Youssef El Bouhassani
Pieter Moesman

Blog Post Feature Importance

How to determine the factors that influence charging behaviour using feature importance techniques?

As the number of EVs increase, the demand for solutions to deal with increased peak demands increases too. We have two options to deal with the increased peak demands.

One is a hardware solution where the grid is adjusted to meet the power demands. This approach requires intrusive measures to replace or extend existing electricity grids which could lead to massive investments in infrastructure renewal.

The other approach is more software and algorithm focused, where algorithms are used to schedule electricity demand for charging EVs in such a way that the increased demand is placed outside the peak moments as much as possible. This approach requires accurate ways to predict the charging and connection durations. Knowing how long a charging sessions will last is an important ingredient for applying smart charging. By predicting the charging needs at a given moment, we will be able to better match supply and demand.

In the SIMULAAD project we focused on developing methods for the second approach. Which is to predict charging time and connection duration to enable applying smart charging strategies. It turns out that the starting time of a session and the duration of previous sessions are by far the most important features to predict connection duration.

Narrowing down

To enable smart charging, two parameters should be predicted as accurately as possible. First, charging duration is the time that an electric vehicle is actually charging. This may range from minutes to more than 10 hours. Second, connection is the actual time an electric vehicle is connected to a charging station. Given that in most cases connection time is (much) longer than charging times, there is flexibility to reschedule the actual charging process, for instance by postponing it.

Predicting the charging- and connection duration is done using historical charging session data. The dataset used for the SIMULAAD project spans multiple years, has more than seventy variables and contains charging sessions data from the so-called G4 Cities (Amsterdam, Rotterdam, the Hague, Utrecht) and the metropolitan regions of Amsterdam (MRA-e) and The Hague and Rotterdam (MRDH) in the Netherlands.

With more than nine million records the dataset is very extensive and requires some treatment prior to model building. The obvious place to start is to narrow down the data to include a specific location and time period. This allows us to study a specific use case. In this blog post we will focus on Amsterdam for the year 2017.

The second step is to choose a limited set of features in the model building stage. Ideally we would like to select those features that add to the quality of prediction.

In this first stage of the SIMULAAD project we focused on developing methods to choose the right features prior to model building.

Feature selection

Feature selection is a widely used technique in the data preparation stage of developing machine learning models. Feature selection is used to find those features that have a significant contribution to predicting the target variable.

In this case the target variable is the connection duration. So out of the seventy plus features in the dataset, we would like to select those that have a significant contribution to predicting the connection duration. Although this blog post is limited to discussing the prediction of connection duration, the same methodology can be applied to other target variables, such as charging duration.

The Boruta library

There are different methods that can be used for feature selection. Comparing the relative importance of the different features is the method that will be discussed in this blog post.

The application of feature importance is done in R using the Boruta library.

In short, the Boruta library makes use of iterative methods to determine the *relative* importance of features based on tree methods. The Boruta package compares the performance of features in the dataset with random features (so called shadow features) for predicting the target variable. The library attempts to find those features that score way better than the shadow features when predicting the target variables.

Implementation

The Boruta library is available from the cran repository and can be installed and activated using the following code:

```
Install.packages(“Boruta”)
Library(Boruta)
```

```
Boruta.data <- Boruta(data = data, TargetVariabel ~ ., maxRuns = *number*, doTrace = 2)
Boruta.data
```

This function takes the dataset including the target variable. The Boruta function is iterative so it does a number of runs predicting the target variable until the required accuracy is achieved. The number of iterations can be limited by assigning a number to the variable maxRuns. One can choose to show the computation progress by setting the variable doTrace equal to 2.

Results

The application of a smart charging algorithm requires the prediction of connection duration. This is used as the target variable for the Boruta function. Although the dataset contains many variables, this analysis is limited to the features as shown in figure 1. In this figure we see the computed mean feature importance on the horizontal axis. Since the Boruta function is iterative, we get the spread in importance based on the feature importance at each iteration. During each iteration the actual features are compared with the shadow features to determine their relative importance based on the z-score. In figure 1 the green features are confirmed to be important features. Blue features are rejected, so these are considered to be irrelevant in predicting connection duration. Red features are features that have importance close to their shadow features.

