University of Cologne

Semester Project – Nextbike

Report

Team Name and Student ID:

Janik Muires, 7317085

Tim Ockenga, 7322154

Tom Celig, 7322863

Maximilian Kaiser, 7365172

Diana Gowalow, 7369900

Supervisor: Philipp Artur Kienscherf, M. Sc.

Department of Information Systems for Sustainable Society Faculty of Management, Economics and Social Sciences University of Cologne

June 10, 2020

Content

Executive Summary

Nowadays, the demand for more environmentally friendly mobility increases steadily in order to meet the UN's goals of reducing our carbon footprint. The goal of this project is to investigate how one of the most successful mobility-as-a-service operators, NextBike, can benefit from raised and analyzed data by ultimately predicting the bike trip duration, the bike demand and the direction of the journey (depending on specific locations). Through extensive data exploration and insightful visualizations, we are able to provide NextBike with business recommendations for profit maximization and operating cost reduction. Our final data set consists of the adjusted data from NextBike, as well as location and weather data for the respective, engineered trips. For each predictive task, we determine relevant features by assessing correlations with the target variable in order to ensure generalizability of our models. The resulting predictive models (polynomial regressions, random forests) are evaluated and assessed using in-sample as well as out-of-sample data. The resulting performance metrics are mixed. Although predicting the trip duration turned out to be equally unreliable in-sample and out-of-sample, the out-of-sample accuracy of the trip direction classifier scored similarly well to the in-sample accuracy. Our bike demand prediction models achieved overall worse metrics in comparison to the in-sample results, yet, in some temporal resolutions (e.g. six and twelve hours), they did not deviate far away from their in-sample prediction quality. In conclusion, we were able to provide NextBike with valuable insights into customer demand patterns, as well as journey patterns. This information can be used to maximize profits, for instance by adjusting the pricing mechanism based on weekdays. We also suggest NextBike to collect supplementary user-related data for each bike reservation (e.g. customer status, used tariff, driven distance), which could be clustered to design an optimized and attractive pricing system for individual customer segments. This would help gaining and retaining more customers. Furthermore, we provided NextBike with tools to predict the bike demand (based on temporal and weather data) as well as the customer journey direction (based on location, temporal and weather data). Both tools are helpful to optimize the bike allocation and, thus, decrease NextBikes' operating costs.

List of Tables

List of Figures

1 Introduction

In the past years, mobility-as-a-service (MaaS) and mobility-on-demand (MoD) offers gained more and more attention since the necessity of alternative, sustainable mobility increases in order to avoid pollution, high emission of Carbon, inefficient utilization of passenger cars and decrease road accidents. The goal of this project is to investigate how fleet operators can benefit from raised data in order to increase their service reliability and, thus, gain and retain more customers. For this, we have been provided with a data set of a bike sharing platform operator named NextBike, which supplies bike rentals in Bremen. NextBike provides its bikes station-based, as well as free-floating. In this report, we will first describe and explore the data set to develop a deep understanding of the data. As we were confronted with missing values and false measurements, we conducted a thorough cleaning. Afterwards we visualize interesting aspects. These steps will provide an indication on which features to use for our predictive analysis. Based on these tasks, we will predict the duration, demand and the direction of the journey (depending on the location of University of Bremen and the main station). To accomplish our forecast goals, location data of Bremen is used. Additionally, we merge the data set with relevant weather data provided by the German weather service (DWD) to further improve our predictive performance. Afterwards, we describe the modelling and validation process of our prediction models. Finally, we will evaluate our predictive models based on an out-of-sample test data set and give business recommendations for the fleet operator. The Python package that enabled us to conduct this analysis user-friendly and reproducible, is also presented.

2 Data Understanding

2.1 Data Description

Our data set contains 1.279.878 data points with thirteen attributes (see table 1). The prefixes p and b stand for place and bike. Place defines the actual locations where bikes are rented or returned. It splits up into fixed stations and a free-floating zone. Bike stands for a class of bike-related information (e.g. bike identification).

