CS4246 Project 1
Depression Prediction

Team 01
Antoine Charles Vincent Garcia - A0159072A
Chan Jun Wei - A0112084
Chen Tze Cheng - A0112092
Eric Ewe Yow Choong - A0112204
Han Liang Wee, Eric - A0065517
Ho Wei Li - A0094679

Abstract

Depression is a worrying issue in modern times. If left unregulated, it can be dentrimental both health and life.

In this report, we illustrate the use of Gaussian Processes (GP) to calculate and model stress levels in society and with the data obtained, is used to estimate depression severity.

1. Introduction

For our experiment, we will use the GP model to measure and compute depression severity via audio recordings. We will also be discussing about the desirable properties of the GP model as well as the technical details such as the GP model requirements for the proposed application and modifications made to enhance performance. We will also include our experimental evaluations and procedure in this report.

1.1 Motivation and Hypothesis

The rationale of depression prediction is enable authorities to take appropriate actions if an area or individual is depressed. For example, suicide and crime are often linked to high depression and stress levels. The data can help authorities to monitor and mitigate crime in areas with marked as 'depressed'. In addition, annual health checks may include psychiatrist recommendations which is given to individuals who falls into the depression category.

1.2 Future Applications

If successful, the data collected can be contributed to the implementation of a depression calculator application. Instead of undergoing the standard Patient Health Questionnaire depression scale (PHQ-8), users will be able to check their depression scale objectively with the app. At dangerously high levels of depression, help and consultation can be issued via notification from the app itself.

2. Gaussian Process Regression Model

As all individuals have varying inherent stress management, the use of the GP model for depression prediction makes use of all samples and feature information to perform the prediction including training data with different or uneven sampling rates. From the mean and variance obtained from previous data, we are able to predict if an individual is depressed.

2.1 Qualitative Advantages

...

2.2 Important Requirements

Our GP model requires multiple audio recordings of an individual's speech.

2.3 Energy The Energy feature of a sound refers to the loudness of the sound at various timeframes, hence it is obvious that energy of a sound is directly proportional to the amplitude of the soundwave. This shows that the higher the energy, the louder the sound is going to be.

2.4 Mel Frequency Cepstral Coefficients (MFCC) : Mel Frequency Cepstral (MFC) is a representation of short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear Mel scale of frequency. MFCC are the coefficients that forms the MFC. The greatest benefit of using MFCC is that the scale approximates to the human's auditory system response more closely, hence it allows for a better representation of sound. In order to obtain MFCC, Fourier transformed is performed on the sound signals.

2.5 Magnitude Spectrum : Magnitude spectrum can be produced by converting the input signal of an audio into frames. Fast Fourier Transform (FFT) is performed on each frames and this will form the Magnitude Spectrum.

2.6 Zero-Crossing Rate : Zero crossing rate is the rate of sign-changes along a signal. This feature is extremely useful in speech recognition and music information retrieval.

2.7 Other Machine Learning Methods

Machine learning is about using model to learn from existing data for some improvement or prediction. The following is some methods used in machine learning:

2.8 k-nearest neighbors algorithm : The k-Nearest Neighbors algorithm (or k-NN for short) is a non-parametric method used for classification and regression. K is the user defined constant which classify the class of a vector data. The training examples are vectors in a multidimensional feature space, each with a class label. The classification process of this algorithm can be visualize as the graph below.

2.9 Support Vector Machine (SVM) : Support Vector Machine (SVM), with the help of the libsvm, a developed library for SVM. SVM is a type of linear classiers that aims at nding a unique solution in the form of an optimal hyperplane. This optimal hyperplane is dened as the one that maximizes the margins between the two classes. It will be positioned at equal distance between the closest points of each class and will create the largest possible corridor with no points from either classes inside. These closest points are on the border of the corridor and are called the support vectors.

2.10 Random Forest Regressor : Random Forest is an ensemble of decision trees, whereas a decision tree is trying to separate the data into different leaf nodes where the data points in each node have certain similarity. Each decision tree will predict a value and all of the values will be averaged. The average value would be the prediction result.