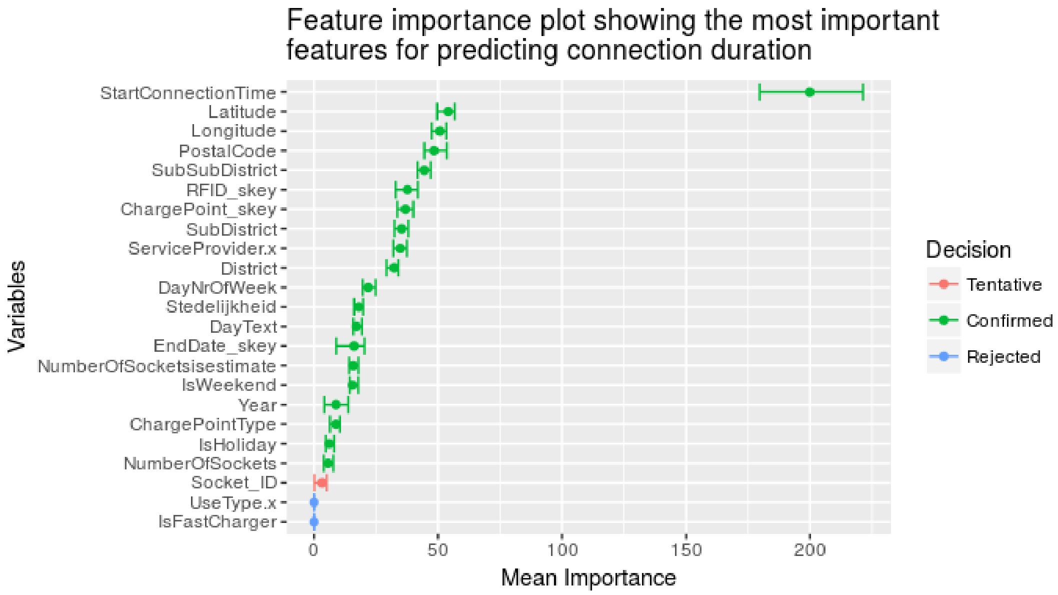

Figure 1: Plot of the importance of each feature for the prediction of connection duration.

What is interesting to see in this case is that the starting time of a session is by far the most important feature when it comes to predicting connection duration. The second most important features are location-related (longitude and latitude). The third most important feature is user-related.

So when it comes to predicting connection duration, what matters, in order of importance, is when the charging session starts, where it takes place and whose charging session it is.

Testing the importance of generated features

As shown in figure 1, the Boruta function helps identify which features are important. In figure 1 only the existing features are considered.

Training machine learning models often requires generating new features based on the existing ones. To test the importance of these newly generated feature, the same method can be used.

For example, one of the hypotheses is that the connection duration of the previous session(s) has influence on the connection duration of the current session. The rationale behind this hypothesis is as follows: to achieve the desired state of charge, the next session can be short if the sessions in the past where long and vice versa.

To test this hypothesis new features are generated based on the connection duration from previous sessions. These new features are then added to the dataset which is then fed to the Boruta function. The result is shown in figure 2.

As can be seen in figure 2, the connection duration previous three sessions are approximately equally important in predicting the connection duration of the current session. Compared to figure 1, it appears that the connection duration of the previous sessions is more important than location and user related features .

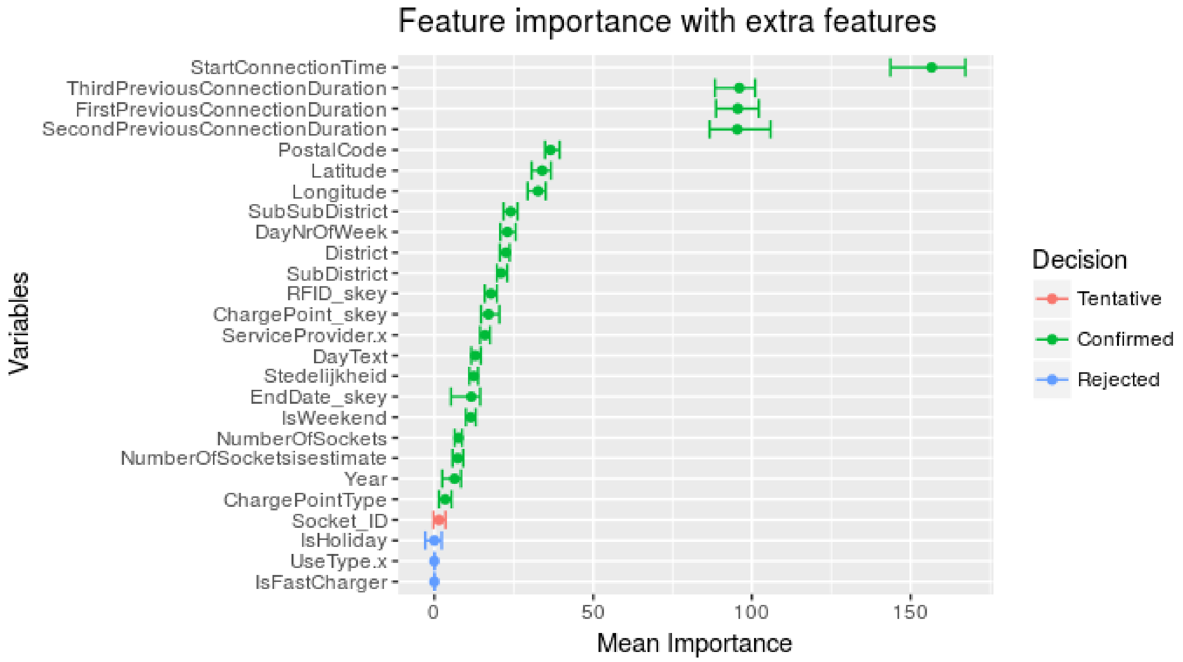

Figure 2: Plot of the importance of each feature for the prediction of connection duration with previous charging sessions.

Training machine learning models can be time consuming, especially when many features are involved. In this blog post, we discussed a possible method to reduce computation time by considering feature importance first, prior to model training. The feature importance method is also useful in determining which generated features are important. Although feature creation is often based on intuition and expert knowledge, having a solid objective method to compare the features helps in choosing the right features.