Mini-Project Report

Comparison between Linear Regression, Naïve Bayes and Neuron Network on heart rate data

by

Phuong Nguyen

Mohamed

***Objectives:***

The objective of this project is to identify which of three different learning approaches has the best performance in predicting the health status of people based on heart rate data. The learning approaches considered were: Naive Bayes, Linear Regression (adding L1 and L2 regularization), Logistic Regression and Neural Networks with a single hidden layer. The criteria to define best performance in terms of 1) accuracy and 2) required runtime.

***Introduction:***

The heart rate data includes 2 features, that are Voltage and distance of heart. The output contains 3 columns involving of normal person, bad person who is recommended to take treatment, and very bad person who needs to get emergency treatment. The report describes 4 approaches including of Naive Bayes, Linear Regression (adding L1 and L2 regularization), Logistic Regression and Neural Networks with a single hidden layer. The reason linear regression which is more suitable for regression problems implemented is that this approach is used as sanity check. Other algorithms are expected to obtain higher accuracy than linear regression. The report also describes how grid search and cross validation with k-folds are used to tune parameters. The graph results are presented for visualization. At the end of the report, some conclusions are presented.

***Methodology:***

In this part, 4 approaches are described in details.

Multinomial regression:

<https://scikit-learn.org/stable/auto_examples/linear_model/plot_logistic_multinomial.html#sphx-glr-auto-examples-linear-model-plot-logistic-multinomial-py>

<https://scikit-learn.org/stable/auto_examples/linear_model/plot_sparse_logistic_regression_mnist.html>

Naïve Bayes:

<https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html#sklearn.naive_bayes.MultinomialNB>

Gaussian: <https://scikit-learn.org/stable/modules/naive_bayes.html>

Neural network:

<https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPRegressor.html#sklearn.neural_network.MLPRegressor>

<https://scikit-learn.org/stable/modules/neural_networks_supervised.html#classification>

Linear:

<https://scikit-learn.org/stable/modules/linear_model.html#ordinary-least-squares>