Motorbike Ambulance Calls

Technical Task Report

Authors:
Oleh Kozynets

November 2019

1 Data Analysis

To build any machine learning model we first need to study our data. The provided data set (see file `motorbike_ambulance_calls.csv`) contained 17379 individual entries and 14 columns. The 'date' column was dropped, since the 'yr', 'mnth', 'hr', 'holiday', 'weekday', 'workingday' columns contain almost the same information. The 'season' column contained four unique values: *winter*, *spring*, *summer*, and *autumn*. We mapped these values to the set of natural numbers.

To see the distribution of values for different variables we plotted histograms for each of them, see Fig. 1. In the majority of cases, feature values are evenly distributed across the input range, except for the 'cnt' and 'holiday' columns. In particular, there are significantly more feature vectors where the number of emergency calls ('cnt') is between 1 and 250, while the overall input range is between 1 and 977. The logarithmic transformation can be applied to this particular column in order to make the distribution more even and bell-shaped. The input range of several selected features was normalized by the data providers.

Our target variable is located in the 'cnt' column, which is the number of motorbike ambulance calls. To select the most relevant variables, we calculated correlation coefficients between the target variable and the others, see Fig. 2. In general, Pearson's correlation coefficient may be used as a test statistic to assess the significance of a variable for a regression task. It is easy to see that the highest correlation moduli have the 'temp', 'atemp', 'hr', 'hum', 'year', 'weathersit', 'mnth', and 'windspeed' columns. Only the 'weathersit' variable is categorical, while the others are numerical. A subset combining these variables should give us a good result, wile reducing the number of predictors in our data. Thus, all other predictors were removed from the data.

Let take a closer look at the selected subset. The 'yr' column have quite high positive correlation with the number of calls, which means that in the second year there were more calls. Unfortunately, the model trained using this particular variable cannot be incorporated to make future predictions of ambulance calls. Thus, we decided to remove it from the data set. The 'mnth' and 'windspeed' are numerical variables, their correlation coefficients are significantly lower then the ones for the 'temp', 'atemp', 'hr', and 'hum' numerical variables, so we removed those two from the data as well. In the end, we obtained a data set containing four numerical predictors ('temp', 'atemp', 'hr', and 'hum') and one categorical feature ('weathersit'). The numerical columns were additionally standardized, while the categorical one was further transformed using one-hot encoding.

Typically, along with irrelevant features, redundant information should be eliminated as well. The scatter plot of numerical variables is presented on Fig. 3. It is easy to see that the 'atemp' and 'temp' predictors are highly correlated. The correlation coefficient between the two is 0.988. Consequently, we removed the 'atemp' variable to avoid redundancy. The other variables are not correlated between each other and were left in the data.

2 Regression

To selected the best regression model we tested several predictors implemented in the `scikit-learn` package: linear model with L_1 regularization (Lasso), linear model with L_2 regularization (ridge regression), random forest regression (RFR), and support vector regression (SVR). First, we evaluated the performance of these models by mean absolute error (MAE). The MAE estimation was done via 5-fold cross validation for models' default hyper-parameters. The best accuracy was demonstrated by RFR with MAE = 87.8 and STD = 13.2, though it was very close to the score obtained by SVR with MAE = 92.6 and STD = 27.2. The worst performance was showed by the linear models with MAE = 116.9 and STD = 22.5. Here STD stands for standard deviation.

In the following step, we selected the best hyper-parameters for the most relevant regression model, i.e. random forest regression, using grid search. They are as follows: the number of trees n_estimator = 50, the maximum depth of a tree max_depth = 5, and the number of features to consider when looking for the best split was set to the number of features in our pre-processed data max_features = 'auto'. We used default values for the remaining hyper-parameters. We evaluated this model in the same manner as in the previous