2.11 AdaBoost : AdaBoost is the short form of Adaptive Boost. The output of the other learning algorithms ('weak learners') is combined into a weighted sum that represents the final output of the boosted classifier. AdaBoost is adaptive in the sense that subsequent weak learners are tweaked in favor of those instances misclassified by previous classifiers. AdaBoost first find out the classifier with the least errors, then focus on those errors later by adding the weight to the outliers, then find out another classifier with least errors again. The process is repeated till the end then all the classifier will be combined to form a final classifier which will indicate the class of data. Therefore, AdaBoost is able to be sensitive to noisy data and outliers.

2.12 Naive Bayes : Naive Bayes is a prediction algorithm that uses Bayes rule in assumption of all naive Bayes classifiers assume that the value of a particular feature is independent of the value of any other feature, given the class variable. Naive Bayes multiplies all the possibilities of a particular event occurs, then give the prediction of possibility that event. All the data are extracted from the dataset to represent the possibilities which are the number of times the particular event occurs in the dataset over the total number of events. Naive Bayes can produce great results if the features in the dataset is as independent as possible.

3. Technical Approach

Application of Gaussian Process model

Using Gaussian Process model as a machine learning model: When a person is depressed, no matter what time the person talks, we should also be able to determine the person is depressed according to the speech signal. And we assume that the speech signals extracted from different depreessed people should be similar and thus is suitable to use Gaussian Process Model.

Some Famous Mathematical Formula

Euclidean Distance Euclidean Distance is used to find the distance between two points in an Euclidean Space. Euclidean Distance is defined as:

$$||x-x'|| = \sqrt{(x_1 - x_1')^2 + (x_2 - x_2')^2 + ... + (x_n - x_n')^2} \quad (1)$$

Squared Euclidean Distance It is just the squared of Euclidean Distance. Nothing special.

$$||x-x'||^2 = (x_1-x_1')^2+(x_2-x_2')^2+...+(x_n-x_n')^2 \quad (2)$$

Kernels

A general name for a function k of two arguments mapping a pair of inputs $x\epsilon X, x'\epsilon X$ into R is a kernel. In other word, the similarity measure of all features is usually called kernel. This term arises in the theory of integral operators, where the operator T_k is defined as

$$(T_k f)(x) = \int_X k(x,x')f(x')d\mu(x'), \quad (3)$$

where μ denotes a measure. A real kernel is said to be symmetric if k(x,x') = k(x',x); clearly covariance functions must be symmetric from the definition.

For a kernel to be used in Gaussian Process, one kernel must be positive semidenite:
A real n x n matrix K which satisfies

$$\text{Q(v)} = v^T K v \geq 0, \forall v \epsilon R^n \quad (4)$$

is called positive semidefinite (PSD). If Q(v) = 0 only when v = 0 the matrix is positive definite. Q(v) is called a quadratic form. A symmetric matrix is PSD if and only if all of its eigenvalues are non negative. A Gram matrix corresponding to a covariance function is PSD.

A kernel is said to be positive semidefinite if

$$\int \text{k(x,x')f(x)f(x')d}\mu\text{(x) d}\mu\text{(x')} \geq 0, \forall f \epsilon L_2(\text{X},\mu) \quad (5)$$

Equivalently a kernel function which gives rise to PSD Gram matrices for any choice of n N and D is positive semidefinite.

The kernels that would be chosen as our kernel is shown below (The kernels are assumed to be defined on two samples $x = (x_1x_2x_3...x_n)$ and $x' = (x_1'x_2'x_3'...x_n')$, represented as feature vectors in some input space):

1. **Radial Basis Function (RBF)**
The RBF kernel is defined as

$$K(x, x') = \exp^{-\frac{||x-x'||^2}{2\sigma^2}} \quad (6)$$

$||x - x'||^2$ represents Squared Euclidean Distance And $\sigma > 0$ can either be a scalar (isotropic variant of the kernel) or a vector with the same number of dimensions as the inputs X (anisotropic variant of the kernel).
It is also known as the squared exponential kernel. This kernel is infinitely differentiable, which implies that GPs with this kernel as covariance function have mean square derivatives of all orders, and are thus very smooth.

