MINI PROJECT PROPOSAL

Xporters-Analysis of the Traffic Data

Eva Agrawal[a], Laetitia Clerc[b], Jean-François Gassie[b], Robin de Groot[a], and Hongyi Luo[a]

[a]EIT Data Science M1, Paris-Saclay, FR; [b]L2 Informatique, Paris-Saclay

ARTICLE HISTORY
Compiled March 15, 2020

Team name/project: VELO Xporters
Challenge URL: `https://codalab.lri.fr/competitions/652`
Github repository of the project: `https://github.com/jf-pnj/velo/tree/master/starting_kit`

1. Background and motivation

The project that this team will be working on has to do with traffic data. The provided document of this challenge has a brief story about a young man. He has a lemonade stand next to a busy road. In order to know how much lemonade he should have in stock and when he should man the stand, it is beneficial to know how many cars will come by in a certain time frame, since more cars equals more potential customers.

Since tabular data is very widespread in many organisations, we thought it would be wise to familiarise ourselves further with this kind of data.

2. Data and problem description

Now a more concrete and formal explanation. We have decided to work on the 'Xporters' project. It deals with predicting the traffic volume on a highway on a per hour, per day basis. We also study effect of additional meteorological factors such as rain, temperature etc. making to a total of 58 features to characterise the data. The target variable is total number of vehicles passing through the highway at an hour in the day.

The Xporters challenge is based on a small standard regression data set from the UCI Machine Learning Repository [1], formatted in the AutoML format. It uses a data set concerning the traffic volume off a highway in the USA from 2012 to 2018 [2].

CONTACT Eva Agrawal. Email: eva.agrawal@u-psud.fr
CONTACT Laetitia Clerc. Email: laetitia.clerc@u-psud.fr
CONTACT Jean-François Gassie. Email:jean-francois.gassie@u-psud.fr
CONTACT Robin de Groot. Email: robin.de-groot@u-psud.fr
CONTACT Hongyi Luo. Email: hongyi.luo@u-psud.fr

3. Approach chosen

The goal is thus to find and train an algorithm that predicts the traffic volumes as accurately as possible.

The purpose of the preprocessing part is to reduce the weight of the data. Why?For making the classification part more efficient and faster. We first want to delete the lines which are outliers using Isolation forest. Isolation forest will create a table where every outliers will be labeled -1. We then can easily wipe them out . Then we also want to reduce the numbers of dimension using the Principal component analysis (PCA) by deleting the data with the minus influence upon the results.We used Data Manager because it is more convenient as it split automatically the data.

In order to find the best algorithm, we will try out some different ones. Since there is such a large number of different target, or traffic volume, values, classification does not make sense for this task. We will instead try different regression algorithms from the scikit-learn library [3]. In order to provide better testing for each algorithms, we will use cross-validation, of which a very basic example is shown in figure 1. This method trains the algorithms multiple times with different train/test splits, and thus giving a more balanced and significant result. We will also use the same method for finding the optimal hyperparameters, though we will also use random search for the hyperparameter search [4].

Although these more conventional regression algorithms have the focus for this assignment. We also want to experiment with Neural Network-based regression if time allows it. These algorithms do not need an introduction anymore as their applications are widespread and their potential is well-known.

Thirdly, and finally, if time allows it, we would like to experiment with time series algorithms. The data set is obviously time series based since it is about hours in a day, so it should be easy to transform this data into something that algorithms like ARIMA can work with. However, this might take too much time for this project and we are not sure if its feasible, still we hope to be able to experiment with this.

After the implementation of various regression model next step is to interpret the results and explore the effect of independent variables from the data on the traffic volume. This exploratory data analysis will help us to test the models applied and spot any anomalies and discover patterns in the data. The graphical representations used will be Pearson correlation matrix, scatter plots . Through the correlation matrix values we will identify the features which have the most effect on the target values. According to this exploratory analysis the models can be improved significantly. The scatter plot for the performance of model on sample test data is shown in Figure 2. Since the model has not been updated, the performance is not good as seen in the scatter plot [8].

In the Figure 3, confusion matrix [9] of different features and their effects are shown by which we can see the features which have the most impact on the target. However, the clustering techniques usually applied do not seem to work well on this data as this is a regression based problem. We implemented the K-means classifier [10] but did not obtain much meaningful results Figure 4. Here target values are clustered on the holiday feature of the data. We can see that the most of the traffic is concentrated on weekdays.

The box plots in the original code are also a good representation of traffic volume per per hour and per hour. Once all the three parts are combined, we expect to get better results than the ones currently available on our Github repository and will be presented in the final report.

4. Brief description of the classes

The project work has been divided into three parts, namely: Pre-processing, Model & Visualization carried out in groups of two each.

