

BACHELOR THESIS  
ARTIFICIAL INTELLIGENCE

**Radboud University**



---

**Music Emotion Recognition using  
Multi-Output Gaussian Processes**

---

*Author:*  
Sigur de Vries  
1009337

*First supervisor:*  
dr. M. Hinne  
Artificial Intelligence  
m.hinne@donders.ru.nl

*Second supervisor:*  
dr. P. L. Lanillos Pradas  
Artificial Intelligence  
p.lanillos@donders.ru.nl



January 19, 2021

## Abstract

Recognition of emotion in music is an important subject within the domain of Music Information Retrieval. When emotion is defined by the continuous parameters arousal and valence, the problem of emotion recognition can be solved with a regression approach. In general, arousal and valence are assumed to be independent, however this paper investigates if including a correlation between the variables will improve the accuracy of emotion recognition in music. A Gaussian Processes model is proposed to solve this regression problem. Gaussian Processes (GP) return a distribution over functions which is described by a mean and a covariance function. This means that the GP are flexible in describing data and can make accurate predictions based on a small amount of training input. A Multi-Output GP (MOGP) model is used to capture the correlation between arousal and valence. The MOGP model adds an extra dimension to the covariance matrix to include the covariance of the outputs. The MOGP is compared with a normal GP and a Singular Value Regression model on the task of predicting arousal and valence values for songs using auditory features. The GP and MOGP obtain equal accuracies for their predictions. However, the predictions of the MOGP have lower variances, which means that the MOGP is more reliable to sample from.

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Related Work</b>	<b>6</b>
<b>3</b>	<b>Theoretical Background</b>	<b>8</b>
3.1	Gaussian Processes . . . . .	8
3.2	Multi-Output Gaussian Processes . . . . .	9
<b>4</b>	<b>Research</b>	<b>11</b>
4.1	Data . . . . .	11
4.2	Methods . . . . .	12
4.3	Results . . . . .	12
<b>5</b>	<b>Discussion</b>	<b>17</b>
<b>6</b>	<b>Acknowledgements</b>	<b>19</b>
	<b>References</b>	<b>20</b>

# Chapter 1

## Introduction

With the shift from physical music medium to online streaming, many opportunities arise for music information retrieval (MIR). MIR is interesting for three groups; streaming services and other music industries dedicated to selling music, the artists and producers, and the users of music services (Casey et al., 2008). A main goal for the music industries and most artists is to make as much profit as possible, for which MIR is an essential strategy to achieve this goal. For users it is most interesting to be introduced to new music. Both personal usage and data about music can be used to improve personalized recommendations and playlists for users.

Nowadays, mainly metadata (Casey et al., 2008) and collaborative filtering are used for personalized recommendations. Metadata is the information about songs like the artists, year of release, album, etc. Collaborative filtering compares user's preferences to make predictions for new users (Su & Khoshgoftaar, 2009). Where Netflix looks at the user's given star ratings, Spotify uses implicit feedback only, for example which songs are streamed often and which are skipped (Giacaglia, 2019). This collaborative filtering data is then combined with metadata, so that the user will receive personalized recommendations.

Nonetheless, these recommendation algorithms are not yet flawless, both in terms of efficiency and accuracy. Collaborative filtering underperforms for unused items and is too dependent on human reviews (Su & Khoshgoftaar, 2009). The main issue with metadata is that it has to be entered manually and it lacks objectivity and precision. A new approach in the field of MIR is the use of audio content. Audio features represent the measurements of an audio signal. The features are either based on a certain time interval of the audio signal or features that are aligned to the beat. The features are subtracted from the spectrum, pitch, frequency, tempo and beat (Casey et al., 2008).

Data about auditory content is predominantly used for genre classification and artist identification (Markov & Matsui, 2014). Machine learning

algorithms such as k-nearest neighbor, k-means, multi-class SVM, and neural networks have been used for genre classification based on auditory features (Haggblade, Hong, & Kao, 2011). Since the ML algorithms can be trained efficiently without being dependent on humans and the accuracy of classification is fair, it becomes appealing to replace collaborative filtering with machine learning algorithms in recommendation systems.

One of the main objective of music is to evoke emotions in listeners (Markov & Matsui, 2014). Therefore defining similarity of songs in terms of their emotional values makes more sense than using artist or genre similarity to generate recommendations. In current research, emotion is only used occasionally for personal recommendations. Song recommendations based on a user's mood could be more effective than on content or metadata. To include an individual's mood in music recommendation, a music recommendation system was implemented which measures the individual's mood with wearable psychological sensors (Ayata, Yaslan, & Kamasak, 2018). These recommendations match the user's personal preferences at all times. However for streaming services physically measuring emotions is an unrealistic option since they lack physical contact with their users and the audience is too large to perform personal measurements.

