

Chapter 5: Results & Discussion

Draft version

This chapter presents and discusses the results of the experiment described in Chapter 4. It is structured as follows. The first section shows the clusters that resulted from the cluster loop, along with their main characteristics, and the chosen locations of the model points. Section two presents the structure of the models that were build in the model loop, and the residual diagnostics for each them. Then, the third section focuses on the accuracies of the forecasts produced by DBAFS, compared to the baseline. Finally, in the fourth section, the limitations of DBAFS are discussed, and recommendations for possible improvements are given.

5.1 Clustering

Figure 5.1a shows the grid overlaying the JUMP Bikes system area in San Francisco, including the centroids of all the grid cells. In total, the grid contains 249 grid cells, each 500 meter high and 500 meter wide. Figure 5.1b shows the calculated number of pick-ups per grid cell, during the training period. On average, there were 218 pick-ups per grid cell, which corresponds to approximately eight pick-ups per day. The maximum number of pick-ups in a grid cell was 1985 (i.e. 71 per day on average), while in 25 of the 249 grid cells, there were no pick-ups at all. It can be seen that high counts of pick-ups occurred in the grid cells along the diagonal axis from south-west to north-east. Mainly in the south-eastern corner of the system area, the usage intensity was very low.

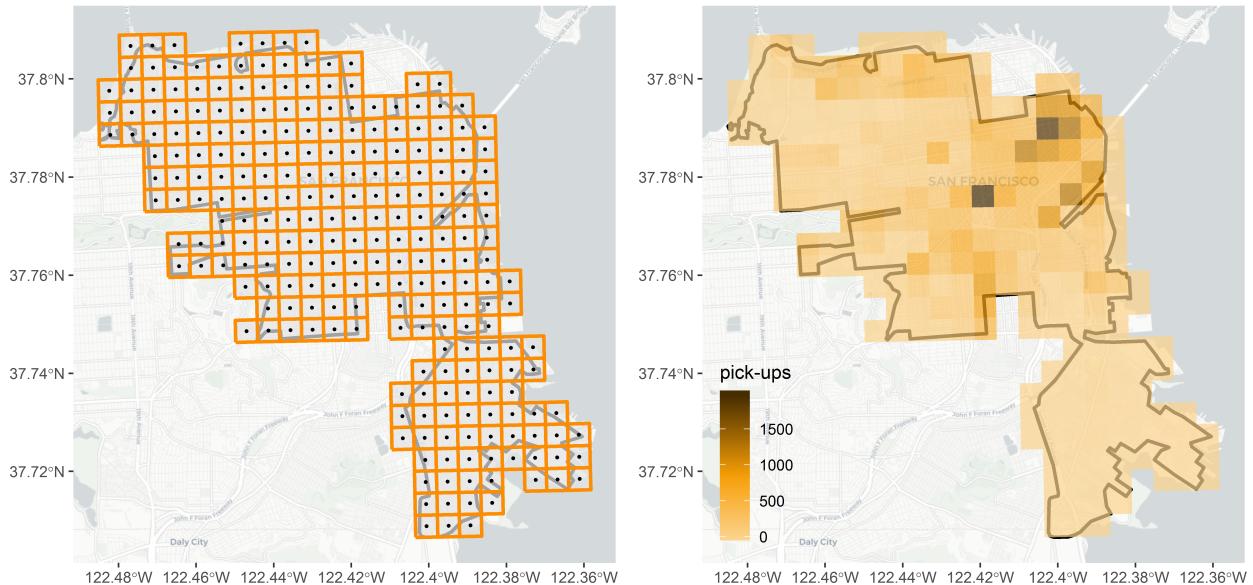


Figure 1: a) grid overlaying the system area; b) number of pick-ups per grid cell

Recall that for each grid cell centroid, a time series of historical distance data was queried, and that the normalized, average weekly patterns in these data were clustered using spatially constrained hierarchical clustering. The automatic procedure of defining the number of clusters k and the mixing parameter α , lead

to a definition of $k = 4$ and $\alpha = 0.6$. This resulted in a partition containing four fully spatial contiguous clusters. The geographical outlines of these clusters are shown in Figure 5.2a. The centroid of all grid cell centroids in each cluster, weighted by the number of pick-ups in the corresponding grid cells, are shown in Figure 5.2b. These weighted centroids serve as the model points in DBAFS.

Roughly speaking, and based on a large study of neighbourhood indicators in San Francisco (San Francisco Department of Public Health 2014), the four clusters can be characterized as follows. The orange cluster covers the Bayview/Hunters Point neighbourhood, which is a rather isolated area, with a high percentage of low-income households and relatively high crime rates. The blue cluster forms the city center of San Francisco, containing the neighbourhoods with the highest population densities, but also with a relatively high job density compared to the residential density, and large areas zoned for commercial usage. The purple cluster mainly contains neighbourhoods where the residential density is high compared to the job density, and the area zoned for commercial usage is relatively small. Finally, the green cluster covers the Presidio Park, a recreational area with few inhabitants, and a relatively high number of bike lanes. For the sake of clarity, the orange, blue, purple and green clusters are from now on referred to as the *Bayview*, *Downtown*, *Residential* and *Presidio* clusters, respectively. Consistently, the four corresponding model points will be called the *Bayview*, *Downtown*, *Residential* and *Presidio* model points, respectively.

