SVR-based Modelling for the MoReBikeS Challenge: Analysis, Visualisation and Prediction

Yu Chen, Peter Flach

Abstract. We present a solution to the MoReBikeS challenge in ECML PKDD 2015 conference by analysing data from different aspects, by visualising latent patterns, by building a set of features. The proposed model is accurate, efficient yet easy to implement.

1 Introduction

During the development, we consider it necessary to know an answer about the following issues:

- Which features are informative and which are noisy?
- How do they affect the target value?
- How to make the best use of them?

There are two parts of this solution for MoReBikeS challenge, one is reconstruction of the feature space and another is the regression.

As the task is to predict the number of bikes of a station three hours in advance, so a regression algorithm is needed to learn the mapping between target value and features, in this part an SVR model is deployed.

The critical part is feature reconstruction, better features can improve the performance of a simple regression model significantly, so based on the analysis and visualisation of given features, a new feature space has been generated for regression.

2 Feature Selection and Representation

2.1 Raw Features

There are 24 given features in total which can be divided to 4 categories.

1. **Facts of stations**. The facts of stations provided in the data set include the station ID, the latitude, the longitude and the number of docks in that station. All these properties for one station do not change over time.
2. **Temporal information**. The timestamp of a data entry consists of eight fields: "Timestamp" in terms of seconds from the UNIX epoch, "Year", "Month", "Day", "Hour", "Weekday", "Weekhour", and "IsHoliday" which

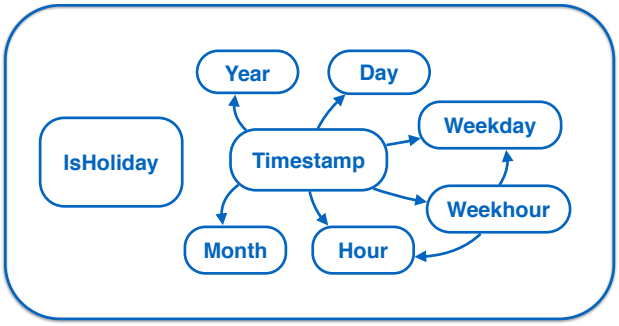


Fig. 1: Relations between Temporal Features

indicates whether the day is a public holiday. These features are giving overlapping temporal information, we only need a subset of them to represent a time point. As shown in Figure 1, the "Timestamp" is actually including information of "Year", "Month", "Day", "Hour","Weekday" and "Weekhour", whereas "Weekday" and "Hour" also can be deduced by "Weekhour". Only "IsHoliday" is independent to any of others.

3. **Weather**. This set of features include "windMaxSpeed", "windMeanSpeed", "windDirection", "temperature", "relHumidity", "airPressure", "precipitation". One major observation of weathers is that all the values of all the seven fields share among all stations.
4. **Counts and their statistics**. This set of features relates to the target value directly. First of all, "bikes_3h_ago" gives the target value of the 3-hour-earlier time point at a station. The full profile features use all previous data points of the same "Weekhour" to obtain long term statistics for each "Weekhour" in each station, accordingly the short profile features only use at most four previous data points to obtain short-term statistics. The long-term statistics of the 200 old stations only have very small changes over time in contrast to the short-term ones.

2.2 Selection of Features

1. **Features to remove:**
   - "Year" and "Month": "Year" and "Month" are fixed to October 2014 in the training data set, there is nothing to learn from them.
   - "Timestamp": accordingly "Timestamp" is too general to distinguish different temporal information and has too many possible values to clearly indicate similarity between different time points.
   - Full profile features: the actual meanings of full profile features are different from 200 old stations to 75 new features, because the historical data used to calculate the long term statistics are spanning over 2 years for the 200 old stations but only several weeks for the new 75 stations.
2. **Features to keep:**

- "IsHoliday": this feature doesn't overlap with any other temporal features(Figure 1) and gives extra information in addition to the timestamp.
- "Day": since "Timestamp" is already removed, "Day" has become a temporal feature without any alternative and probably includes some periodical information.
- Bikes of 3 hours ago and short profile features: unlike long term statistics, these features of 75 new stations are aligned with the 200 old stations, so they can be very informative.
- Temperature: the existing linear models only keep temperature among seven weather features, which implies it is a useful information.

3. **Features to keep or remove:**
   - Facts of stations: these features provide characteristics of a station, as they do not change over time, so they can only provide static knowledge about a station, but they might be useful to recognise the pattern of a certain station.
   - "Weekday+Hour" or "Weekhour": although "Weekhour" is a general representation of "Weekday+Hour", it emphasises the difference between every hour in one week, while the "Weekday+Hour" can represent two types of differences: "Weekday" differs or "Hour" differs. It needs further analysis to decide which representation is more appropriate for this data set.
   - Other weather features: it is not clear whether these weather features are signals or noises before further analysis.