An alternative would be to generate song recommendations based on the song's emotion similarity. Just like genres, songs can be classified into a set of emotions defined by humans. The best emotion classifier in music only achieves an accuracy of 65% (Cardoso, Panda, & Paiva, 2011), which is not good enough to generate playlists with. Additionally, moods can vary even within classes, therefore emotion-based recommendations are inaccurate as well (Yang, Lin, Su, & Chen, 2007).

Similarity between songs is better expressed when the songs are represented by continuous values instead of distinct classes. Therefore, Russell's model is proposed (Markov & Matsui, 2014), which describes emotions in a continuous model. In this model, mood is represented by two independent continuous variables: arousal and valence. Arousal describes the energy of mood and valence describes the positivity of mood. Russell's model is shown in figure 1.1.

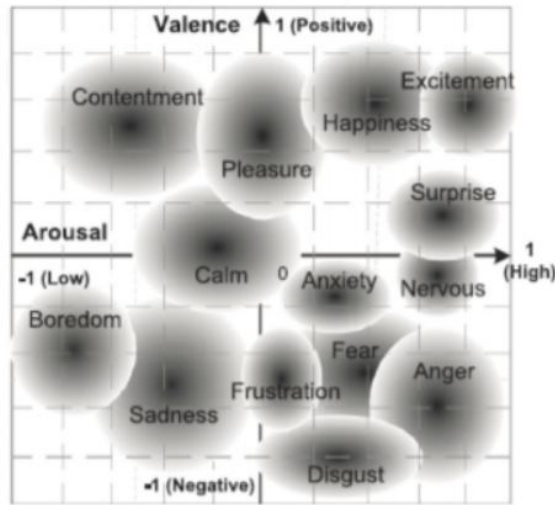


Figure 1.1: Russell’s model (Markov & Matsui, 2014). Valence is shown on the y-axis and arousal on the x-axis. Moods are represented accordingly to these values in the plane.

With the assumption that emotions are explained by two continuous values, these values can be used to compute similarities among songs and generate playlists with emotionally similar songs. In this paper, several models will be trained to predict valence and arousal values based on the song’s auditory features. The datasets used in this research includes the valence and arousal values, which have been annotated by participants (Soleymani, Aljanaki, & Yang, 2016), (Soleymani, Caro, Schmidt, Sha, & Yang, 2013). Averaging the annotations over a number of human measurements is a representative value to train algorithms with, therefore the algorithms can predict sensible values.

However, the notion that valence and arousal are independent is questionable and recent studies show that they are actually dependent to some degree (Bestelmeyer, Kotz, & Belin, 2017). This means that the regression model has to be adapted in order to predict multiple, correlated parameters. There has not been done research on emotion recognition using dependent continuous variables yet, therefore it is unknown if it is more accurate than regarding the variables as independent.

The machine learning algorithms used for genre classification can be used in emotion classification as well. Currently, Support Vector Machine (SVM) is the dominant algorithm in the domain of MIR in general as well as in emotion classification (Markov & Matsui, 2014), however, recently Gaussian Processes (GP) gained interest for both regression and classification tasks in multiple domains, including music analysis.

A Gaussian Process is a probability distribution over functions and is

defined by a mean and a covariance matrix between inputs. The resulting functions are only limited to a mean and covariance function in terms of structure, which means they are flexible in fitting the data. The main advantages of GP are that it can be trained using small training sets (Platt, Burges, Swenson, Weare, & Zheng, 2001) and the predictions are probabilistic, thus include the uncertainty of the prediction (Markov & Matsui, 2014).

Gaussian Processes can be applied to the regression problem of emotion prediction as well (Markov & Matsui, 2014). Markov defines valence and arousal as “continuous and independent parameters”, therefore the values can be predicted by two different regression models, but with the same data. For independent parameters, different models are sufficient to solve the regression task. However, to include the covariance of two parameters in a model, a multi-output model has to be used.

Multi-Output Gaussian Processes are used to predict multiple correlated outputs. The main difference between standard GP and MOGP is that an extra covariance function has to be included to explain the covariance between the outputs. The covariance of outputs is either constant for every input or it depends on the input (Alvarez, Rosasco, & Lawrence, 2011). If valence and arousal are dependent, the MOGP should obtain more accurate predictions than two separate GPs.

The goal of this work is to apply a Multi-Output Gaussian Process to the continuous emotion regression problem. The MOGP is compared with the single-output GP model to discover if there is an improvement when the model is trained with a covariance between the variables. Additionally, both are compared to a Singular Value Regression model (SVR) since it is the state of the art algorithm in MIR.

The paper is structured as follows: first related work will be discussed to create an image of the state of art. Next, a theoretical background about Gaussian Processes will be given, explaining the general form and the Multi-Output form. Then the details of the experiment will be mentioned, followed by the results. This paper is concluded with a discussion of the results and possible future research.