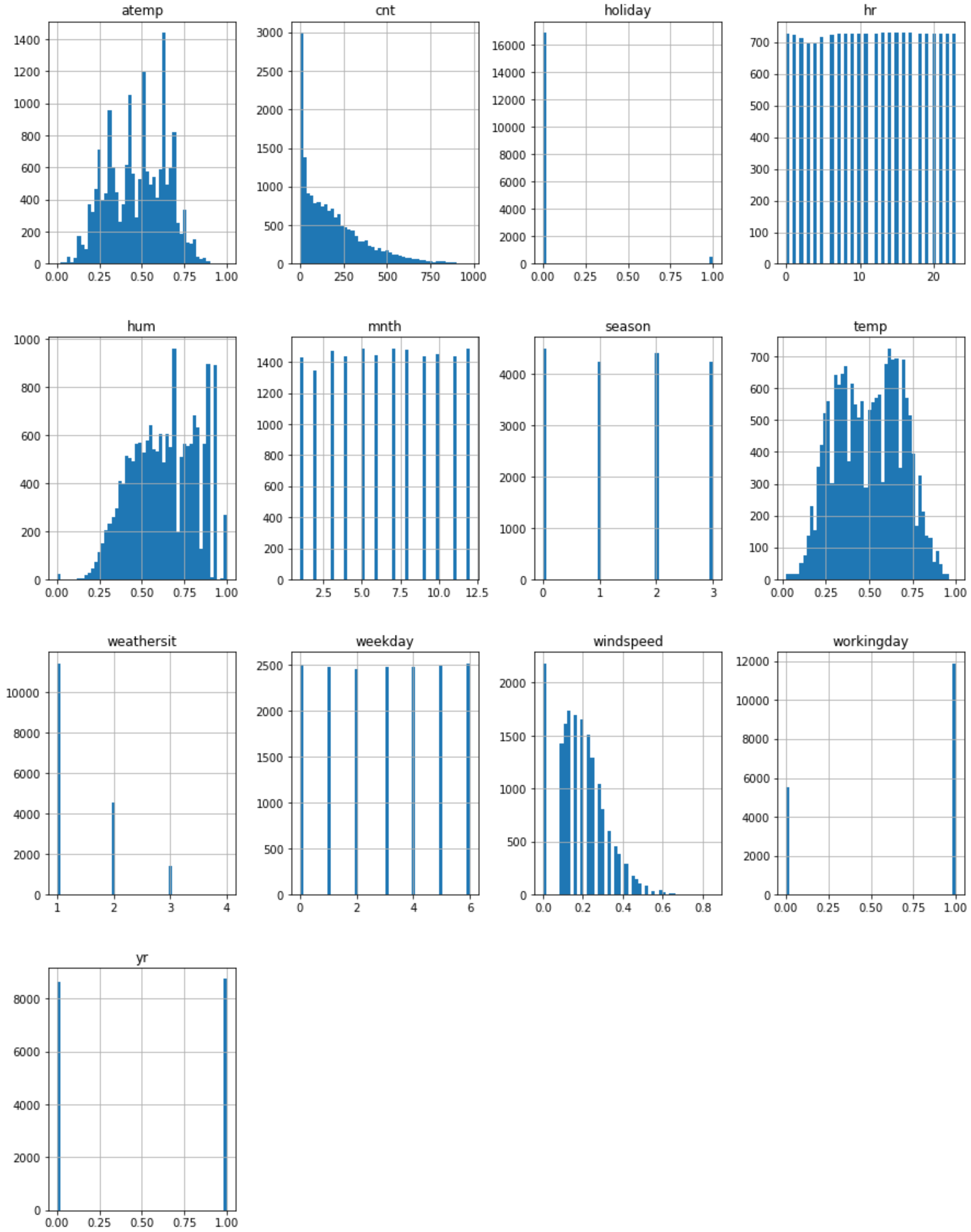

Figure 1: Histograms showing distribution of samples with respect to different predictors. The majority of features exhibit evenly distributed values across the input range, except for the 'cnt' and 'holiday' columns. In particular, there are significantly more feature vectors where the number of emergency calls ('cnt') is between 1 and 250, while the overall input range is between 1 and 977. The input range of numerical features was normalized by the data providers.

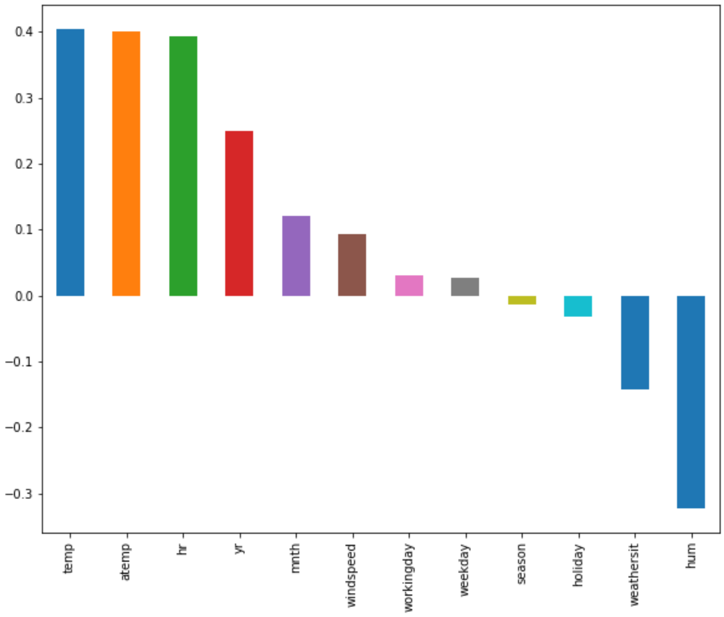

Figure 2: Correlation between the number of ambulance calls `cnt` and other predictors. The highest correlation moduli are shown by the 'temp', 'atemp', 'hr', 'hum', 'year', 'weathersit', 'mnth', and 'windspeed' predictors.