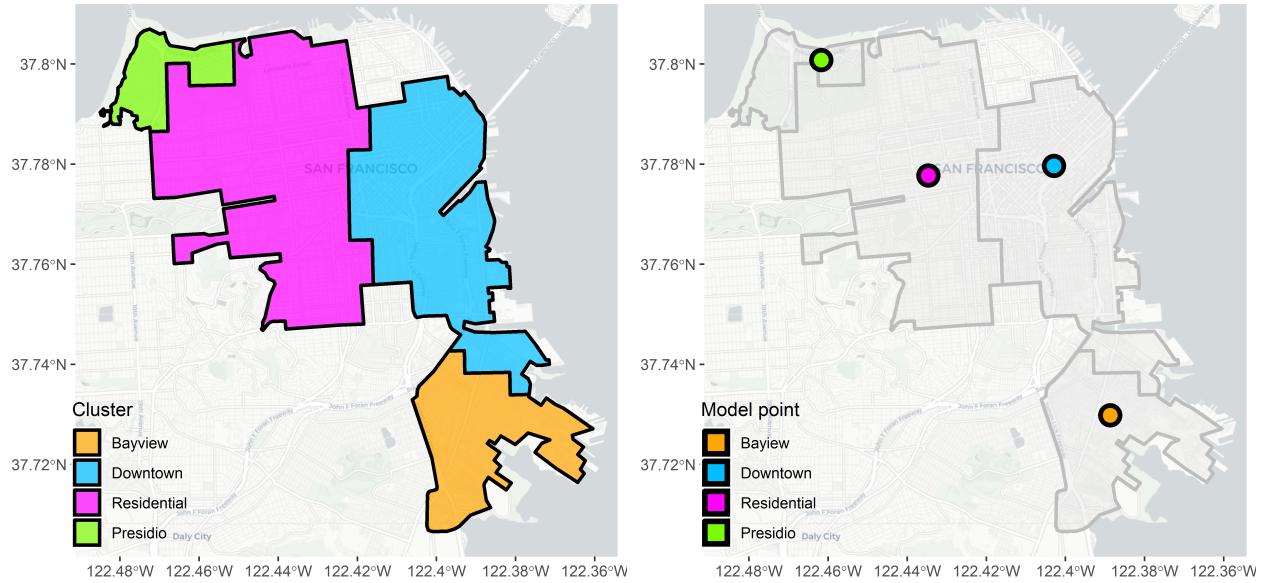


Figure 2: a) geographical outline of the clusters; b) geographical locations of the model points

Table 1 presents some descriptive statistics of the time series, averaged per cluster, and averaged over the whole system area. From the 249 grid cells, more than a hundred are located within the Residential cluster, while the Presidio cluster is by far the smallest of the four.

During the training period, the nearest available bike was on average located 619 meters from the grid cell centroids. In the Bayview cluster, however, this was more than one kilometer, a difference of almost a factor two compared to the Downtown cluster, and even more compared to the Residential and Presidio clusters. The Bayview cluster also showed the largest variation in the data, with a high average standard deviation compared to the other clusters, and an average range that spanned more than four kilometers. This can possibly be explained by the low usage intensity of the bike sharing system in this part of the system area. When the number of bikes in an area is low, the nearest available bike and the second nearest available bike are more likely to be far away from each other. In that case, when the closest of them gets picked-up, the distance to the nearest available bike will suddenly increase substantially. The other way around, when

all available bikes are far away, and one bike gets dropped-off inside the area, the distance to the nearest available bike will suddenly decrease substantially.

Although not as extreme as the Bayview cluster, also the other clusters had on average high ranges when compared to the mean and standard deviation. However, the standard deviation itself turned out to be rather small relative to the mean. This implies either the presence of outliers, or population distributions with thin, but wide tails.

The first order autocorrelation measures the average dependency between data values at time t and corresponding data values at time $t - 1$. In the whole system area, this dependency was strong, especially in the Bayview and Presidio clusters. These high autocorrelation values are important, since they imply that it is reasonable to use past observations when forecasting future ones. However, the calculated spectral entropy values show that in general, the data are also very complex, and the forecastability is low. This mainly concerns the Downtown and Residential clusters, which contain, as could be seen in Figure 5.1b, the areas where the pick-up density is high. In such areas, the data are more dynamic, since bikes get picked-up and dropped off constantly, and the location of the nearest available bike will change often. In most cases, the more dynamic the data, the harder to forecast.

Table 1: Descriptive statistics of the grid cell centroids distance data

	N	μ	<i>range</i>	σ	$\rho(1)$	H
Total	249	619	2726	155	0.82	0.77
Bayview	46	1080	4021	155	0.95	0.67
Downtown	81	557	2551	77	0.77	0.81
Residential	103	490	2410	112	0.79	0.81
Presido	19	462	2057	137	0.92	0.68

Note:

Except N , all metrics are calculated for each time series separately, and averaged afterwards.

¹ N is the total number of grid cell centroids

² μ is the mean of the data, in meters

³ *range* is the difference between the maximum and minimum data value, in meters

⁴ σ is the standard deviation of the data, in meters

⁵ $\rho(1)$ is the first order autocorrelation, see section 2.2.1

⁶ H is the normalized spectral entropy, see section 2.2.3

Figure 5.3 shows the normalized, average weekly patterns of the time series, averaged once again per cluster. The patterns can be explained intuitively. The Bayview cluster has a low usage intensity, and although there are peaks in the data every day, a clear and consistent pattern is absent.

The Downtown cluster has a high density of jobs and commercial activities. During working hours, the demand for bikes is low, which leads to a high number of available bikes, and consequently, short distances to the nearest available bike. In the afternoon, just after working hours, the demand starts increasing, and it gets harder to find an available bike nearby. Later in the evening, the demand decreases again. In the weekends, the daily peaks are less clear, and spread out over a longer timespan.

The Residential cluster shows the exact opposite pattern. In the morning rush hours, commuters use the bike to get to work, and not many available bikes are left in the residential areas. Hence, in those areas, the distance to the nearest available bike is higher during working hours. In the afternoon, commuters come back from work, and leave the bikes in the residential areas, causing a decrease in distance to the nearest available bike. Just as in the Downtown cluster, the peaks and valleys in the weekends are less strong. Furthermore, they happen later on the day, corresponding to the same periods as the Downtown cluster.

Finally, the Presidio cluster is mainly a recreational area. There are a lot of bikes, but during weekdays, they are used less, leading to small and constant distances to the nearest available bike. In weekends, and mainly on Sunday afternoon, the usage intensity is high, and it takes longer to find an available bike.