## Chapter 2

# Related Work

In general, not much research is done on emotion recognition using continuous values. Compared to genre recognition, emotion recognition still is in an early stage. In recent years, the amount of research on emotion recognition using contextual information and auditory content increased (Kim et al., 2010). The emotion recognition task can be split up in solely classifying emotion in music and defining emotion as continuous variables to be able to apply regression on it. Regression can be applied on the valence and arousal parameters to classify emotions (Yang et al., 2007). In their research, they compared the predictions made by a multiple linear regression, support vector regression and a third nonlinear regression algorithm. The SVR obtained the highest accuracies for both arousal and valence. Since SVR performs the best on emotion regression and genre classification tasks in general, it is frequently used as a baseline for comparison to new algorithms.

Additionally, Yang also helped to create the dataset Medieval 2013, containing auditory features and annotations for arousal and valence values for 1000 songs (Soleymani et al., 2013). Their motivation was to stimulate the studies on the relation between emotion and music. Although there is a significant difference in the quality of the regression on arousal compared to valence, the dataset has been used in multiple other researches. For example, the dataset is used to train MOODetector (Cardoso et al., 2011), a software application that generates playlists. The playlists are constructed using one or more seed songs, where the songs are represented by their valence and arousal values. The application creates high quality playlists, therefore if it could be scaled to a larger database of music and eventually all music, applications like MOODetector could be used for automatic personal playlist generation.

Markov and Matsui did research on both genre and emotion recognition, but they state that the main goal of music is to trigger emotions and obtaining emotional information about music is an important task (Markov & Matsui, 2014). In this research, the Medieval 2013 dataset is used for

the emotion recognition task as well. Markov compares the predictions of a SVR and a Gaussian Process regression model. The two algorithms perform roughly similarly, however, the GP performs better when the data contains more features.

Considering Gaussian Processes are relatively new in the domain of research in music and already equals the performances of the SVR, GPs are an interesting field to do more research on. A deep GP algorithm was tested on an emotion classification task (Chen, Lee, Hsieh, & Wang, 2015), where it performed better than a standard GP and the SVR. This shows that more complex applications of GP improves the quality even more.

## Chapter 3

# Theoretical Background

### 3.1 Gaussian Processes

In the book of Rasmussen and Williams, they introduced Gaussian Processes (GP) for machine learning tasks and since GPs have been applied to various domains (Williams & Rasmussen, 2006). They define a Gaussian Process as ‘a collection of random variables, any finite number of which have a joint Gaussian distribution.’ GPs are used to describe a distribution over functions and are completely specified by a mean and a covariance function. The mean function is defined as:

$$m(x) = E[f(x)]$$

and the covariance function as:

$$\text{cov}(x, x') = E[(f(x) - m(x))(f(x') - m(x'))]$$

where  $E[x]$  is the expected value of  $x$ . The GP is defined as:

$$f(x) \sim \mathcal{G}(m(x), \text{cov}(x, x'))$$

Often the mean is assumed to be zero for simplicity (Williams & Rasmussen, 2006). The covariance function returns the similarity between two inputs, which is computed by a kernel function. There exist many kernel functions, but the most commonly used is the squared exponential in GPs:

$$k(x, x') = \exp\left(-\frac{(x - x')^2}{2\ell}\right)$$

where  $\ell$  is the length-scale of the process. The covariances for all training inputs  $x_1 \dots x_n = X$  are represented by a covariance matrix  $K_{ij} = K(x_i, x_j)$ , which is of size  $n \times n$  and symmetric. The covariance matrix specifies the distribution of functions around  $m$  i.e.

$$f(X) \sim \mathcal{G}(m, K(X, X))$$

which can be used as the prior belief of a model.

It becomes more interesting when the prior distribution is adapted to fit the training data and to predict values of test inputs. If the observations are assumed to be noise-free, the joint distribution of the training inputs  $f(X)$  and the test outputs  $f(X_*)$  is specified as

$$\begin{bmatrix} f(X) \\ f(X_*) \end{bmatrix} = \mathcal{N} \left( 0, \begin{bmatrix} K(X, X) & K(X, X_*) \\ K(X_*, X) & K(X_*, X_*) \end{bmatrix} \right)$$

where  $K(X, X_*)$  is the matrix with covariances between all train and test inputs. The mean and covariance of the posterior are computed by conditioning the prior on the test input (Williams & Rasmussen, 2006), which results in:

$$\begin{aligned} m_* &= K(X_*, X)K(X, X)^{-1}f(X) \\ cov_* &= K(X_*, X_*) - K(X_*, X)K(X, X)^{-1}K(X, X_*) \end{aligned}$$