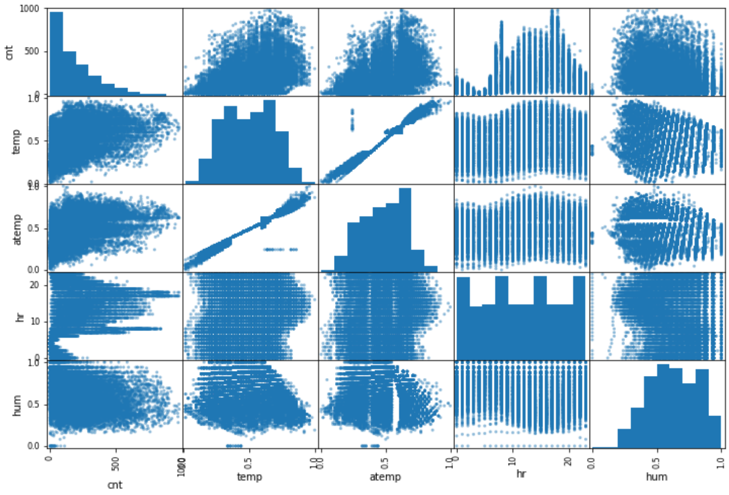

Figure 3: Scatter matrix of selected numerical variables. The variables were selected based on their correlation with the target predictor. The 'atemp' and 'temp' predictors are highly correlated with each other, their correlation coefficient is 0.988. The other variables do not display significant mutual correlation.

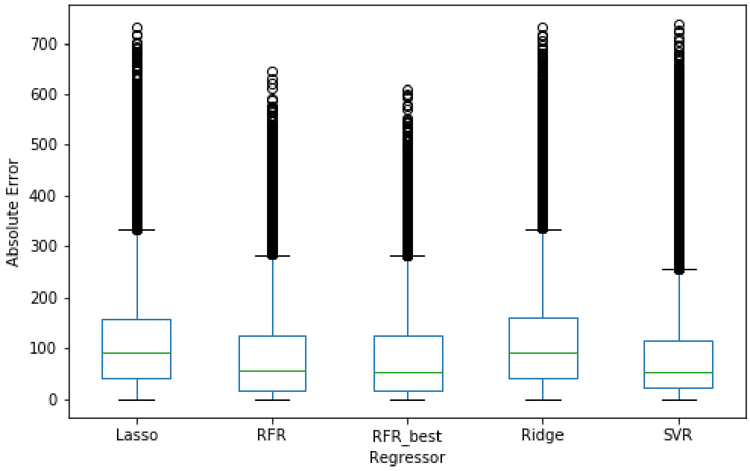

Figure 4: Distribution of absolute error values for the selected predictors: linear model with L_1 regularization (Lasso), linear model with L_2 regularization (Ridge), random forest regression (RFR), and support vector regression (SVR), trained using default parameters. The RFR_best model parameters were fine-tuned via grid search.

step and obtained MAE = 85.2, STD = 12.4. Unfortunately, it is not a significant improvement over the model trained using default parameters. One can compare the performance of the used models via a boxplot of the error moduli distribution Fig. 4. In addition to the lower mean absolute error, the fine-tuned random forest regression model has less outliers than the other models.

3 Event and Anomaly Detection

After the data exploration step we obtained a data set with 17379 rows and 7 columns ('temp', 'hr', 'hum', and 'weathersit'). In order to be able to visualize the distribution of out data entries we need to reduce dimensionality to 3 or less. For that purpose we selected the principal component analysis algorithm (PCA), while keeping either 2 or 3 most significant components. Prior to running PCA on our data, we performed several pre-processing steps. In detail, we transformed information in the 'season' column from text to natural numbers. We also dropped the 'cnt' and 'date' columns, then we projected obtained data into low-dimensional feature spaces.