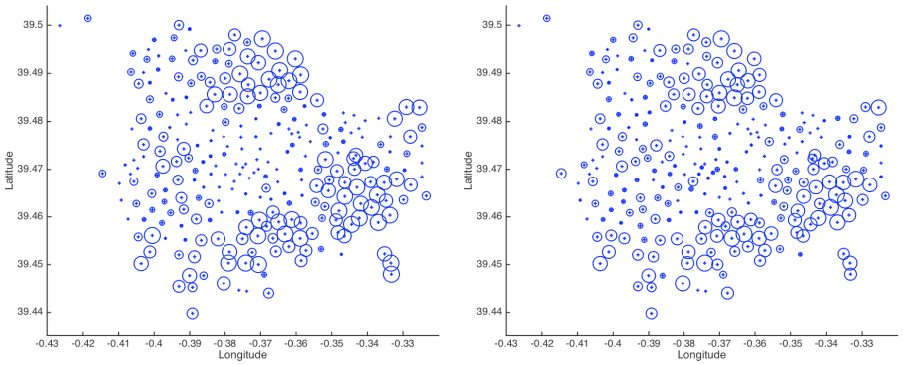
2.3 Visualisation

The provided statistical features use "Weekhour" as the criteria to choose time points, it implies "Weekhour" may have a strong effect on the target value. Therefore, we visualise the target of each station at different week hours to capture latent knowledge of the data.

Below figures give overall prospects of all stations in various hours and weekdays. The "+" marks the location of a station, the circle around it indicates how many bikes in that station in 3 hours later, the radius of a circle has been normalised by number of docks of that station.

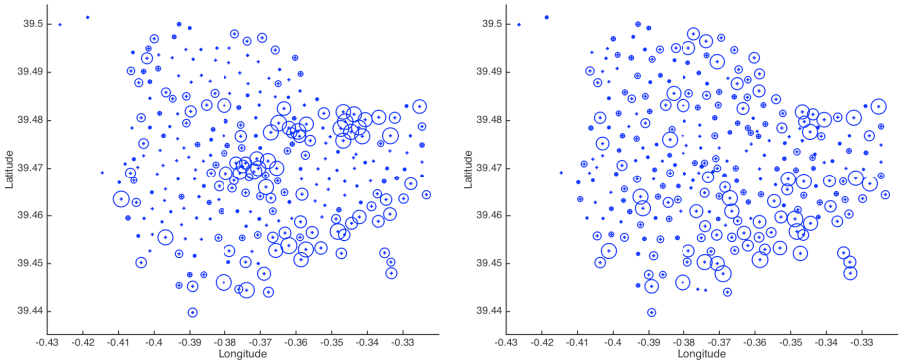
Figure 2 shows four chosen hours in a Wednesday. Obviously the bike storage in that city is stable during night and most bikes are stored in stations near the north and south edges(Figure 2 (a) and (b)), these areas are probably the popular residential areas of the city. In contrast, most bikes have been transferred to the centre of the city in a typical working hour(Figure 2 (c)), and in an off-work hour bikes are spread over the city(Figure 2 (d)).

Concerning the doubts between "Weekday+Hour" and "Weekhour", we need some comparison between different temporal categories. Figure 3 gives overall prospects of same hours in the daytime of a Saturday and Monday. At 10:00 on Saturday(Figure 3 (a)), there are still a number of bikes stored in residential areas and stations in the city centre are nearly empty, and at 16:00(Figure 3 (b)) a new hot spot emerges, quite a lot of bikes have been moved to the east edge of