2. **Matern**
The class of Matern kernels is a generalization of the RBF and the absolute exponential kernel parameterized by an additional parameter nu. The smaller nu, the less smooth the approximated function is. For nu=∞, the kernel becomes equivalent to the RBF kernel and for nu=0.5 to the absolute exponential kernel. Important intermediate values are nu=1.5 (once differentiable functions) and nu=2.5 (twice differentiable functions).

3. **Dot Product**
The Dot Product kernel is non-stationary and can be obtained from linear regression by putting N(0, 1) priors on the coefficients of x_d (d = 1, . . . , D) and a prior of N(0, σ_0^2) on the bias. The Dot Product kernel is invariant to a rotation of the coordinates about the origin, but not translations. It is parameterized by a parameter σ_0^2. For $\sigma_0^2 = 0$, the kernel is called the homogeneous linear kernel, otherwise it is inhomogeneous. The kernel is given by

$$K(x, x') = \sigma_0^2 + x \cdot x' \quad (7)$$

4. **Constant Kernel**
The constant Kernel just set the similar measures to be a constant. The kernel is given by:

$$k(x, x') = C \quad (8)$$

where C is a constant, $C \epsilon Rand C \geq 0$

5. **Compound Kernel**
Kernel which is composed of a set of other kernels.

6. **Exp-Sine-Squared Kernel**
The ExpSineSquared kernel allows modeling periodic functions. Therefore, it is also called Periodic Kernel. It is parameterized by a length-scale parameter, $l > 0$ and a periodicity parameter $p > 0$. The kernel given by:

$$k(x, x') = \exp^{(\frac{-2sin(\frac{\pi||x-x'||}{p})}{l})^2} \quad (9)$$

where $||x - x'||$ represents Euclidean Distance.

7. **Rational Quadratic Kernel**
The RationalQuadratic kernel can be seen as a scale mixture (an infinite sum) of RBF kernels with different characteristic length-scales. It is parameterized by a length-scale parameter $l > 0$ and a scale mixture parameter $\alpha > 0$. The kernel given by:

$$k(x, x') = (\frac{1 + ||x - x'||^2)}{2\alpha l^2})^{-\alpha} \quad (10)$$

where $||x - x'||^2$ represents Squared Euclidean Distance

8. **Linear Kernel**
Linear Kernel is a simpler kernel which can run efficiently. However, its linearity assumption is its downside. The kernel is given by:

$$k(x, x') = x^T x' \quad (11)$$

9. **White Kernel**
The kernel is used to estimate the noise-level(δ) of the input. The kernel is given by:

$$if x == x', k(x, x') = \delta; \text{else } 0 \quad (12)$$

10. **OrnsteinUhlenbeck Kernel**
The kernel is given by:

$$k(x, x') = \exp^{(\frac{-||x-x'||}{l})} \quad (13)$$

where $||x - x'||$ represents Euclidean Distance.

11. **Sum**
Sum-kernel k1 + k2 of two kernels k1 and k2.

Novel Modifications

Using Gaussian Process model to optimize the parameters of the ensemble of different maching learning model: The outcome of different machine learning model should be similar and thus is suitable to put inside Gaussian Process model.

4. Evaluation

In order to test our Gaussian Process model, we conducted tests on data obtained from Audio/Visual Emotion Challenge and Workshop(AVEC 2016). The goal of AEVC is to weigh-in on the various approaches(visual, audio) used to recognize emotions under unambiguous conditions. AVEC 2016 provided 2 pieces of data as input: visual and auditory data. However, we would be reducing the scope of the experiment, limiting the experiment to only the auditory data. Two Sub-Challenges are lised in AVEC 2016. We are only interested in the Depression Classification Sub-Challenge, which requires participants to classify inputs by the PHQ-8 score.