However it is highly unlikely that the observations are noise-free. To create a prior which takes noise into account, a Gaussian noise with variance  $\sigma_{n^2}$  is added to the output:

$$y(X) = f(X) + \sigma_{n^2}$$

The noise is independent for every input and zero if the input is noise-free. The kernel function is influenced by noise as well:

$$k(x, x') = \sigma_f \exp\left(-\frac{|x - x'|^2}{2\ell}\right)$$

where  $\sigma_f$  is the signal variance (Williams & Rasmussen, 2006).  $\sigma_f$ ,  $\sigma_{n^2}$  and  $\ell$  are called the free hyperparameters. These parameters are learned during training and adapted to fit a posterior given a test input. To get the parameters that fit the data the best, the log of the marginal likelihood should be minimized. The log of the marginal likelihood of a GP is given by (Williams & Rasmussen, 2006)

$$\begin{aligned} \log(p(y|X)) &= \log(\mathcal{G}(0, K(X, X) + \sigma_{n^2}I)) \\ &= \log\left(\frac{1}{(2\pi)^n(K(X, X) + \sigma_{n^2}I)} \exp\left(-\frac{1}{2}y^T(K(X, X) + \sigma_{n^2}I)^{-1}y\right)\right) \\ &= -\frac{1}{2}y^T(K(X, X) + \sigma_{n^2}I)^{-1}y - \frac{1}{2}\log((2\pi)^n(K(X, X) + \sigma_{n^2}I)) \end{aligned}$$

## 3.2 Multi-Output Gaussian Processes

A Gaussian Process  $f(X)$  can return only one output for every input. Multiple outputs can be predicted by several GPs, but this assumes that the outputs are independent. Multi-Output Gaussian Processes (MOGP) can

be trained to predict multiple dependent outputs using one set of hyperparameters. A MOGP is necessary to predict valence and arousal values with the assumption that they are dependent variables. The output of a MOGP for an input vector  $\mathbf{x}$  is a vector  $f(\mathbf{x})$  with one element for every desired output.

In MOGP models, an important distinction is made between the types of inputs the models receive (Alvarez et al., 2011). Isotopic models receive the same set of inputs for every output, while the outputs in heterotopic models can have different input sets. The implementation in this research will be an isotopic model, because all songs will have the same audio features and the MOGP uses the same features for both outputs.

The covariance of the outputs is defined by adding a third dimension to the covariance matrix  $K(X, X)$ . The dimension contains the covariances between all outputs for every input. The matrix is now defined as  $K(X, X)_{d,d'}$  with entries  $d, d' = 1 \dots D$ , where  $D$  represents the number of outputs (Alvarez et al., 2011). Then, for every pair of outputs  $d, d'$ , a covariance matrix  $K(x_i, x_j)_{d,d'}$  remains with entries  $i, j = 1 \dots N$ , where  $N$  represents the number of inputs. Therefore, the dimensions of the kernel are  $D \times D \times N \times N$ . To compute the covariance corresponding to  $K(x_i, x_j)_{d,d'}$ , a kernel function  $k(f_d(x_i), f_{d'}(x_j))$  is used, where  $f_d(x_i)$  is a function which computes output  $d$  given input  $x_i$ .

Multi-output kernels can be divided into separable kernels and non-separable kernels (Alvarez et al., 2011). Separable kernels are regarded as simpler, because the covariance of the output is separated from the input covariance kernel. The linear model of coregionalization (LMC) is the most common form of separable kernels. The LMC consists of the product of two kernels, one for the input space only and one kernel that describes the correlations between the outputs. In this case  $K(X, X)_{d,d'}$  can be written as  $k(X, X)\mathbf{B}$  where  $\mathbf{B}$  is the covariance matrix for the outputs. If the outputs are independent, the scalar kernel is given by Kronecker delta function  $\delta_{i,j}$ , which only returns 1 if  $i = j$  (Alvarez et al., 2011).

The separable kernels are often considered simpler because they can't capture correlations as good as non-separable kernels can, yet they are more convenient for computations (van der Wilk et al., 2020). Separable kernels assume that the covariance among outputs is constant for all inputs, however non-separable kernels can approximate an input-dependent correlation function between outputs as well. Non-separable kernels cover all classes outside of the domain of separable kernels, but the most general form is the Convolutional kernel (Alvarez et al., 2011). Non-separable kernels are still a very new direction of Gaussian Processes. The amount of literature on non-separable kernels is limited, as well as examples and implementations. Hence, a MOGP with a non-separable kernel will not be implemented in this research.