Attribute Name	Data Type	Description
p_spot	boolean	if it is an official station: True, we have a name of fixed stations -> opposite of p_bike
p_place_type	int64	ID for type of place, 0 = fixed station, 12 = flex zone, 3 = special case: group booking, 14 = e-bike station
datetime	datetime64[ns]	The date of a booking
b_number	int64	ID of the bike (443 total number of bikes)
trip	object	states: first, last, start, end
p_uid	int64	unique ID of location (our data includes 76 out of 83 ids of fixed stations)
p_bikes	int64	number of available bikes at position/station
p_lat	float64	Latitude of a current location
b_bike_type	int64	ID of the bike type (bike types 71, 29)
p_name	object	Name of location
p_number	float64	short ID for fixed stations, if 0 it is a flex zone
p_lng	float64	Longitude of a current location
p_bike	boolean	opposite of p_spot, free-floating=True, station-based=False

Table 1: Original attributes of our Bremen data set

2.2 Trip Column Description

As table 1 shows, there are four different states for the trip column. We conclude that start and end pings (records in the data) are bookings where the bike was rented and returned on the same day. First and last pings emit when a day ends and starts. We assume these are status pings. First and last pings also occur, in case a trip started on one day and ended on the next day (see Appendix A). In this case, an additional change of location is the indicator that differentiates from a mere status ping.

3 Creating the Trip Dataset

First, we filtered the data set so that it only includes data points located in Bremen. The resulting data set contains 1,279,966 data points. We have a total number of 88 null values for p_number. As this is a very small number compared to all data points, we dropped those rows. Moreover, we checked for duplicates and discovered that half of the records are one duplicate per data point. Therefore, we dropped all duplicates leaving us with 639,939 data points.

Afterwards, we combined the trip states to generate data points that resemble true trips with a start and end destination, creating 228,949 trips.

After that, we analyzed round trips where the longitude and latitude of the start and end destination are the same. From 228,949 trips approximately 37.5% are round trips. Approximately 50% of those round trips do not start/end at a fixed station but within the flex zone. This means the bike would have been rented and returned at the same geographic coordinates, which is highly unlikely and led us to drop these trips as well.

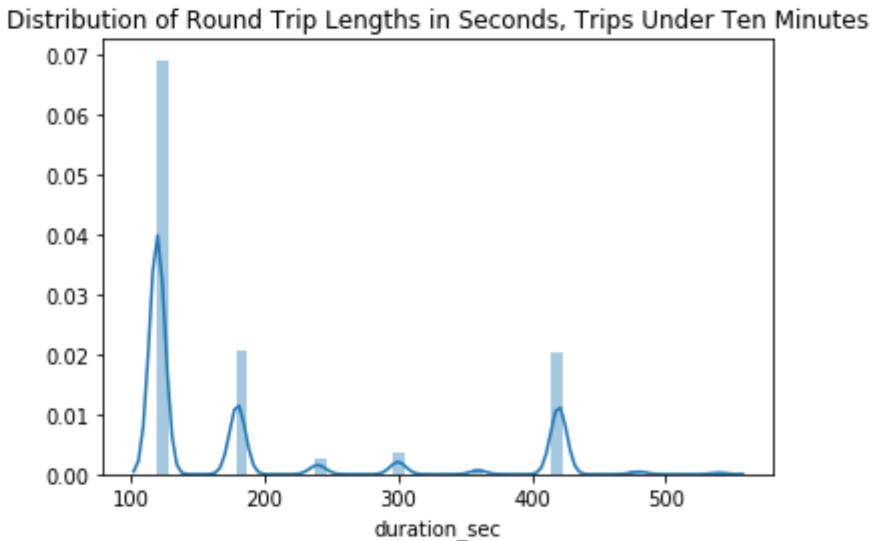


Figure 1: Distribution of round trips under ten minutes

Figure 1 shows the distribution of round trips with a trip duration fewer than ten minutes. As we can see, station-bounded round trips are two to seven minutes long. According to the FAQs from NextBike [1] a rental transaction that is rejected (because a lock does not open e.g.) is automatically returned after 3 minutes. We assume that such mechanisms act on minutes two to seven due to connection or technical issues. We dropped all round trips shorter than or equal to seven minutes as they are accurate to the second. 4,495 round trips remain. An even deeper analysis showed that we have several peaks, where round trips are unusually high compared to other dates (see Appendix B). We could not find information about particular events on these days, that could have led to these abnormalities. Consequently, we decided to drop round trips

[1] see "Das Rahmenschloss öffnet nicht. Was soll ich tun?“ https://www.nextbike.de/de/faq/

showing such abnormalities by automating it. Every round trip that deviates positively more than three standard deviations away from the mean will be dropped. Now 3,745 round trips remain, which equals approximately 2.6% of 146,822 trips in total.