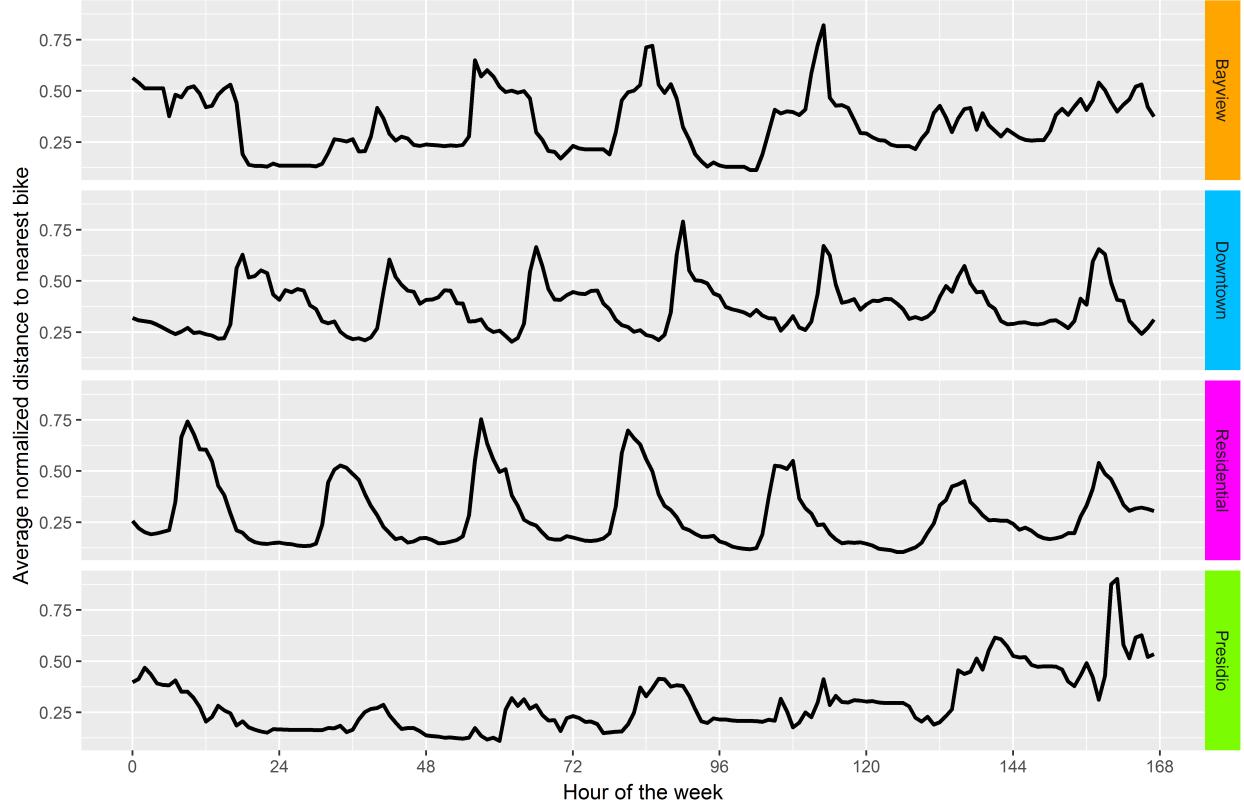


Figure 3: Normalized, average weekly patterns of the grid cell distance data, averaged per cluster

5.2 Model building

Figure 5.4 shows the time plots of the distance data that were queried for each of the model points in Figure 5.2b, with the dark grey shaded areas representing weekends. The plots endorse the findings in the previous sections. The data corresponding to the Bayview model point show large variation, interspersed with flat sections, and lack a clear repeating pattern. The variance decreases to the end of the training period, together with the mean.

The data corresponding to the Downtown and Residential model points are most dynamic. A daily pattern shows for both of them, and in the Downtown data, the peaks in the weekend are generally lower than those on weekdays. For the Residential model point, however, this difference is less visible. In both datasets, the daily peaks vary considerably in height from day to day, and look far less smooth than the averaged weekly patterns shown in Figure 5.3. Although not as noticeable as for the Bayview model point, the variances of the Downtown and Residential data seem to decrease slightly towards the end of the training period, together with the mean. This justifies the use of a multiplicative decomposition, rather than an additive one.

The Presidio model point shows the most constant data, with a low mean and long flat sections. In the first part of the training period, the Sundays stand out clearly, but later, the peaks in the data get smaller, and seem to occur more randomly.

Finally, what strikes, is the large period of missing data in the last week of the training period. These data are missing in all datasets, probably caused by problems on the JUMP Bikes server. Some smaller periods of missing data occur in the first weekend of the training period, and during the second week of

November. Again, the data for all four time series are missing in these periods. Recall here that both STL and the implementation of ARIMA in R, as used by DBAFS, automatically handle missing values. Hence, the periods of missing data are not problematic for neither the model fitting nor the forecasting process.

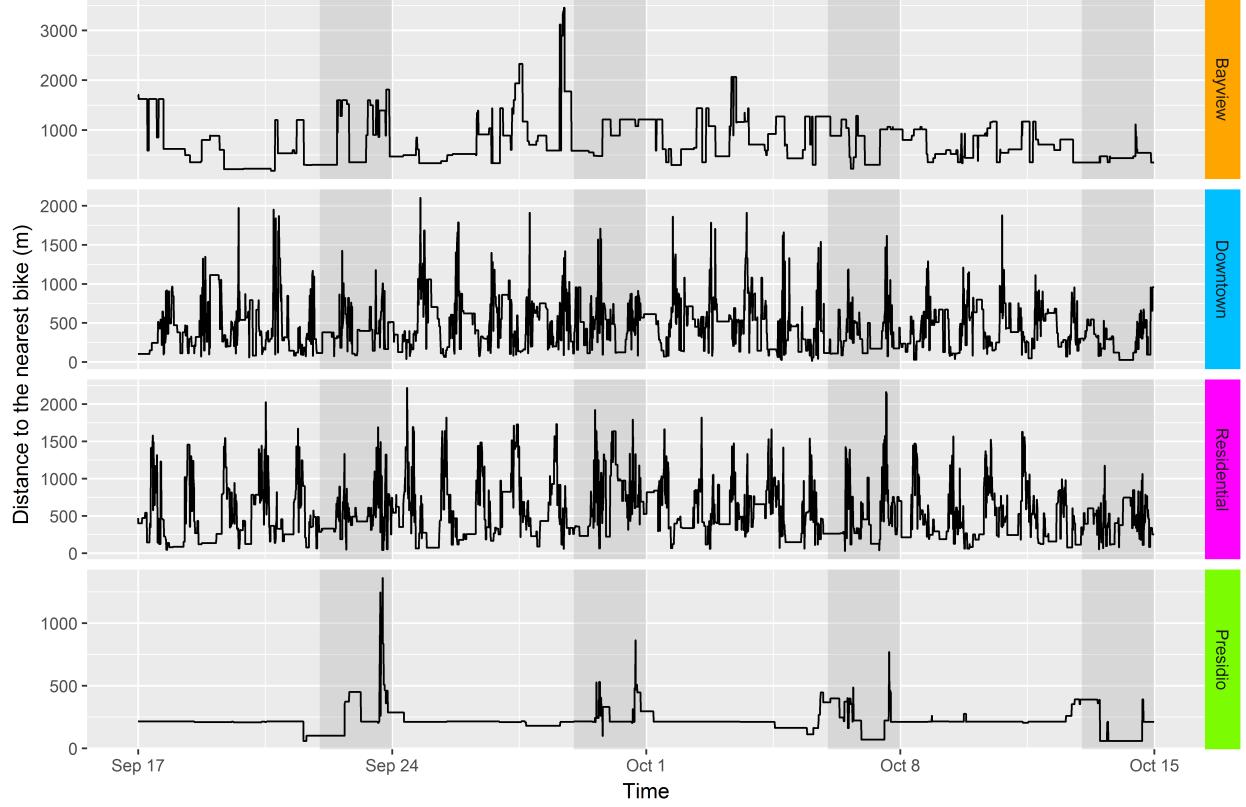


Figure 4: Time plots of the model points distance data

The structures of the models that were fitted on the four model point time series are shown in Table 5.2. The automatic seasonality detection resulted, as expected, in a double seasonal pattern for the Downtown data. For the Residential data, only a daily pattern was detected, while also the Bayview data were considered seasonal by the cross-validation process. Only Presidio data were labeled non-seasonal, and skipped the decomposition process sequence. Additionally to the seasonal periods, Table 5.2 also shows the strength F_s of the corresponding seasonal patterns, calculated with Equation 2.x. The daily pattern in the Bayview data is weak, just as the weekly pattern in the Downtown data. The seasonal strengths of the daily patterns in the Downtown data and the Residential data are larger, but still, none of them can be considered very strong.