## Chapter 4

# Research

### 4.1 Data

The dataset used for this research is called “Medieval 2013” (Soleymani et al., 2013). 744 songs were collected from the Free Music Archive and feature extraction was applied on the songs using the openSMILE toolkit (Eyben, Weninger, Gross, & Schuller, 2013). Additionally, all songs were given annotations for valence and arousal by a number of annotators and the dataset includes detailed statistics like the mean, standard deviation and the range. These annotations are used to train the various models and to rate the models’ predictions. The dataset also includes detailed information about all the features for every timestamp. In this research, only the mean of the features over all the timestamps is used. Both the feature data and the annotations are normalized and a principal component analysis is applied to only include the features that explain 95% of the variance.

Although not used in similar research papers, the DEAM dataset can be used for the research problem addressed in this paper (Soleymani et al., 2016). The DEAM dataset also includes valence and arousal annotations, however the set of auditory features differs from the features of the Medieval 2013 dataset. DEAM contains 1744 songs, while Medieval 2013 has 744 songs. In the Medieval 2013 dataset, a large difference between the quality of arousal and valence annotations is present, which causes significant differences in prediction accuracies of the models. The Pearson correlations are 0.599 and 0.588 in the Medieval 2013 dataset and DEAM dataset respectively, therefore the arousal and valence values are correlated.

## 4.2 Methods

The algorithms are compared by the R-squared score of the models' predictions. The R-squared score is defined as:

$$R^2 = 1 - \frac{(y_i - \hat{y}_i)^2}{(y_i - \bar{y})^2}$$

where  $y_i$  is the true output for input  $i$ ,  $\hat{y}_i$  is the predicted output of input  $i$  and  $\bar{y}$  is the mean of the outputs (Markov & Matsui, 2014). The R-squared score can be interpreted as the squared error relative to the variance of the real outputs.

Both a SVR model and a GP model are used as the baseline. The SVR model uses a Radial Basis Function kernel. The GP model consists of two sets of hyperparameters, which are trained independently for the two outputs. The GP uses a similar kernel to the RBF kernel, which is the Squared Exponential kernel. The GP needs a significantly smaller amount of iterations during training than the SVR.

The MOGP is implemented with the separable coregionalization kernel. The inputs are labeled with an index to represent the corresponding output. The purpose of this functionality is to include heterotopic inputs, however the input used in this research is homotopic, therefore the complete input set is labeled with both outputs. For comparison with the SVR and GP, the RBF is used as well, however other kernels are used to see if the MOGP obtains consistent results across different kernels.

## 4.3 Results

Figure 4.1A shows the average R-squared scores obtained by the SVR, GP and MOGP over multiple cross-validations. The GP and MOGP obtain similar scores for predicting the arousal values, although the normal GP is slightly better in predicting valence values. Both the GP and MOGP perform better than the SVR. The standard deviations show that all models make equally constant predictions. The SVR only returns one prediction for every input, while the GPs return both a mean and a variance for each input. To be able to make a fair comparison, the R-squared scores are computed using the predicted mean values made by the GP and MOGP.

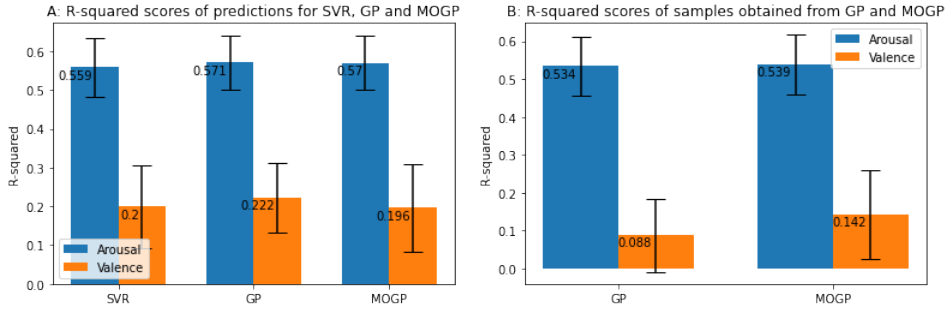


Figure 4.1: A: The R-squared scores for the SVR, GP and MOGP models. B: The R-squared scores of samples made by the GP and MOGP models. For both graphs, the scores are averaged over 10-fold cross-validations and the error bars show the standard deviation.

Nonetheless the standard deviations are interesting values to look at. Although the R-squared scores are slightly higher for the normal GP, the MOGP returns significantly lower standard deviations; 0.201 for arousal and 0.114 for valence, while the GP returns 0.523 for arousal and 0.135 for valence. This means that if the mean and variance of the MOGP are used to sample from, the samples will be more accurate than when the mean and variance of the GP are used, and in fact get a higher R-squared score than the samples obtained from the normal GP. Figure 4.1B shows the R-squared scores of the samples made by the GP and MOGP models. The MOGP samples scores slightly higher arousal, though there is a big increase in the score of valence with respect to the single-output GP.