(a) Wednesday 0:00

(b) Wednesday 7:00



(c) Wednesday 10:00

(d) Wednesday 16:00

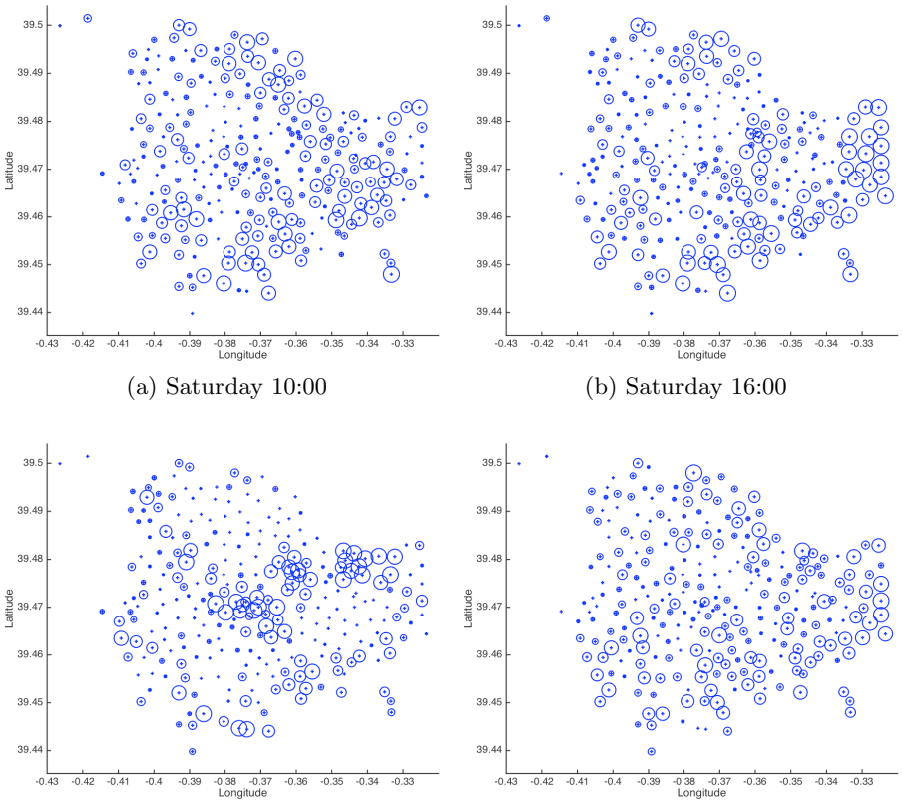
Fig. 2: Different hours on Wednesday

the city. These two hours have completely different properties comparing to the same hours on Wednesday. On the contrary, the two hours on Monday(Figure 3 (c),(d)) are very similar to those on Wednesday(Figure 2 (c),(d)). They clearly have a mutual pattern and it is easy to understand: Monday and Wednesday are both working days.

Conclusions from visualisation results:

- Two time-points with different hours in the same weekday are not necessarily more similar than different hours in different weekdays.
- Two time-points with the same hour in different weekdays are not necessarily more similar than different hours in different weekdays.

So it is clear now that "Weekday" + "Hour" may indicate false similarity between time points whereas "Weekhour" can avoid such errors without losing useful information.



(a) Saturday 10:00

(b) Saturday 16:00

(c) Monday 10:00

(d) Monday 16:00

Fig. 3: Same hours in different weekdays

The visualisation results also indicate that there are different behaviour patterns underlying different groups of stations, such as the behaviour of stations near north and south edges are similar, but very different with the stations in the city centre. So it could be helpful to recognise patterns in groups, the method to do so will be introduced later in section of fast test. Accordingly the facts of stations (station ID, longitude, latitude, the number of docks) are not the criteria to identify a group, hence we can remove them from the feature space as well.

2.4 Feature Representation

How to represent these selected raw features is another important aspect of reshaping the feature space. There are two simple but powerful methods applied in this task.

1. **Vectorisation**

In this context the two selected temporal features("Day", "Weekhour") are categorical variables, according to the visualisation results the distance calculated by their numerical values certainly can not represent the similarity between two time points. Vectorisation is a common choice for such problem, each value of a categorical feature is transformed to a bit in a binary vector. After this transformation, "Day" and "Weekhour" have generated 199 new features.

2. **Normalisation**
   The profile features are counting numbers of bikes and they have different upper limits due to various capacities of stations. These features can be normalised by number of docks of each station so that they are comparable between all stations:

$$\hat{f}_k(t) = \frac{f_k(t)}{N_k(t)} \tag{1}$$

   $f_k(t)$ represents one of the following profile features of station k at time t:
   - "bikes_3h_ago"
   - "short_profile_bikes"
   - "short_profile_3h_diff_bikes"

   $N_k(t)$ is the number of docks of station k at time t.

So far we have selected and reconstructed temporal features by visualisation and vectorisation, also transformed profile features by normalisation. The remaining uncertainty about weather features will be pinned down by feature filtering during tests.

3 Regression Model

To perform the regression task, there are abundant options to choose a regression model, like Linear Regression, Gaussian Process Regression, Nearest Neighbour Regression, Support Vector Regression and Neural Networks, etc.. Considering the prediction performance and computational complexity, Support Vector Regression(SVR) is a safe and handy choice here. It's probably not the best regression model for this task, however, it is capable of performing well.