NOTE: Maybe give the detailed results of the cross-validation?

In the ARIMA(p, d, q) models for the Bayview and Downtown data, no autoregressive terms are included. Instead, they contain a high order of moving average terms. All datasets passed the KPSS test for stationarity after one differencing operation. The full details of the components and fitted models, including parameter estimates, can be found in Appendix B.

Figure 5.5 shows the residuals of each model, plotted over time. All models have residuals with an approximately zero mean, and the variances look approximately constant. Comparing the residual time plot with the data time plot of Figure 5.4, it can be seen that in for the less dynamic data of the Bayview and Presidio model points, the models struggle to find a good fit for the peaks and valleys in the data, while the flat

sections are explained accurately.

Table 2: Model structures

	STL		ARIMA		
	seasonality	F_s	p	d	q
Bayview	daily	0.25	0	1	5
Downtown	daily and weekly	0.40, 0.27	0	1	5
Residential	daily	0.47	2	1	1
Presidio	none	-	1	1	1

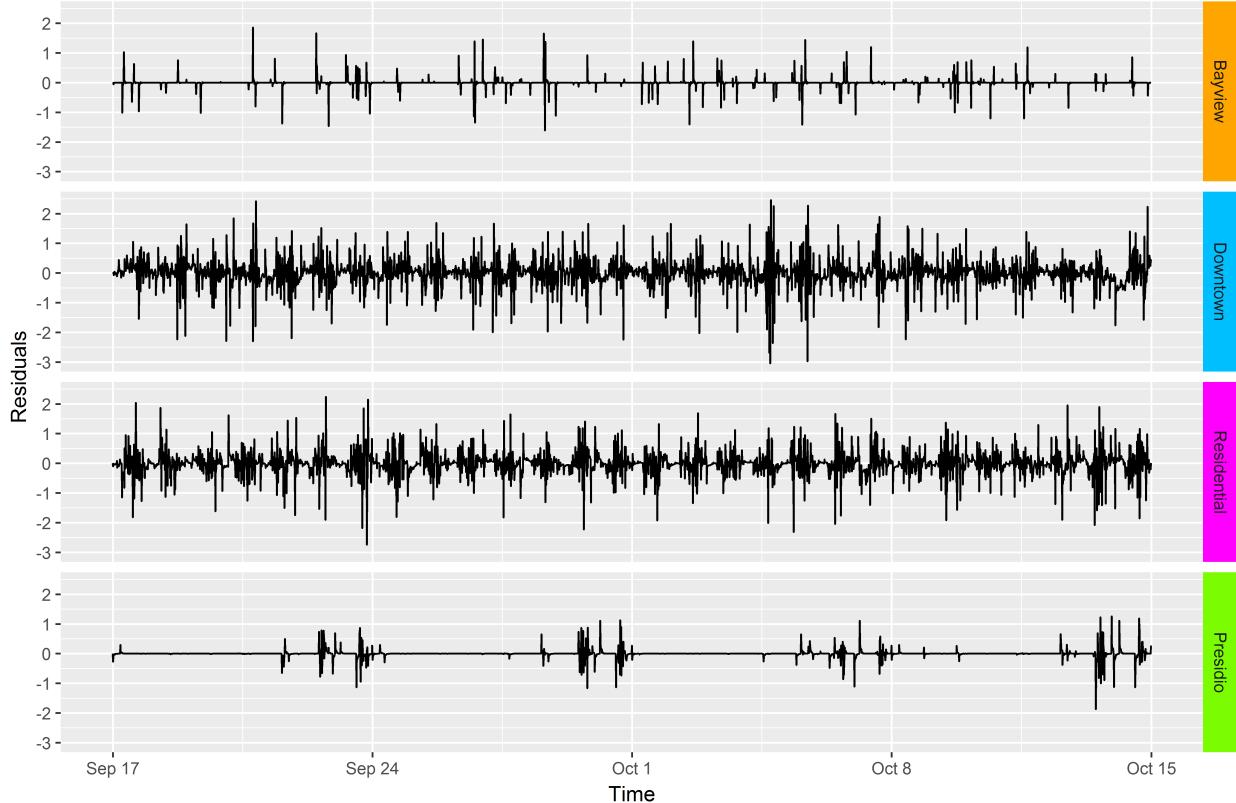


Figure 5: Time plots of the model residuals

The autocorrelations at several time lags in the residuals are shown in Figure 5.6. Since the data have a temporal resolution of 15 minutes, 96 time lags correspond to one day, and 672 time lags, the total span of the x-axis in the figure, to one week. The dotted orange lines form the lower and upper 95% confidence bounds, assuming a normal distribution. This means that the residuals are considered to be a realization of a white noise process when at least 95% of the autocorrelation values fall within these bounds. It is important to note here that when working with real-world data, finding perfectly random model residuals is an exception, especially when the data have a high entropy. Taking that into account, the autocorrelation plot of the Downtown model look good, and their residuals seem to approximate white noise.

However, for the Bayview and Residential models, and in lower extent also the Presidio model, autocorrelations show peaks at several lags that correspond to full days. This implies that there is still some information left in those data that the models did not capture. Recall that no weekly seasonal pattern was included in the

Bayview and Residential models. Only using a daily seasonality, systematic differences between weekdays will not be explained, which could be a possible reason for the significant autocorrelations that show up. In the Presidio, no seasonal pattern was identified at all. However, the seasonal patterns were chosen such that the RMSE of the forecasts in the cross-validation process was minimized, and in the end, minimizing these forecast errors is the real aim of DBAFS.