Including a correlation between the outputs has a small effect on the performance of predictions. It is interesting to see if the same effect can be found in the DEAM dataset. Figure 4.2 shows the R-squared scores of the predictions made on the DEAM dataset. The MOGP and GP models perform almost identically, although they both obtain better scores than the SVR. Again, the average variance of the predictions is lower for the MOGP (0.037 for arousal and 0.029 for valence, while the GP obtains 0.074 for arousal and 0.049 for valence). The standard deviation of the predictions made on the DEAM dataset are substantially lower than the predictions of the Medieval 2013 dataset. Possible explanations are that the DEAM dataset contains roughly twice as many data points and that the large difference between the valence and arousal predictions in the Medieval 2013 dataset is not present in the DEAM dataset.

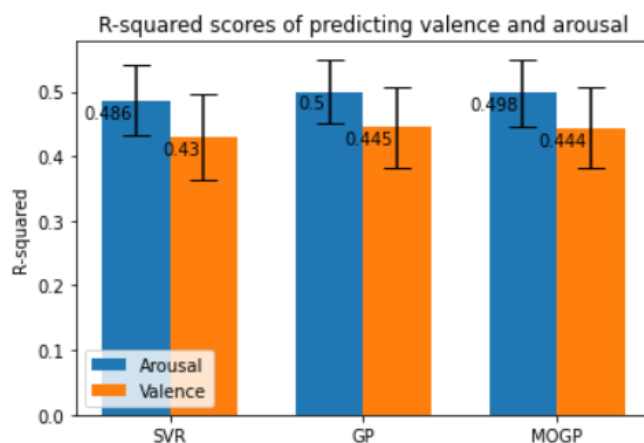


Figure 4.2: The R-squared scores for the SVR, GP and MOGP models on the DEAM dataset. The results are averaged over 10-fold cross-validation. The error bars show the standard deviation.

Even though the GP and MOGP obtain similar R-squared scores, both models have their preferred types of inputs. In figure 4.3, the scatterplots of both the Medieval 2013 and DEAM dataset can be seen. The input songs are located on their corresponding valence and arousal values. The colour differences represent the difference of the GP and MOGP in terms of the Euclidean distance from the predicted values to the true values. The MOGP is preferred for the input songs with lower arousal, while it performs worse for the songs with higher arousal values. The covariance matrix of the MOGP can be translated into a correlation matrix to see how the MOGP models the correlation. This correlation is 0.85 on average, which is substantially higher than the Pearson correlation for both datasets. That the resulting correlation is higher can be explained by the fact that MOGP expects to find a correlation, therefore it will find a higher correlation. As can be seen in figure 4.3, there are many songs for which the MOGP performs better than the GP, however there is only a small amount of songs where the opposite is true. This means that the covariance of the outputs has a positive effect on predicting a large fraction of the songs.

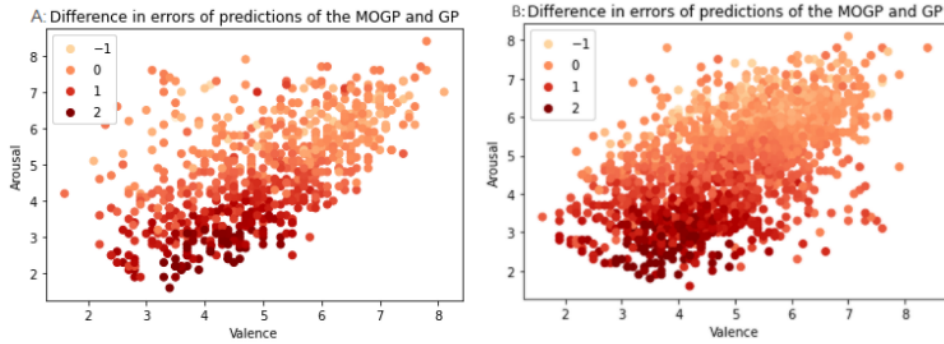


Figure 4.3: The input songs are located on their corresponding arousal and valence values. Arousal is represented on the y-axis and valence on the x-axis. The colour represents the difference in Euclidean distance from the prediction MOGP to the truth value compared to the distance for the GP. Figure 4A shows the Medieval 2013 dataset and 4B the DEAM dataset.

However, the improvement the MOGP shows relative to the GP only holds for the Squared Exponential kernel so far. In figure 4.4A and B the R-squared scores of the predictions of the MOGP and GP are compared using different kernels as the covariance function. As expected, the Linear kernel performs the worst for both the MOGP and GP, because it can only capture simple relations in the data. The Exponential kernel achieves the highest scores for arousal and valence when used in both the GP and MOGP. While there is no difference between the Squared Exponential kernel and the Matern32 kernel when used in the MOGP, the Matern32 kernel works better in the GP model. For all kernels, the GP obtains an equal or a higher R-squared score than the MOGP. In figure 4.4C and D the variances of the predictions are shown. In terms of variance of the predictions, only the Linear kernel and the Squared Exponential kernel benefit from including the covariance of the outputs, although this difference is insignificant for the Linear kernel. The variances of the Matern32 kernel and the Exponential kernel are lower for the GP than for the MOGP. Therefore the covariance of the output only has a positive effect on the predictions when the Squared Exponential kernel is used.