We used transformed data to test three outlier detection algorithms, namely, robust covariance, isolation forest, and local outlier factor. The outlier fraction was set to 15 % for every algorithm. The results for 2 principal components can be seen on Fig. 5. Though, the relative explained variance was around 90.6%, the transformed data is distributed almost uniformly across the feature space. Consequently, it is hard to visually spot any anomaly based on the data visualization. The robust covariance and isolation forest algorithms showed similar results by trying to eliminate the data on the boundaries of the reduced feature space. The local outlier factor algorithm produced completely different output. As it was mentioned above, there is no obvious inherent structure to the data at hand or any ground truth result to test against, so it is impossible to compare the produced results. Although, the rejection of the outer data entries made by first two algorithms is quite reasonable. Experiments for 3 dimensional feature space (PCA with 3 principal components) did not display any improvement over 2D feature space.

In addition, we tested whether the exploitation of the outlier detection algorithms for data pre-processing could improve regression accuracy. We trained the random forest regression with the optimal parameters

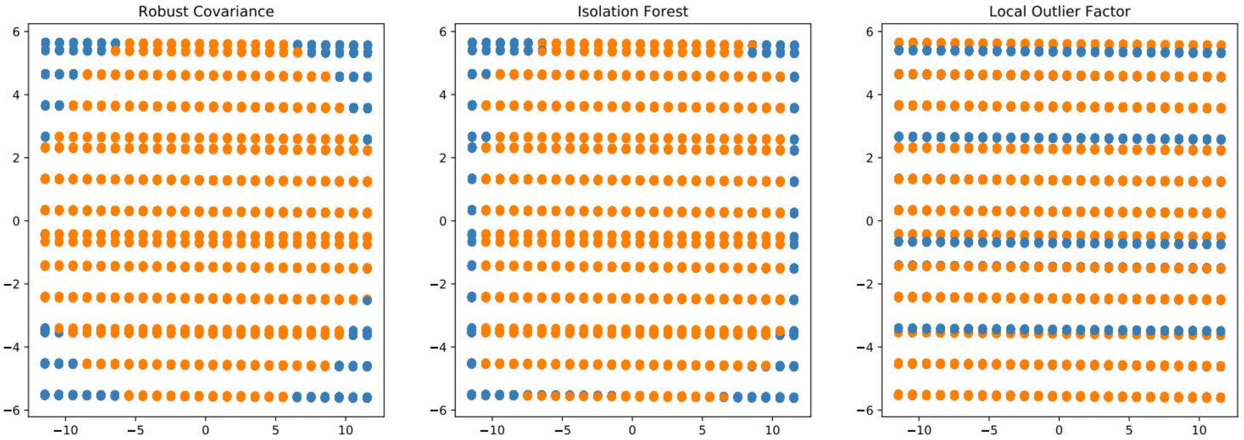

Figure 5: Outlier detection on the 2D data set. Dimentionality reduction was implemented using PCA. The 'cnt', 'date' columns were excluded. The orange points correspond to inliers, while the blue ones are outliers.

found in Section 2 on the inliers. The obtained results are: robust covariance MAE = 92.8, STD = 11.4; isolation forest MAE = 92.5, STD = 11.0; and local outlier factor MAE = 87.6, STD = 14.4. Therefore, the new pre-processing pipeline did not show any improvement over the one without outlier rejection. It is worth noting that the pipeline incorporating local outlier factor algorithm showed the best result among others.

4 Repository Structure

The source code was implemented using Python 3, Jupyter Notebook, NumPy, matplotlib, pandas, and scikit-learn. In addition to the code, we provided the `environment.yml` file, which one can use to install all the packages required.

We put our source code into three notebooks. In detail, the data exploration scripts can be found in the `motobike_analysis.ipynb` file. The `motobike_regression.ipynb` file contains scripts for building regression models. Finally, the results of testing the outlier detection algorithms can be seen in the `motobike_outliers.ipynb` file. In addition to those ones, we created a separate python module, `motobike_transformers.py`, which stores helper classes utilized for feature pre-processing and was shared across notebooks.

5 Conclusions

The provided data set contained records of motorbike ambulance calls. To built a robust regression model we added several pre-processing steps: we removed irrelevant and redundant information from the data, added standardization for the numerical columns and one-hot encoding for the categorical one. The best accuracy was showed by the random forest regression model with MAE = 85.2 and STD = 12.4. Regarding the input data distribution (Fig. 1) and relatively low correlation between separate features and the target column (Fig. 2), we can say that our regression models does quite a good job. Just to compare the scale, the best obtained MAE = 85.2 is well below the average value of calls $< N_{\text{calls}} >= 189.5$, while the input range was from 1 to 977 calls. We used PCA and outlier detection algorithms to spot outliers in the provided data. Incorporation of outlier rejection algorithms did not give us any significant improvement over the initial pipeline. The source code was attached to this report.