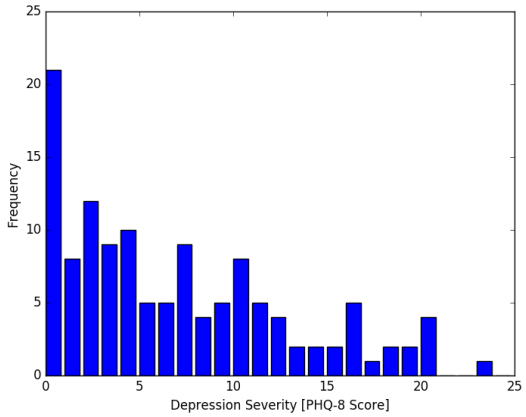

Figure 1: PHQ-8 scores' histogram of both training and development set

	Training	Development	All
n	95	31	126
μ	6.326	7.548	6.626
σ	5.597	6.690	5.909

Table 1: Summary of Datasets provided

4.1 Data

The depression data used in AVEC 2016 was obtained from the benchmarking database, the Distress Analysis Interview Corpus - Wizard of Oz(DAIC-WOZ). Data collected from DAIC-WOZ include audio and video recordings and the corresponsing PHQ-8 score[CITE:27](0-24), which is a frequently used self-report scheme to access severity of depression[CITE]. Henceforth, we would need to pre-process the auditory data before we use it in our Gaussian Process Model. The data is pre-processed as described in the Section [REF]. The distribution of the depression severity scores in both training and development set is given in Figure 1. The data provided are split into 2 sets: training and development. A summary of the data is given in Table 1.

4.2 Measure of Accuracy

AVEC 2016 provided a baseline classifier that consistently predicts the PHQ-8 score with RMSE $= 6.7418$[CITE]. In order to provide a meaningful and consistent comparison to the baseline provided, we would be only using Root Mean Square Deviation Error(RMSE) to measure the error rate on both Training and Development datasets. RMSE(Equation 14) is a commonly used in machine learning communities to measure the differences between the values predicted by a model and the values actually observed.

$$\text{RMSE} = \sqrt{\frac{\sum_{t=1}^{n}(\hat{y}_t - y_t)^2}{n}} \tag{14}$$

[CITEDBLP:journals/corr/ValstarGSRLTSSC16]

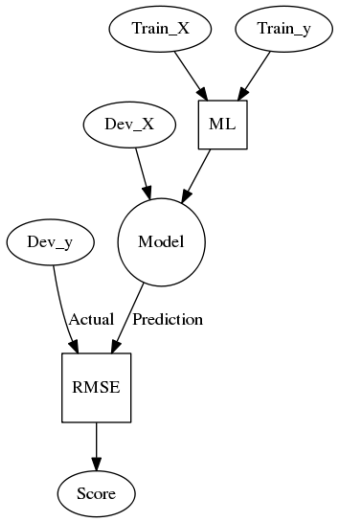

Figure 2: Experimental process

Algorithm	Hyper-parameters
K-Nearest Neighbors	x
Linear SVM	x
RBF SVM	x
Decision Tree	x
Random Forest	x
AdaBoost	x
Naive Bayes	x
Decision Tree	x

Table 2: List of Machine Learning Algorithms with their corresponding hyper-parameters

4.3 Experimental Setup

We compared our Gaussian Model against commonly used machine learning algorithms. The list of algorithms and their hyperparameters are given in Table 2. The hyper-parameters are either determined by the defaults used in the popular machine learning library, Scikit Learn[CITE] or some reasonable values were used. Each machine learning algorithm is trained against the training set and thereafter tested against the development set using RMSE as the error metric. The process used is shown in Figure 2.