(1) Pre-Processing: The binome will create a function to erase the outliers. The PCA method will be use to reduce the dimensions of the data.
(2) Model: This binôme will work on selecting the most optimal model for this regression task. The data for this binôme will have been prepared by the pre-processing binôme. The product will be a single class that takes in the data from the pre-processing, trains the model, and produces results.
(3) Visualization: The binôme will work on presenting the results of regression models applied. The typical visualisation techniques used will be scatter plots, confusion matrices, and histograms to show the effect of features on the traffic volume.

5. Preliminary results

In our GitHub group repository, the code for these preliminary results can be found. As of yet, we have not combined the work of the different binômes into one notebook however, so the results and visualisations are more of an idea of what the results will look like than that they are trustworthy.

The results show that the performance of different models varies widely. This will be further looked into, especially with some hyperparameter optimisation.

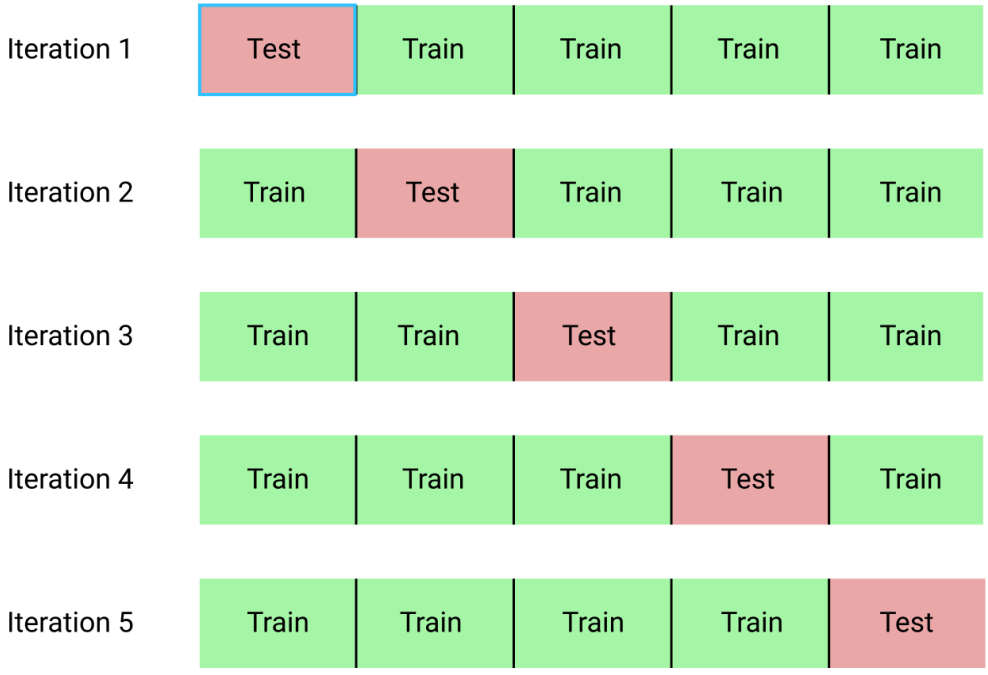

Figure 1. *KFold Cross Validation [5]*

Model	CV score
Random Forest Regressor	**0,95**
Support Vector Regression	0
Linear Regression	-0,07
K Nearest Neighbor Regression	0,77
Gradient Boosting Regression	0,92
Multilayer Perceptron Regression	0,41
AdaBoost Regression	0,82
Stacking (RF and GBT)	**0,95**

Table 1. Preliminary Cross Validation results. Closer to 1 is better. Metric used is R^2