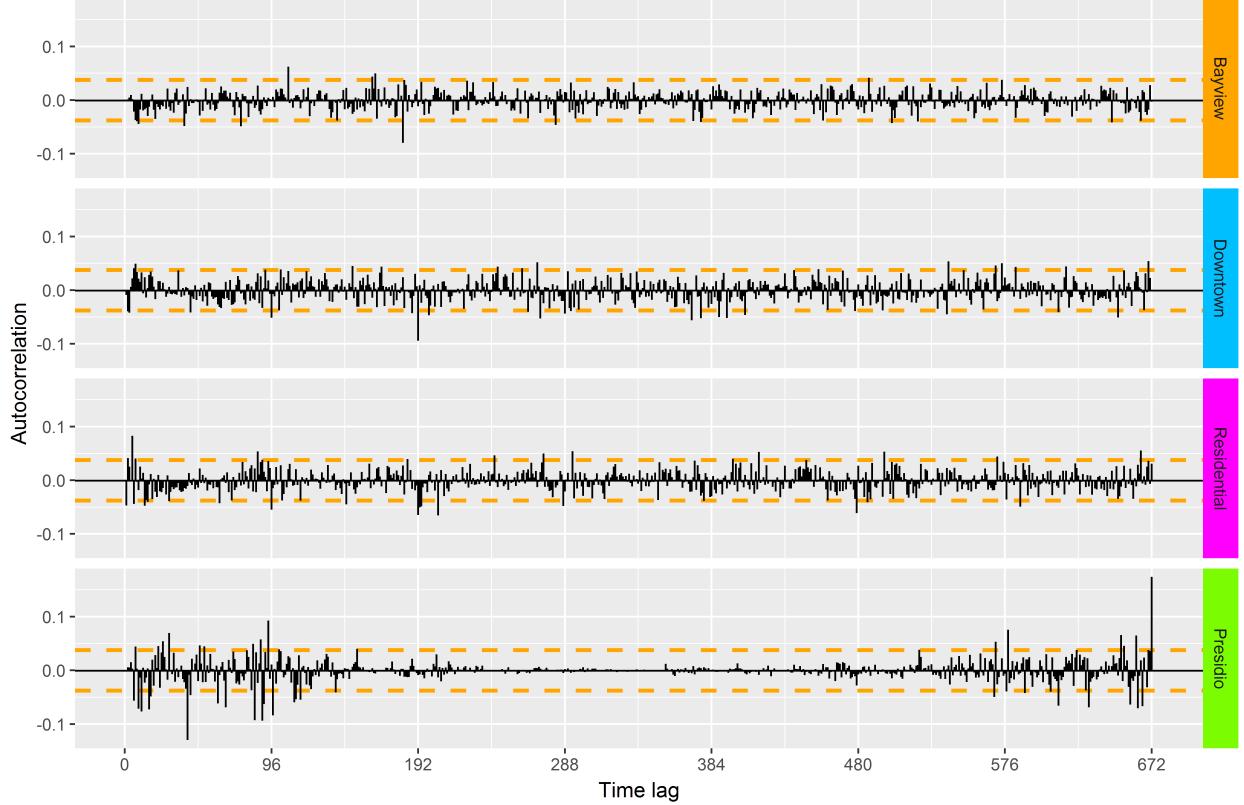


Figure 6: ACF plot of the model residuals

Finally, Figure 5.7 shows the histograms of the model residual distributions. As expected, for the Bayview and Presidio models, most values are clustered closely around the zero mean, with the tails being extremely thin and long, especially for the Bayview model. The residuals of the Downtown and Residential models follow a distribution that comes closer to a normal one, but also here, the tails are wide.

As discussed in section 2.4.2.4, using Gaussian likelihood is sensible even when non-normally distributed residuals show up. Of course, it could be a possibility to try different likelihood functions, but this will make the process much more complex and much slower, for, probably, only a little gain in forecast accuracy. The non-normality of the residuals does have an effect on the validity of the prediction intervals, however. When 95% prediction intervals are calculated, assuming normality of the forecast distribution, they can not be interpreted as such. This issue will be covered further in the next section.

