2005.
- [16] Andrew Gelman, John B Carlin, Hal S Stern, David B Dunson, Aki Vehtari, and Donald B Rubin. *Bayesian data analysis*. Chapman & Hall/CRC, 3rd edition, 2014.
- [17] Edward I George and Robert E McCulloch. Variable selection via Gibbs sampling. *Journal of the American Statistical Association*, 88(423):881–889, 1993.
- [18] Samuel J Gershman, Eric J Horvitz, and Joshua B Tenenbaum. Computational rationality: A converging paradigm for intelligence in brains, minds, and machines. *Science*, 349(6245):273–278, 2015.
- [19] Dorota Głowacka, Tuukka Ruotsalo, Ksenia Konuyshkova, kumaripaba Athukorala, Samuel Kaski, and Giulio Jacucci. Directing exploratory search: Reinforcement learning from user interactions with keywords. In *Proceedings of the 2013 International Conference on Intelligent User Interfaces*, IUI ’13, pages 117–128, New York, NY, USA, 2013. ACM.
- [20] Ashish Kapoor, Bongshin Lee, Desney Tan, and Eric Horvitz. Interactive optimization for steering machine classification. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI ’10, pages 1343–1352, New York, NY, USA, 2010. ACM.
- [21] Josua Krause, Adam Perer, and Enrico Bertini. INFUSE: interactive feature selection for predictive modeling of high dimensional data. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1614–1623, 2014.
- [22] Bing Liu. *Sentiment analysis: Mining opinions, sentiments, and emotions*. Cambridge University Press, 2015.
- [23] Gary Marchionini. Exploratory search: From finding to understanding. *Communications of the ACM*, 49(4):41–46, 2006.
- [24] Luana Micallef, Iris Sundin, Pekka Marttinen, Muhammad Ammad-ud din, Tomi Peltola, Marta Soare, Giulio Jacucci, and Samuel Kaski. Interactive elicitation of knowledge on feature relevance improves predictions in small data sets. In *Proceedings of the 22Nd International Conference on Intelligent User Interfaces*, IUI ’17, pages 547–552, New York, NY, USA, 2017. ACM.
- [25] Thomas Mühlbacher and Harald Piringer. A partition-based framework for building and validating regression models. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):1962–1971, 2013.
- [26] Anthony O’Hagan, Caitlin E. Buck, Alireza Daneshkhah, J. Richard Eiser, Paul H. Garthwaite, David J. Jenkinson, Jeremy E. Oakley, and Tim Rakow. *Uncertain Judgements. Eliciting Experts’ Probabilistic*. Wiley, Chichester, England, 2006.