4.4 Results

The results of the experiment is shown in t

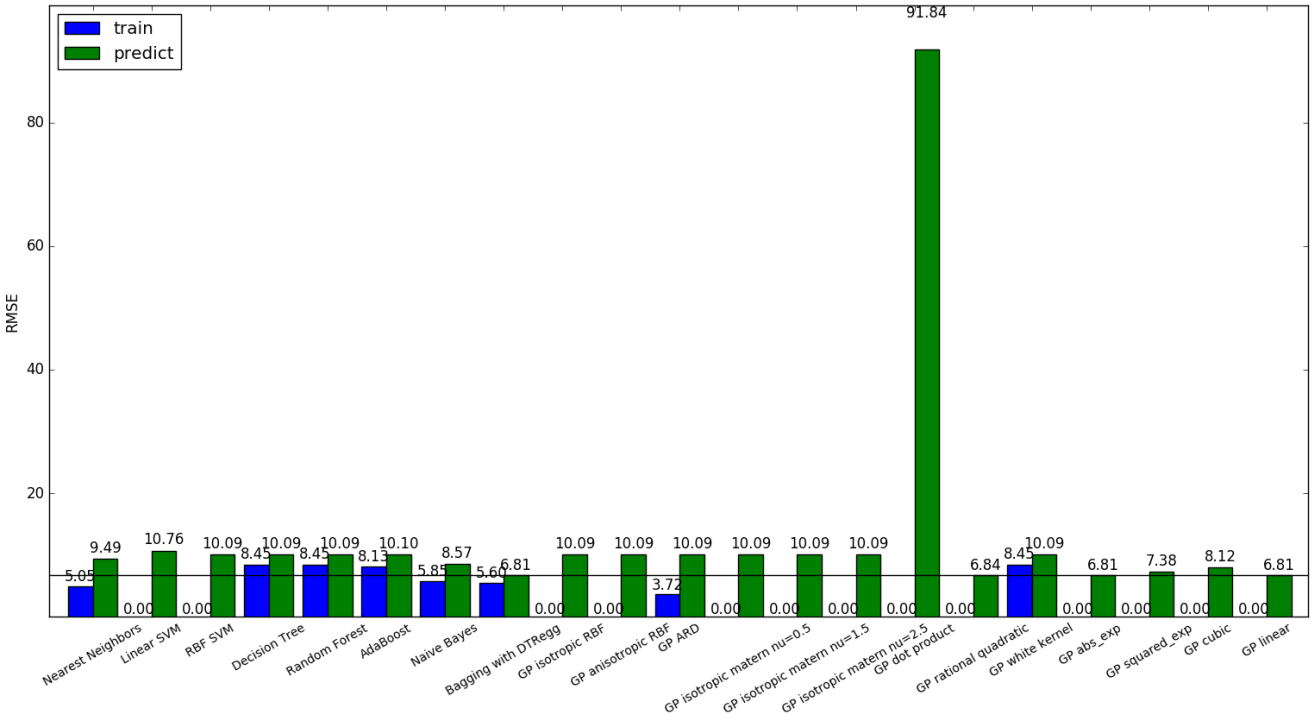

Figure 3: Chart showing RMSE(Training and Development) for the different classifiers

	RMSE	
Algorithm	Training	Development
K-Nearest Neighbors	x	x
Linear SVM	x	x
RBF SVM	x	x
Decision Tree	x	x
Random Forest	x	x
AdaBoost	x	x
Naive Bayes	x	x
Decision Tree	x	x
Gaussian Process	x	x

Table 3: RMSE results of the different machine learning algorithms

5. Conclusion

...

6. Main Roles of Each Member

- **Antoine Charles Vincent Garcia**: Scripting the program, setting up machine learning libraries and running tests.
- **Chan Jun Wei**: Project technicalities such as problem formulation and modelling, mathematics and experiment planning.
- **Chen Tze Cheng**: Project technicalities such as problem formulation and modelling, mathematics and experiment planning.
- **Eric Ewe Yow Choong**: Documentation especially writing of the motivation, recording research findings and keeping track of requirements.
- **Han Liang Wee, Eric**: Scripting the program, setting up machine learning libraries and running tests.
- **Ho Wei Li**: Documentation especially writing up the motivation, recording research findings and keeping track of requirements.