The deployed SVR model is implemented by scikit learn [3] and it is an Epsilon-Support Vector Regression model [1]. The chosen kernel function is sigmoid kernel function [2]:

$$K_{ij} = \tanh(\gamma x_i^T x_j + c_0) \tag{2}$$

Where $x_i^T x_j$ represents the inner product of 2 data points.The parameters of this SVR model are chosen by the fast tests which will be introduced in later section:

$$C = 2, \epsilon = 0.02, \gamma = 0.25, c_0 = -1 \tag{3}$$

To align with normalised profile features and to fit the sigmoid kernel better, the target value is transformed by below equation for training the SVR model:

$$\hat{\nu}_k(t) = \frac{\nu_k(t)}{N_k(t)} - \frac{\nu_k(t-3h)}{N_k(t-3h)} \tag{4}$$

Where $\nu_k(t)$ is the number of bikes at time t in station k; $N_k(t)$ is the number of docks at time t in station k. This equation can be simplified when the number of docks of a station does not change over time, i.e. there exist a positive integer N_k such that $N_k = N_k(t)$ for all $t \in \mathbb{R}^+$, which is the case in our data.

$$\hat{\nu}_k(t) = \frac{\nu_k(t) - \nu_k(t-3h)}{N_k} \tag{5}$$

4 Testing and Evaluation

4.1 Fast Test

As the computational cost of training the SVR model by all data points is quite expensive, only using partial data to build the SVR model can reduce the cost efficiently. The idea of the fast test is to use data of **K** neighbours of a certain station **n** to train an SVR model and then test it on data of the station **n**.

According to the visualisation result, there are different behaviour patterns underlying different groups of stations, so we can obtain a group for a certain station by identifying its neighbours with similar behaviour. The neighbours are obtained by ranking euclidean distances between stations, the target value of each time point of a station is treated as a feature, as there are 745 time points of each station, then the euclidean distance is calculated in a feature space with 745 dimensions, and if there are missing values of some time points, the mean value of all time points of that station will be used instead.

More neighbours means more data points and a robuster regression model, meanwhile more data points also increase computational complexity for both training and testing vastly. Setting the number of neighbours **K** to 10 can give a reasonable performance with considerable reduction of computing expense. Here the reasonable performance means it is better than the average performance obtained from the existing linear regression models.

4.2 Validation

There are two datasets used to validate optimisations of the model, one is the data of 75 new stations in October 2014, another is the data of 10 old stations in November, December and January from 2012 to 2014. The reason to do so is that the test data is from 75 new stations in November, December 2014 and January 2015, such validation sets could avoid overfitting on a certain month or certain stations. Only those changes which can improve the performance on both validation sets will be adopted.

The hyper-parameters of the SVR model are selected by this validation strategy. The weather features are filtered out one by one through such validation as well, each time we remove a weather feature to see whether the performance become better or worse, if it has become worse, we keep this feature, otherwise, we just remove it.

4.3 Online test

As we have submission opportunities for the online test by a small set of test data, these test results provided important feedback about the model, such as the SVR model trained by all data points is more robust than those trained by neighbours, the importance of "Weekhour" is consistent with our analysis, the full profile features are confusing the model, etc.. The comparison between some attempts is shown in Table 1.

	Training Set		**Feature Options**		
MAE	20 Neighbours	275 Stations	"Weekhour"	"Weekday"+"Hour"	Full Profile
2.625	✓			✓	✓
2.612	✓		✓		✓
2.52	✓		✓		
2.496		✓	✓		✓
2.46		✓		✓	
2.37		✓	✓		

Table 1: Comparison between Online Test Results

4.4 Final Model and Full Test

The final model for this challenge is decided eventually with following features:

1. "IsHoliday", "Day", "Weekhour": vectorised.
2. "bikes_3h_ago","short_profile_bikes","short_profile_3h_diff_bikes": normalised by equation (1).
3. "temperature".

The SVR model is as described in section 3 and the training set is data of 275 stations in October 2014.

The MAE(Mean Absolute Error) score of the final model on full test data is 2.051. Since all online tests included vectorisation, so we have tested the same combination of features without vectorisation on the full test data as well, the MAE score is 3.519. It appears the vectorisation plays a key role in this model to boost the performance.

5 Conclusion

The deployed methods are quite simple and easy to apply. The main idea is to extract important information from given features and then build new features upon them to optimise the feature space for regression. The performance highly relates to how well the feature space represents patterns underlying the data. The disadvantage of the final model is that new features are not obtained automatically by learning algorithms, and the computational cost is still high which might be improved by a better sampling method to shrink the training set or a faster regression algorithm instead of SVR.

References

1. Chang, C.C., Lin, C.J.: Libsvm: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology (TIST) 2(3), 27 (2011)
2. Lin, H.T., Lin, C.J.: A study on sigmoid kernels for svm and the training of non-psd kernels by smo-type methods. submitted to Neural Computation pp. 1–32 (2003)
3. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research 12, 2825–2830 (2011)