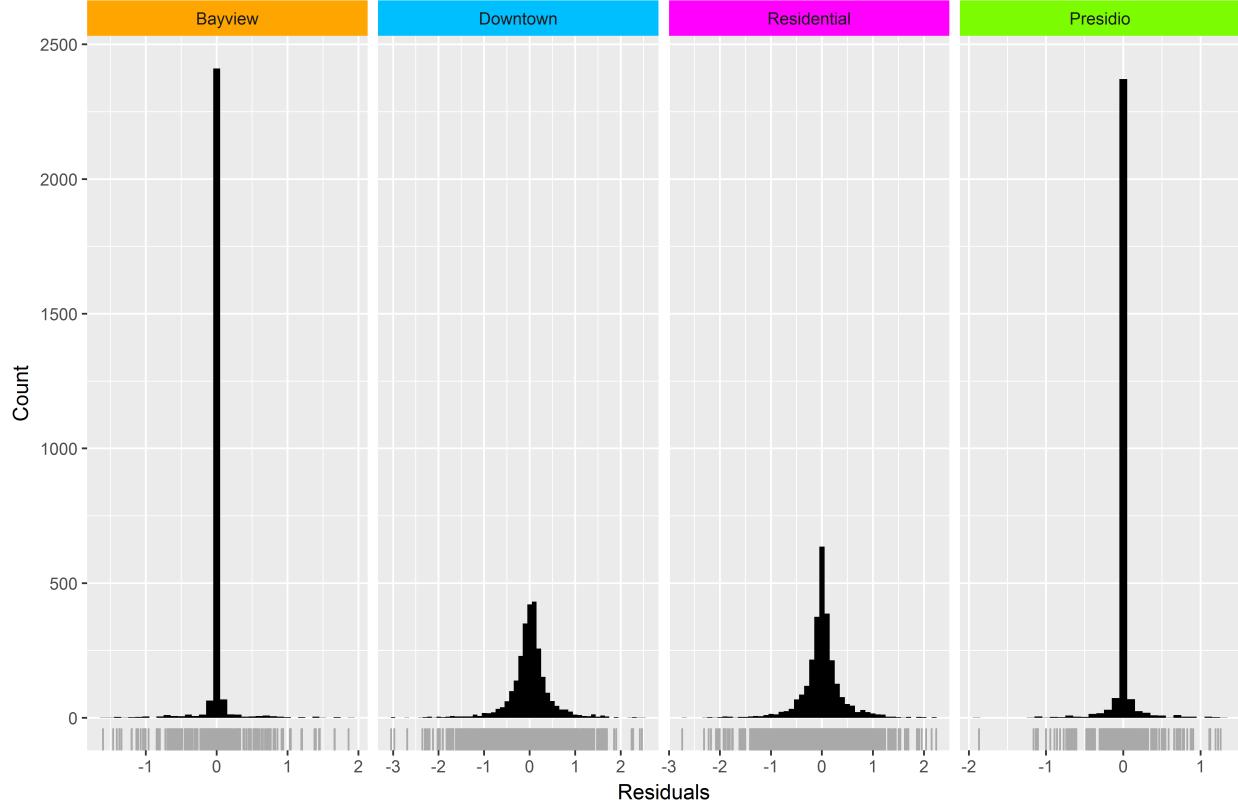


Figure 7: Histograms of the model residuals

5.3 Forecasting

Figure 5.8a shows the geographical locations of the 1000 test points. As planned, areas with high usage intensity have more test points, with almost 95% located in the Downtown and Residential clusters. Figure 5.8b shows the time stamps of the test points, counted per hour. The whole two week period is covered, with less test points during working times, and more during rush hours and in the evening. What strikes, is a high peak in the afternoon of the 4th of December, and the fact that relatively many test points occur during the night. The full information on the test set, with all unique location-time combinations, can be found in Appendix C. The RMSE and the MAE, both described in section 2.4.2.6, are calculated for the forecasts of each test point separately. Then, these values are averaged, either over the whole system area, or per cluster.

Table 5.3 lists those averaged RMSE's and MAE's of the forecasts produced by DBAFS, and of the forecasts produced with the naïve method. DBAFS outperforms the naïve method, but the accuracy gain is minimal. Where the average RMSE of DBAFS is 241 meters, the naïve method produces forecasts with RMSE's that are on average only 45 meters lower. In the Bayview cluster, an area with long flat sections in the data, the naïve method even outperforms DBAFS, with the side note that only ten test points were located here.

The range of RMSE values is much higher for naïve forecasts when compared to DBAFS. For some test points the naïve method manages to produce perfect forecasts, implying that the data were completely flat during that day. When a seasonal pattern is identified, DBAFS expects such a pattern to continue over time, and will forecast peaks even when the real data remains flat. However, there are also time points for which the naïve forecast resulted in an RMSE of almost 1.5 kilometers. For DBAFS, the RMSE is lower than 900 meters at all test points. Both in DBAFS and the naïve method, the average RMSE is considerably larger

in the Downtown and Residential clusters, just as the range between the minimum and maximum RMSE. This was expected, since those clusters showed the most dynamic and complex data.

The MAE can be interpreted more intuitive. When DBAFS produces a forecast, on average over the whole system area, the nearest available bike will either be 185 meters further, or 185 meters closer. Using the naïve methods, this distance is 222 meters.

Table 3: Forecast results

n	DBAfs						Naïve						
	RMSE			MAE			RMSE			MAE			
	mean	min	max	mean	min	max	mean	min	max	mean	min	max	
Total	500	282	38	1004	213	11	912	408	37	1644	322	11	1610
Bayview	10	389	38	1004	332	11	912	389	38	1004	332	11	912
Downtown	259	248	122	523	185	73	435	414	116	927	327	47	892
Residential	211	317	97	705	241	65	544	411	37	1644	321	28	1610
Presidio	20	299	80	577	227	61	442	320	175	699	253	55	666

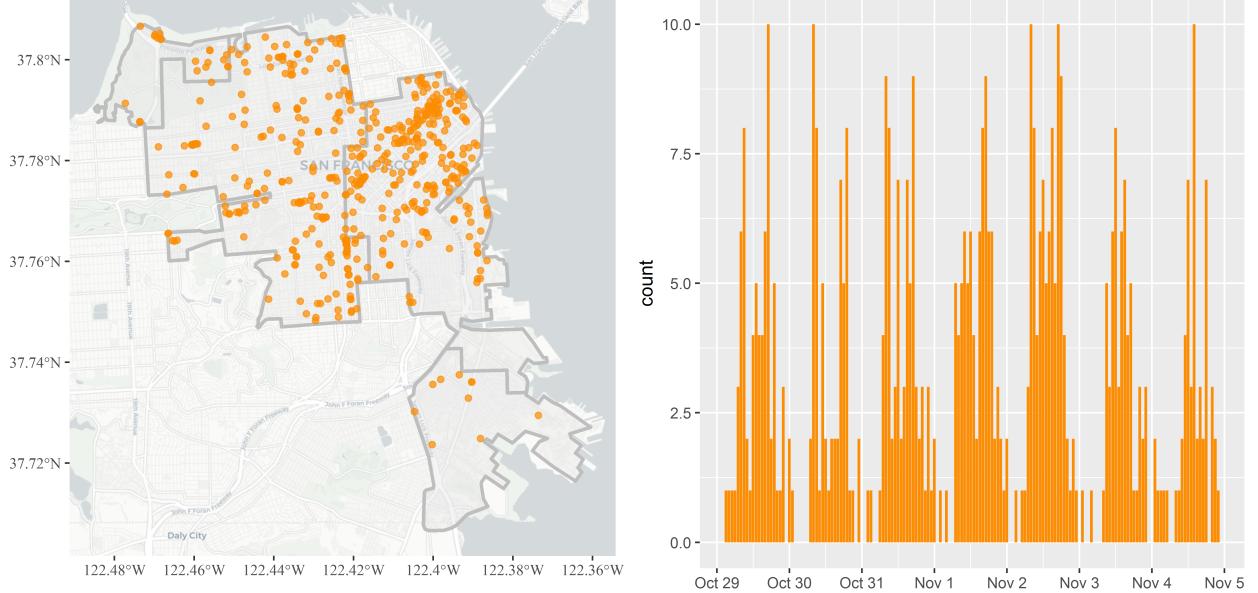


Figure 8: a) geographical locations of the test points; b) time stamps of the test points, counted per hour

The spatial and temporal patterns of the forecast RMSE values are shown together in Figure 5.9, where each point represents the RMSE of a single forecasted test point. The points are plotted with time on the x-axis, where the values correspond to the starting time of each forecast. The shaded areas are weekends. The patterns that show endorse the findings above. In the Bayview cluster, only a few forecasts are made, all with a relatively low RMSE, and little difference between DBAfs and the Naïve method. To less extent, the same yields for the Presidio cluster. In the Downtown and Residential clusters, where the majority of the forecasts are made, the RMSE's of the forecasts from DBAfs are on average just slightly lower than those from the Naïve method, and more compact.

Especially in the Downtown cluster, the RMSE's of forecasts in the weekend are lower than during weekdays, both for DBAfs and the Naïve method. The average of the RMSE's is higher in the second week of the

test period, compared to the first. This was expected, since the second week is further away from the time at which the models were build. However, the difference is very small.