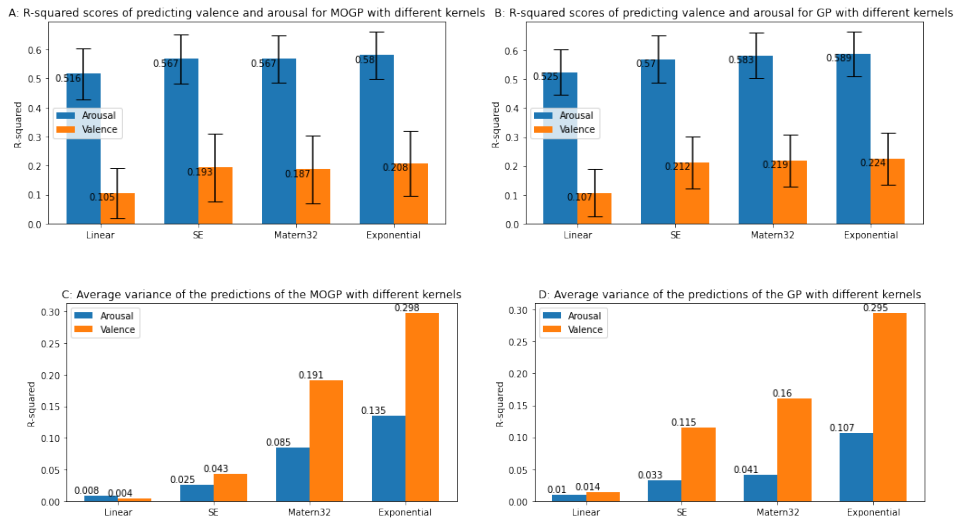


Figure 4.4: A: The R-squared scores of the predictions of valence and arousal of the MOGP using different kernels. B: The R-squared scores of the predictions of valence and arousal of the GP using different kernels. In A and B the standard deviation is shown by the error bar. The results are averaged over 10-cross validation. C: Variances of the predictions of the MOGP using different kernels. D: Variances of the predictions of the GP using different kernels.

## Chapter 5

# Discussion

In this paper, a Multi-Output Gaussian Processes model was compared to a single-output Gaussian Processes model at the task of predicting valence and arousal values. Multi-Output Gaussian Processes use an extra kernel to model the covariance between the two outputs. The effect of the covariance kernel on the quality of the model's predictions is limited, however the MOGP model produces more reliable samples due to a decrease of the standard deviation of the predictions.

The effect is most measured for songs with a lower arousal. A potential cause could be that in general songs with a higher arousal gradually increase the energy to a climax. In this research the input data was averaged over the whole songs, therefore the correlation between arousal and valence weakens for those songs. An extension of this research would be to use the timestamps of the auditory features as the input data. The Squared Exponential kernel is the only kernel for which the covariance of the outputs has a positive effect. However, more research could be done to generalize this effect for other regression tasks as well.

Non-separable kernels could even further increase the accuracy of the predictions. Although there is a limited amount of literature of non-separable kernels, some forms are already discussed and applied. The most general form constructed at the moment is the Convolutional kernel (Alvarez et al., 2011). The convolution kernel can be seen as a dynamic version of the linear model of coregionalization.

Additionally, the LMC itself can be extended to a Multi-Output form as well. The coregionalization matrix is adapted to be dependent on  $\mathbf{x}$ , therefore it allows the correlation to be explained by a function (Gelfand, Schmidt, Banerjee, & Sirmans, 2004). Other examples of non-separable multi-output kernels are Multi-Output Spectral Mixture Kernel (Parra & Tobar, 2017) and Invariant kernels (Alvarez et al., 2011). Implementations of these kernels on this paper's research problem would be an interesting next step, however structured code lacks for non-separable kernels.

A future application of the predictive models would be to generate playlists based on the valence and arousal values of songs. The only requirement is that all songs need to have the same auditory features. Generating playlists on the basis of emotional similarity can be an important improvement of personal playlists, since currently they are mostly based on artist similarity. It would improve the recommendations within genres, however it could also stimulate the discovery of new genres.