4 Data Exploration

For generalizing the usage of our predictive methods, we built a preprocessor package, which returns a data set prepared for our prediction task. Therefore, we conducted an intensive descriptive data exploration.

4.1 Initial Feature Creation

We created a new DataFrame with more intuitive variable names containing the required information: bike number (bike), start time (start_time), weekend, start position (start_place), duration (duration_sec), end time (end_time), end position (end_place). In addition to the NextBike data set, we imported GeoJSON data of Bremen administrative boundaries, NextBike service zone and station information, to include the postcodes and station names. Throughout our analysis we added the following columns:

Attribute Name	**Description**
start_lng	geo data (longitude) where a trip started (generated from p_lng)
start_lat	geo data (latitude) where a trip started (generated from p_lat)
end_lng	geo data (longitude) where a trip ended (generated from p_lng)
end_lat	geo data (latitude) where a trip ended (generated from p_lat)
start_plz	from which region in bremen (postcode) started (generated from plz column)
end_plz	in which region in bremen (postcode) ended (generated from plz column)
duration_min	Seconds can be too abstract to understand immediately, therefore we changed it in minutes
month	month [1,...,12]
booking_date	date format "01-01-2019", without information of time

start_name	the full name of the station starting the trip from (e.g. “Flughafen Bremen”)
end_name	the full name of the station ending the trip at (e.g. “Messe Bremen”)
hour	the hour of the day (e.g. for “01-02-2019 14:45:04” -> 14
weekdays	weekdays [0-6], 0 = monday
weekday	binary, 0 = weekday, 1 = weekend

Table 2: Additional attributes generated throughout the analysis

4.2 Aggregate Statistics for Various Temporal Resolutions

To identify possible features and further interesting business cases we started with trip duration. In general, we have a right-skewed distribution (*n*= 146,822, *mean* = 256, *median*= 13, *s* = 3,759). The huge difference between median and mean indicates outliers. The upper quantile is 23 and the maximum duration is 365,655 minutes (~8 months), which is very high. Figure 2 shows the distribution of all minutes over 2019.