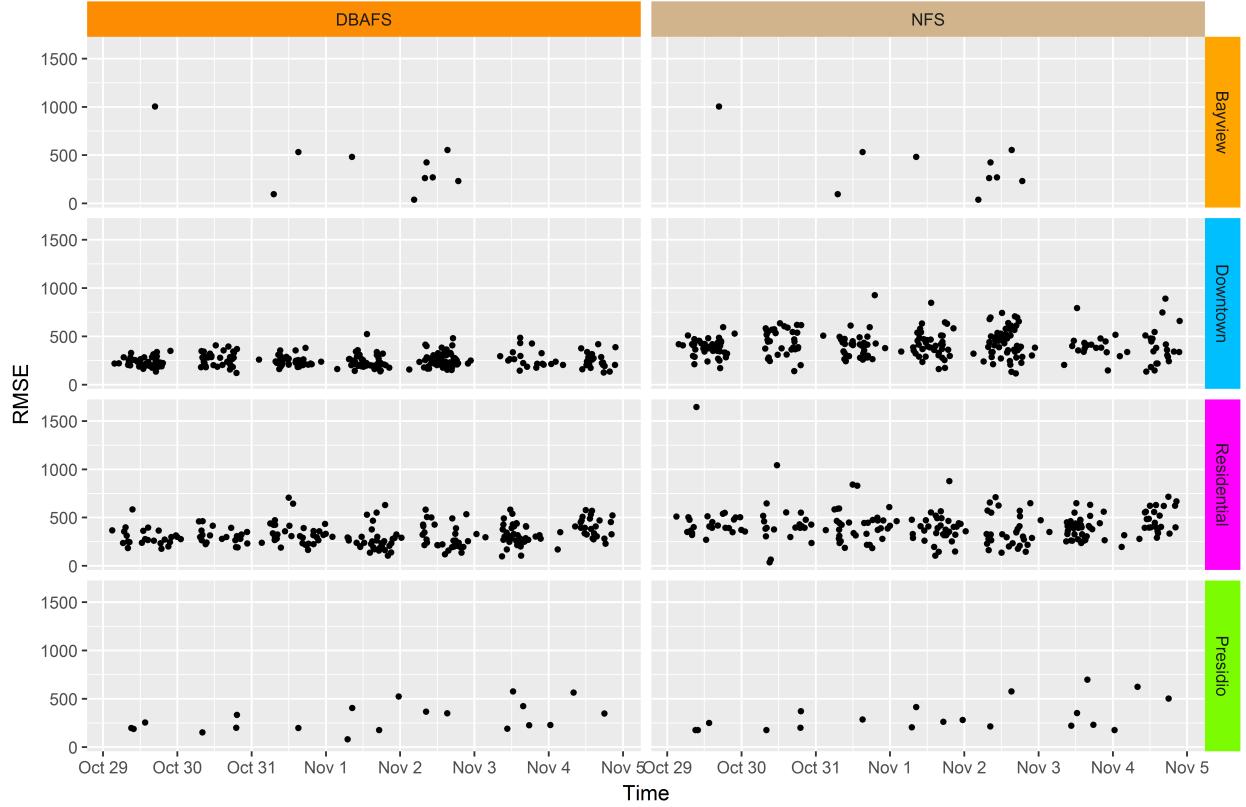


Figure 9: Spatio-temporal patterns of the forecast RMSEs

Figure 5.10a shows the average RMSE per hour of the day. During the night time, the errors are low for both DBAFS and the Naïve method, followed by a strong increase during the morning peak hour. The afternoon peak hour shows lower RMSE's, and again more similarity between the two methods. During times where the average RMSE is high, and apparently, the data have a low forecastability, DBAFS performs considerably better than the naïve method.

Figure 5.10b shows the average RMSE per forecast lag. For the naïve method, the forecast error at lag 0 is always 0, since it simply forecasts the last known value of the time series. When using DBAFS, this is not necessarily the case, since the forecast is a function of past observations and noise values, and additionally, the corresponding seasonal term of that time stamp is added. The identified patterns in the data are in most cases not strong and constant enough to enable forecasts without any error. Therefore, in short term forecasting, the Naïve method performs slightly better, although for both methods the RMSE increases very fast during the first lags. After about two hours, the increase of RMSE for DBAFS starts flattening out, while for the Naïve method, it keeps increasing. Further into the future, the average RMSE's converge to the mean absolute error of the corresponding methods.

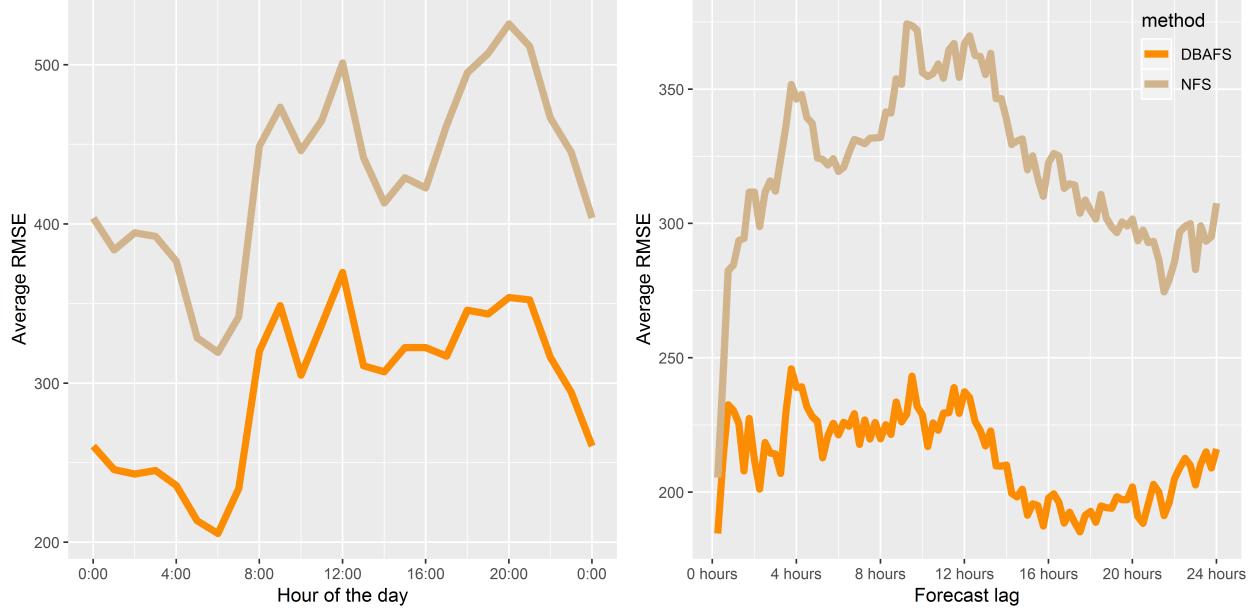


Figure 10: a) RMSE averaged per hour of the day; b) RMSE averaged per forecast lag