## Chapter 6

# Acknowledgements

In this research, an attempt was made to implement a Multi-Output Gaussian Process model using a non-separable kernel as well, specifically a convolutional kernel. However, because there is a limited amount of literature and a lack of examples and code on non-separable kernels, the implementation could not produce constant and comparable results. The MOGPTK toolkit was used to implement the convolutional kernel (de Wolff, Cuevas, & Tobar, 2020). The author of the MOGPTK toolkit also helped me with the implementation of the convolutional kernel.

# References

- Alvarez, M. A., Rosasco, L., & Lawrence, N. D. (2011). Kernels for vector-valued functions: A review. *arXiv preprint arXiv:1106.6251*.
- Ayata, D., Yaslan, Y., & Kamasak, M. E. (2018). Emotion based music recommendation system using wearable physiological sensors. *IEEE transactions on consumer electronics*, *64*(2), 196–203.
- Bestelmeyer, P. E., Kotz, S. A., & Belin, P. (2017). Effects of emotional valence and arousal on the voice perception network. *Social cognitive and affective neuroscience*, *12*(8), 1351–1358.
- Cardoso, L., Panda, R., & Paiva, R. P. (2011). Moodetector: A prototype software tool for mood-based playlist generation. In *Simposio de informatica-inforum* (Vol. 124).
- Casey, M. A., Veltkamp, R., Goto, M., Leman, M., Rhodes, C., & Slaney, M. (2008). Content-based music information retrieval: Current directions and future challenges. *Proceedings of the IEEE*, *96*(4), 668–696.
- Chen, S.-H., Lee, Y.-S., Hsieh, W.-C., & Wang, J.-C. (2015). Music emotion recognition using deep gaussian process. In *2015 asia-pacific signal and information processing association annual summit and conference (apsipa)* (pp. 495–498).
- de Wolff, T., Cuevas, A., & Tobar, F. (2020). Mogptk: The multi-output gaussian process toolkit. *arXiv preprint arXiv:2002.03471*.
- Eyben, F., Weninger, F., Gross, F., & Schuller, B. (2013). Recent developments in opensmile, the munich open-source multimedia feature extractor. In *Proceedings of the 21st acm international conference on multimedia* (p. 835–838). New York, NY, USA: Association for Computing Machinery. Retrieved from <https://doi.org/10.1145/2502081.2502224> doi: 10.1145/2502081.2502224
- Gelfand, A. E., Schmidt, A. M., Banerjee, S., & Sirmans, C. (2004). Nonstationary multivariate process modeling through spatially varying coregionalization. *Test*, *13*(2), 263–312.
- Giacaglia, G. (2019, March). *Behind spotify recommendation engine*. Data Driven Investor. Retrieved from <https://medium.com/datadriveninvestor/behind-spotify-recommendation-engine-a9b5a27a935>

- Haggblade, M., Hong, Y., & Kao, K. (2011). Music genre classification. *Department of Computer Science, Stanford University*, 131, 132.
- Kim, Y. E., Schmidt, E. M., Migneco, R., Morton, B. G., Richardson, P., Scott, J., . . . Turnbull, D. (2010). Music emotion recognition: A state of the art review. In *Proc. ismir* (Vol. 86, pp. 937–952).
- Markov, K., & Matsui, T. (2014). Music genre and emotion recognition using gaussian processes. *IEEE access*, 2, 688–697.
- Parra, G., & Tobar, F. (2017). Spectral mixture kernels for multi-output gaussian processes. *Advances in Neural Information Processing Systems*, 30, 6681–6690.
- Platt, J., Burges, C. J., Swenson, S., Weare, C., & Zheng, A. (2001). Learning a gaussian process prior for automatically generating music playlists. *Advances in neural information processing systems*, 14, 1425–1432.
- Soleymani, M., Aljanaki, A., & Yang, Y. (2016). *Deam: Mediaeval database for emotional analysis in music*. Geneva, Switzerland.
- Soleymani, M., Caro, M. N., Schmidt, E. M., Sha, C.-Y., & Yang, Y.-H. (2013). 1000 songs for emotional analysis of music. In *Proceedings of the 2nd acm international workshop on crowdsourcing for multimedia* (pp. 1–6).
- Su, X., & Khoshgoftaar, T. M. (2009). A survey of collaborative filtering techniques. *Advances in artificial intelligence*, 2009.
- van der Wilk, M., Dutordoir, V., John, S., Artemev, A., Adam, V., & Hensman, J. (2020). A framework for interdomain and multioutput gaussian processes. *arXiv preprint arXiv:2003.01115*.
- Williams, C. K., & Rasmussen, C. E. (2006). *Gaussian processes for machine learning* (Vol. 2) (No. 3). MIT press Cambridge, MA.
- Yang, Y.-H., Lin, Y.-C., Su, Y.-F., & Chen, H. H. (2007). Music emotion classification: A regression approach. In *2007 IEEE international conference on multimedia and expo* (pp. 208–211).