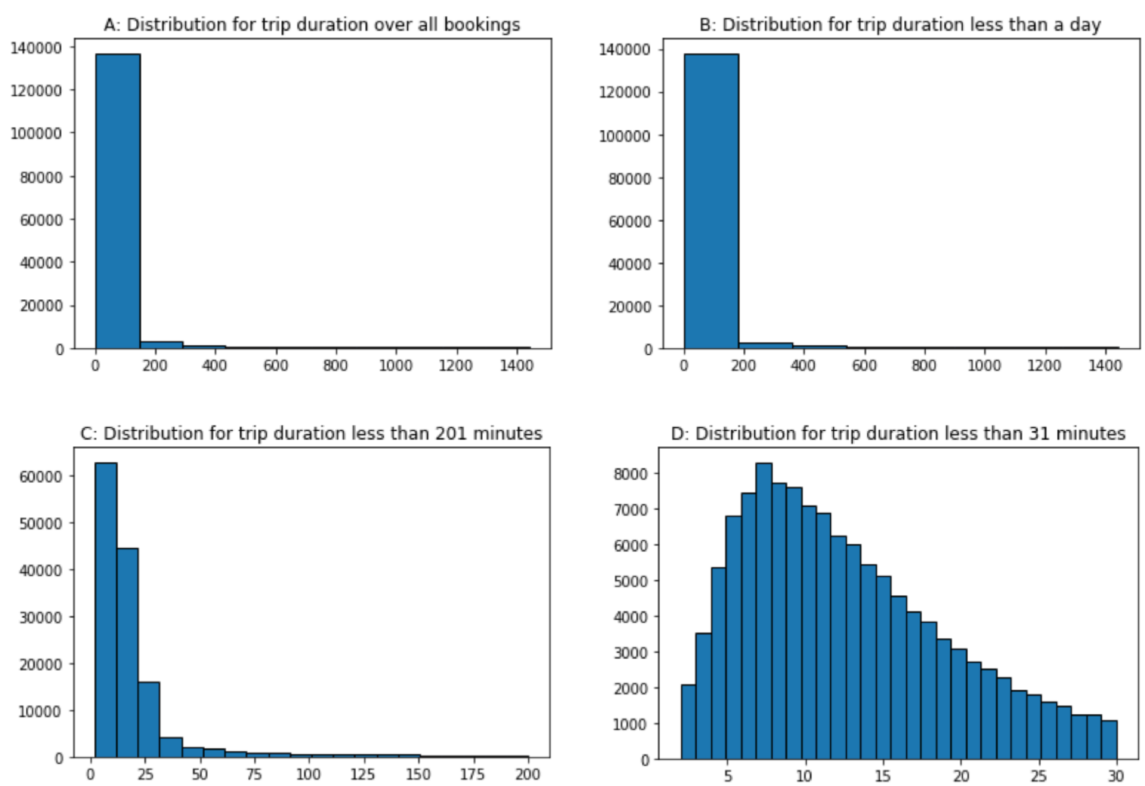


Figure 2: Distribution of trip duration of all bookings over 2019 for different trip durations

In figure 2a we actually cannot recognize anything, because most of the trips seem to be under 200 minutes and only a few take more than 200 minutes. Thus, we checked the distributions for all trips shorter or equal to one day (figure 2b), shorter than 201 minutes (figure 2c) and shorter than 31 minutes (figure 2d). 122,345 of the bookings have a duration of fewer than 30 minutes (85%) and half of the trips are short trips up to ~10-15 minutes. Only 2% of the trips are longer than one day. Due to the pricing structure of NextBike[2], we decided to drop those two percent. After dropping the outliers, we still have a right-skewed distribution (n= 143,879, *mean* = 40, *median*= 13, s = 126), but compared to our previous statistics our median and mean are way closer, and the standard deviation decreased drastically.

Next, we analyzed average trip duration, total trip duration and total bookings on a daily basis. First, we looked at the total daily booking minutes. Figure 3 shows that there are missing values and a huge peak in November.