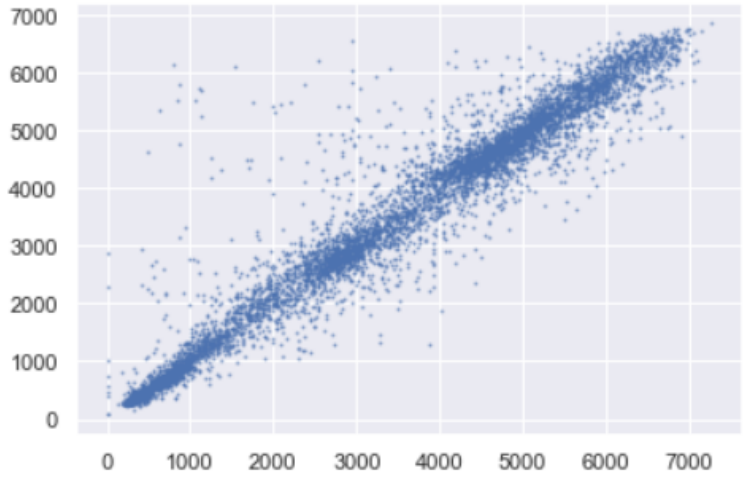

Figure 2. *Scatter plot [8]*

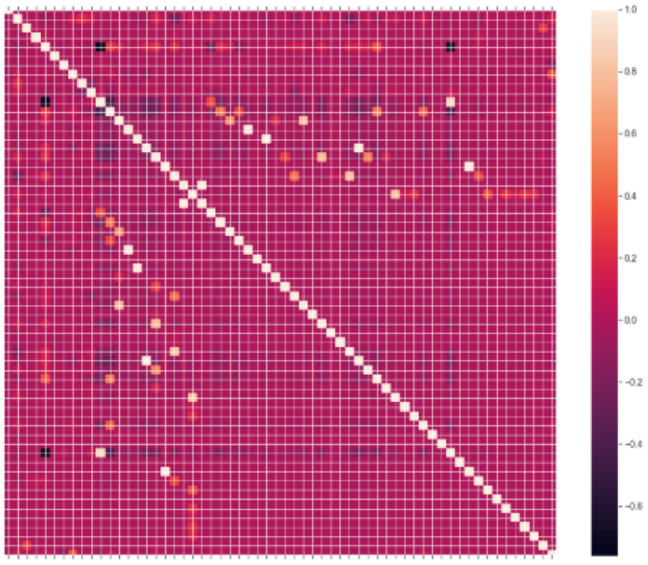

Figure 3. *Confusion Matrix [9]*

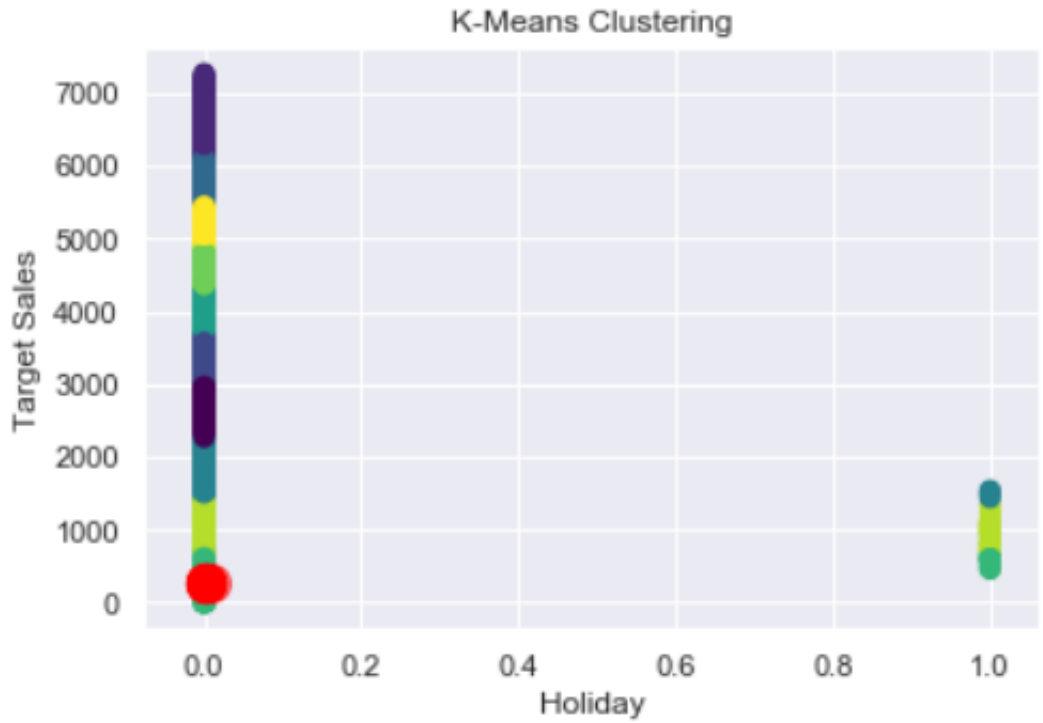

Figure 4. ***K-Means [10]***

6. References

References

[1] Dua, D. and Graff, C. (2019). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

[2] README.ipynb from the Velo starting kit

[3] Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.

[4] Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. Journal of machine learning research, 13(Feb), 281-305.

[5] Shaikh, Raheel. "Cross Validation Explained: Evaluating Estimator Performance." Medium, Towards Data Science, 26 Nov. 2018, towardsdatascience.com/cross-validation-explained-evaluating-estimator-performance-e51e5430ff85.

[6] Scikit-Learn Outlier Detection: `https://scikit-learn.org/stable/modules/outlier_detection.html`

[7] Scikit-Learn PCA:`https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html`

[8] Scikit-Learn Scatter Plot: `https://matplotlib.org/3.2.0/api/_as_gen/matplotlib.pyplot.scatter.html`

[9] Scikit-Learn Confusion Matrix: `http://scikit-learn.org/stable/auto_examples/model_selection/plot_confusion_matrix.html`

[10] Scikit-Learn K-Means: `https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html`