Project Definition:
Comparative Analysis of Regression Models for Personal Finance Predictions

CSI 4106 – Introduction to Artificial Intelligence
Fall 2018
Project Coordinator: Yazan Otoum
Date: November 5, 2018

Group Number: 22
Student Name: Dominic Roy-Stang
Student Number: 7483706

Project Report Information

Project Title: Comparative analysis of regression methods for personal finance predictions.
Project Type: Type 2 - Comparative analysis of 2 (or 3) approaches of problem solving.

Description of Chosen Topic and Its Link to Artificial Intelligence

The objective of this research project is to evaluate the feasibility of predicting an individual's personal finances based on past results with a greater accuracy than a simple linear regression. This sort of result could, in practice, be used to help people predict their retirement date, which would in turn, help them assess if they are living within their means.

To accomplish this, I will first start by establishing a linear regression predictor as a baseline because of its simplicity. Then, I will generate two separate models: a *Support Vector Regression (with RBF kernel)*, and a *Gaussian Process Regression (with Exp-Sine-Squared kernel)*. These two predictors will first be compared to the linear regression to see if more complex models can outperform a simple linear regression on this specific task. Then, the three models will be compared to each other to be ranked based on their performance on the given data set.

Support Vector Regression (SVR) was selected after reading this paper from the University of New Mexico, which found that SVR outperformed other algorithms for stock market predictions. It will be interesting to see if the same applies to personal finance predictions.

Gaussian Process Regression (GPR) was selected after reading its scikit-learn documentation. Its properties seemed to fit the problem at hand, given my relatively small data set. Additionally, I was unable to find a comparison of SVR and GPR in a financial prediction context, so I saw this as a good opportunity to contribute new research to the field of machine learning.

Members of The Team and Their Role

As the sole member of this team, I (Dominic Roy-Stang) will be working on every task related to this project on my own. An overview of the tasks is listed in the section below.

Planned Tasks and Time Estimates

Task 1: **Environment setup**
Time estimate: 2 hours
How it will be achieved:

- Create code repository
- Install Python 3.7 and a linter for VScode
- Install libraries in a Python virtualenv: Numpy, Matplotlib, Scikit-learn, Pandas

Task 2: **Importing and formatting a dataset**
Time estimate: 2 hours
How it will be achieved:

- Link all bank account information to Mint™
- Download all historical transactions as CSV
- Store the data from the CSV in a structure in the code and remove features that are just noise
- Calculate a column (feature) for "total net worth" after each transaction

Task 3: **Visualizing data and splitting it into training and validation sets**
Time estimate: 2 hours
How it will be achieved:

- Using matplotlib to show the data points (date, net worth)

- Using train_test_split scikit-learn to split the sets

Task 4: **Creating a basic linear regression and visualizing it**
Time estimate: 3 hours
How it will be achieved:

- Using the scikit-learn linear regression estimator
- Using matplotlib to show the data points and predicted (linear regression) values

Task 5: **Creating a basic Support Vector Regression (with RBF kernel)**
Time estimate: 3 hours
How it will be achieved:

- Using the scikit-learn linear regression estimator
- Using matplotlib to show the data points and predicted (linear regression) values

Task 6: **Creating Gaussian Process Regression (with Exp-Sine-Squared kernel)**
Time estimate: 3 hours
How it will be achieved:

- Using the scikit-learn linear regression estimator
- Using matplotlib to show the data points and predicted (linear regression) values

Task 7: **Tweaking parameters in an attempt to improve results**
Time estimate: 3 hours
How it will be achieved:

- Looking at scikit-learn documentation and tweaking function parameters

Task 8: **Code cleanup, unplanned errors, and front-end**
Time estimate: 6 hours
How it will be achieved:

- I am estimating approximately 30% of the development time will be spent on unplanned issue fixing, as this will be my first personal machine learning project.
- If I have some additional time, I will be building a nicer web-based (localhost) front-end to upload and display data. Otherwise, I will rely on matplotlib diagrams generated at run-time.

Task 9: **Report writing**
Time estimate: 8 hours
How it will be achieved:

- Gather screenshots of data
- Display and discuss results in accordance to the type 2 report format

Total time: 32 hours