To provide detailed view of how the forecasts look like, Figure 5.11 shows the forecasts of one randomly selected test point per cluster. Some interesting observations can be made. The first forecast, in the Bayview cluster, shows an example where DBAFS expected a seasonal peak in the data to occur, but in reality, the data remained flat on that day. In such cases, the naïve method obviously gives better results. The second forecast, in the Downtown cluster, shows an example where DBAFS managed to accurately capture the dynamics of the data, and the outperforms the naïve method clearly. The third forecast, in the Residential cluster, shows a situation opposite to the one in the first plot. Here, an extreme and unusual peak in the data occurs, with distances higher than 2 kilometers. DBAFS can not capture such extreme cases, and expects peaks of a common size. The last forecast, in the Presidio cluster, is one where DBAFS used only the ARIMA model, and no seasonal decomposition. For this day, the DBAFS and naïve forecasts are very similar, and both don't recognize the small peak in the afternoon.

Finally, the prediction intervals of the forecasts should be interpreted. Table 5.4 shows the percentage of true observations that fall within the calculated 95% prediction interval of the forecasts. As can be seen, for all clusters where seasonality was detected, this percentage is extremely low. Hence, the uncertainty of the seasonal forecasts is so high, that the approach of only shifting the prediction intervals of the non-seasonal forecasts, without adjusting for the uncertainty of the seasonal forecasts, is far from appropriate. However, the uncertainty of the seasonal forecasts is hard to quantify, since no parameters are estimated there.

For the Presidio cluster, the only one where no seasonality was involved in the forecasts, approximately 65% of the observations fall within the calculated 95% prediction interval. This happens because normality was assumed in the interval calculation, while the residuals of the model did showed a distribution with wider tails than would be expected in a normal distribution. Therefore, the 95% prediction intervals of these forecasts should be interpreted as 65% prediction intervals instead.

Table 4: Interpretation of the calculated prediction intervals

	Total	Bayview	Downtown	Residential	Presidio
Percentage of observations within 95% prediction interval	0.9	17.5	0.5	0.6	0.1

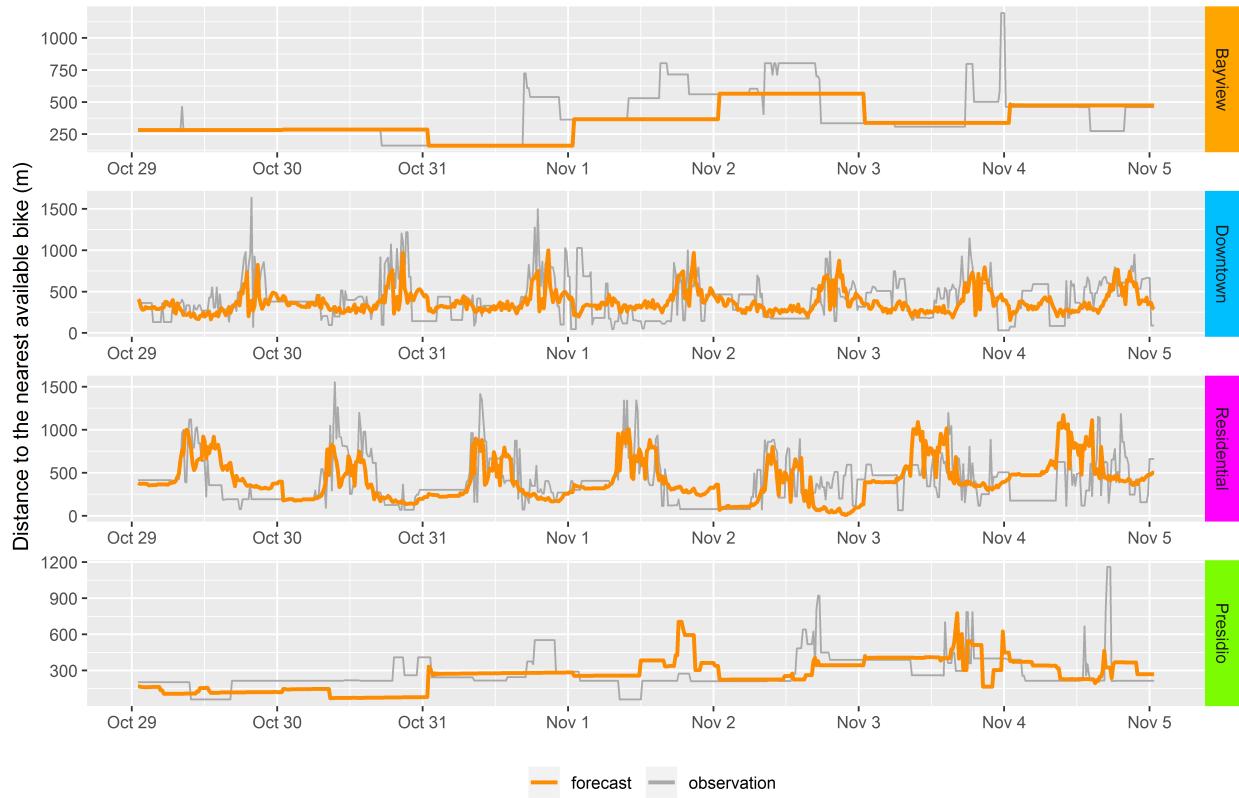


Figure 11: Forecast of one randomly selected test point per cluster

5.4 Computation times

NOTE: show the results of the comparison of computation times of the forecasts.

5.5 Limitations & Recommendations

*NOTE: write about limitation of DBAFS, and give recommendation how things could be improved. Cover at least the following points:

- the low forecastability of dockless bike sharing data in general, compared to station based bike sharing systems
- methods that can include exogenous variables. For example, the new forecast packages in R, that are still in development, like FASSTER, and a Dynamic Harmonic Regression with seasonality in Fourier terms and an ARIMA model for the errors
- talk about possible exogenous variables, based on the literature. Weather, special events like football matches, holidays, etcetera
- machine learning possibilities
- higher density of model points
- clustering based on ARIMA parameters instead of raw data*

References

San Francisco Department of Public Health. 2014. "The San Francisco Indicator Project." <https://www.sfindicatorproject.org/>.