- [27] Juho Piironen and Aki Vehtari. Sparsity information and regularization in the horseshoe and other shrinkage priors. *Electron. J. Statist.*, 11(2):5018–5051, 2017.
- [28] Hema Raghavan, Omid Madani, and Rosie Jones. Active learning with feedback on features and instances. *Journal of Machine Learning Research*, 7(Aug):1655–1686, 2006.
- [29] Tuukka Ruotsalo, Giulio Jacucci, Petri Myllymäki, and Samuel Kaski. Interactive intent modeling: Information discovery beyond search. *Communications of the ACM*, 58(1):86–92, 2015.
- [30] Tuukka Ruotsalo, Jaakko Peltonen, Manuel J. A. Eugster, Dorota Głowacka, Patrik Floréen, Petri Myllymäki, Giulio Jacucci, and Samuel Kaski. Interactive intent modeling for exploratory search. *ACM Trans. Inf. Syst.*, 36(4):44:1–44:46, October 2018.
- [31] Dominik Sacha, Michael Sedlmair, Leishi Zhang, John A Lee, Jaakko Peltonen, Daniel Weiskopf, Stephen C North, and Daniel A Keim. What you see is what you can change: Human-centered machine learning by interactive visualization. *Neurocomputing*, 268:164–175, 2017.
- [32] Advait Sarkar, Mateja Jamnik, Alan F Blackwell, and Martin Spott. Interactive visual machine learning in spreadsheets. In *Visual Languages and Human-Centric Computing (VL/HCC), 2015 IEEE Symposium on*, pages 159–163. IEEE, 2015.
- [33] Daniel J Schad, Michael Betancourt, and Shravan Vasishth. Toward a principled bayesian workflow in cognitive science. *arXiv preprint arXiv:1904.12765*, 2019.
- [34] Matthias W Seeger. Bayesian inference and optimal design for the sparse linear model. *Journal of Machine Learning Research*, 9:759–813, 2008.
- [35] Burr Settles. Closing the loop: Fast, interactive semi-supervised annotation with queries on features and instances. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1467–1478, 2011.
- [36] Justin Talbot, Bongshin Lee, Ashish Kapoor, and Desney S. Tan. Ensemblematrix: Interactive visualization to support machine learning with multiple classifiers. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI ’09, pages 1283–1292, New York, NY, USA, 2009. ACM.
- [37] Jaime Teevan, Christine Alvarado, Mark S. Ackerman, and David R. Karger. The perfect search engine is not enough: A study of orienteering behavior in directed search. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI ’04, pages 415–422, New York, NY, USA, 2004. ACM.
- [38] Robert Tibshirani. Regression shrinkage and selection via the lasso: a retrospective. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(3):273–282, 2011.
- [39] J. Ting, A. D’Souza, and S. Schaal. Automatic outlier detection: A bayesian approach. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 2489–2494, April 2007.
- [40] Amos Tversky and Daniel Kahneman. Judgment under uncertainty: Heuristics and biases. *Science*, 185(4157):1124–1131, 1974.
- [41] Stef Van Den Elzen and Jarke J van Wijk. BaobabView: Interactive construction and analysis of decision trees. In *Proceedings of IEEE Symposium on Visual Analytics Science and Technology (VAST)*, pages 151–160. IEEE, 2011.
- [42] E. Wall, L. M. Blaha, L. Franklin, and A. Endert. Warning, bias may occur: A proposed approach to detecting cognitive bias in interactive visual analytics. In *2017 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 104–115, Oct 2017.

References

- [43] Ryen W White and Resa A Roth. Exploratory search: Beyond the query-response paradigm. *Synthesis lectures on information concepts, retrieval, and services*, 1(1):1–98, 2009.

Publication I

Pedram Daee, Joel Pyykkö, Dorota Glowacka, and Samuel Kaski. Interactive Intent Modeling from Multiple Feedback Domains. In *Proceedings of the 21st International Conference on Intelligent User Interfaces*, Sonoma, California, USA, 71–75, March 2016.

© 2016 ACM

Reprinted with permission.

Publication II

Pedram Daee^{*}, Tomi Peltola^{*}, Marta Soare^{*}, and Samuel Kaski. Knowledge elicitation via sequential probabilistic inference for high-dimensional prediction. *Machine Learning*, 106, 9-10, 1599–1620, 2017.

© 2017

Reprinted with permission.

Publication III

Iiris Sundin*, Tomi Peltola*, Luana Micallef, Homayun Afrabandpey, Marta Soare, Muntasir Mamun Majumder, Pedram Daei, Chen He, Baris Serim, Aki Havulinna, Caroline Heckman, Giulio Jacucci, Pekka Marttinen, and Samuel Kaski. Improving genomics-based predictions for precision medicine through active elicitation of expert knowledge. *Bioinformatics*, 34, 13, i395–i403, 2018.

© 2018

Reprinted with permission.

Publication IV

Pedram Daee^{*}, Tomi Peltola^{*}, Aki Vehtari, and Samuel Kaski. User Modelling for Avoiding Overfitting in Interactive Knowledge Elicitation for Prediction. In *Proceedings of the 23rd International Conference on Intelligent User Interfaces*, Tokyo, Japan, 305–310, March 2018.

© 2018 ACM

Reprinted with permission.

Publication V

Giulio Jacucci, Oswald Barral, Pedram Daei, Markus Wenzel, Baris Serim, Tuukka Ruotsalo, Patrik Pluchino, Jonathan Freeman, Luciano Gamberini, Samuel Kaski, Benjamin Blankertz. Integrating Neurophysiological Relevance Feedback in Intent Modeling for Information Retrieval. *Journal of the Association for Information Science and Technology*, 2019.

© 2019

Reprinted with permission.