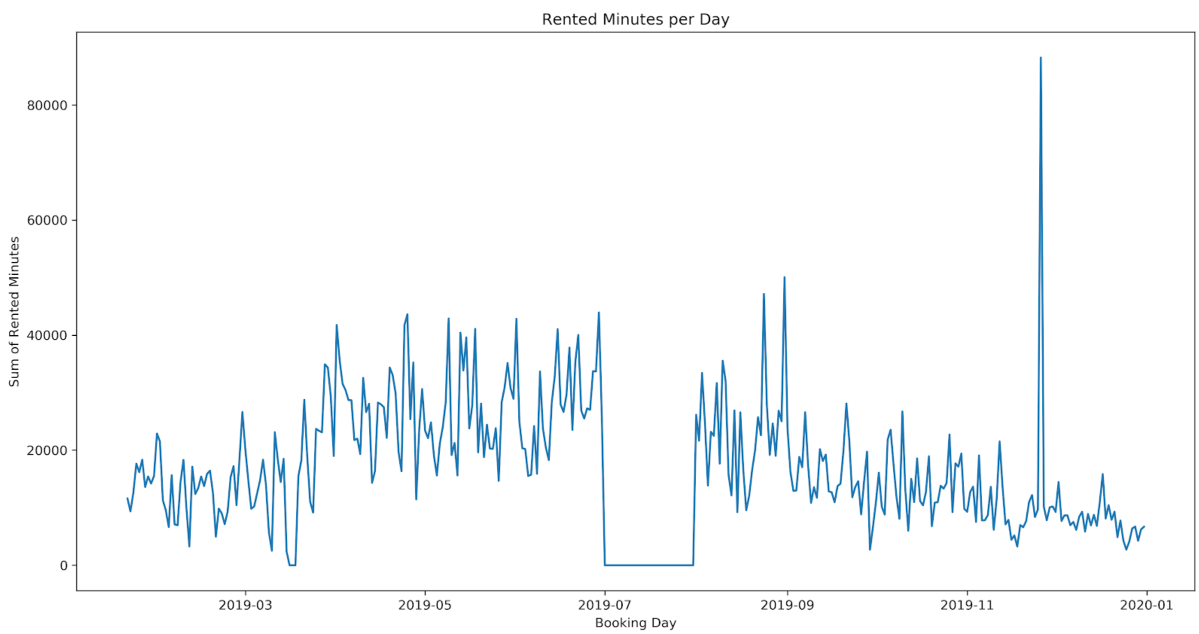


Figure 3: Total rented minutes over all bookings of 2019

Our analysis for November pointed out that we have data points where the start and end time are exactly alike but location related information differs randomly (see exploration notebook). This is different from the exact duplicates mentioned above. However, this use case seems very unusual, especially when it concerns 212 trips. We decided to use a cut off value of 10 and to drop these trips as well. Additionally, we noticed that data from the month of July is missing.

[2] https://www.nextbike.de/media/Preisverzeichnis_wk-bike_04-2020_dt.pdf

We could not impute the missing data, because we cannot generate trips related information such as geographic coordinates.

Second, we examined the trip duration on average (figure 4).

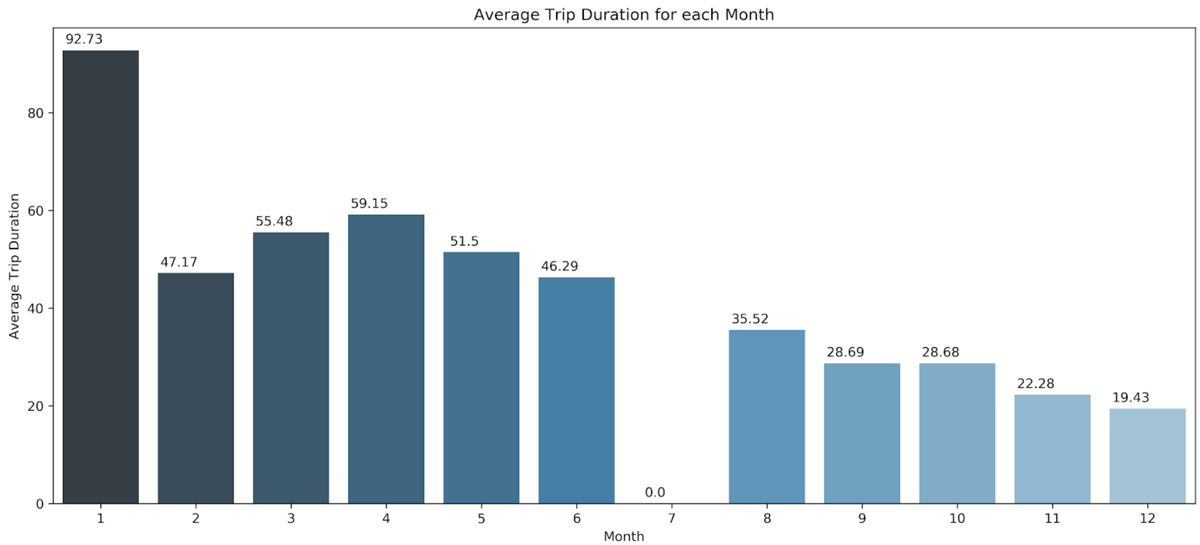


Figure 4: Average trip duration per month 2019

Strangely enough, the trip duration was by far the highest in January at approximately 92,73 minutes. We analyzed this issue extensively but concluded not to change or drop a data point. For January, we only have data points for eleven days. Yet, we have 72 trips that are longer than 800 minutes. Compared to February, we have 89 trips that are longer than 800 minutes but distributed over 28 days, rather than eleven days. We assume the high average duration in January is caused by this disproportionate distribution. However, the trips themselves do not seem unusual in January. Moreover, we compared the average trip duration per day for each month to a normal distribution and we found that our trip duration is not normally distributed. This can be an indicator that trip duration won't be a good indicator for a predictive analysis (e.g. linear regression). Because every month led to similar results, we will not go into further detail. Please refer to the exploration notebook for additional information.

Third, we looked at the total number of daily bookings. They show a seasonal trend, with more bookings during the summer months and fewer bookings during the winter months. We have a slightly left-skewed distribution (n= 312, $mean$ = 460, $median$= 464, s = 164), but the median and mean are very close, which indicates having no significant outliers. In figure 5 you can see the total amounts of bookings per month. The month august is the most booked month with 20,706 bookings.

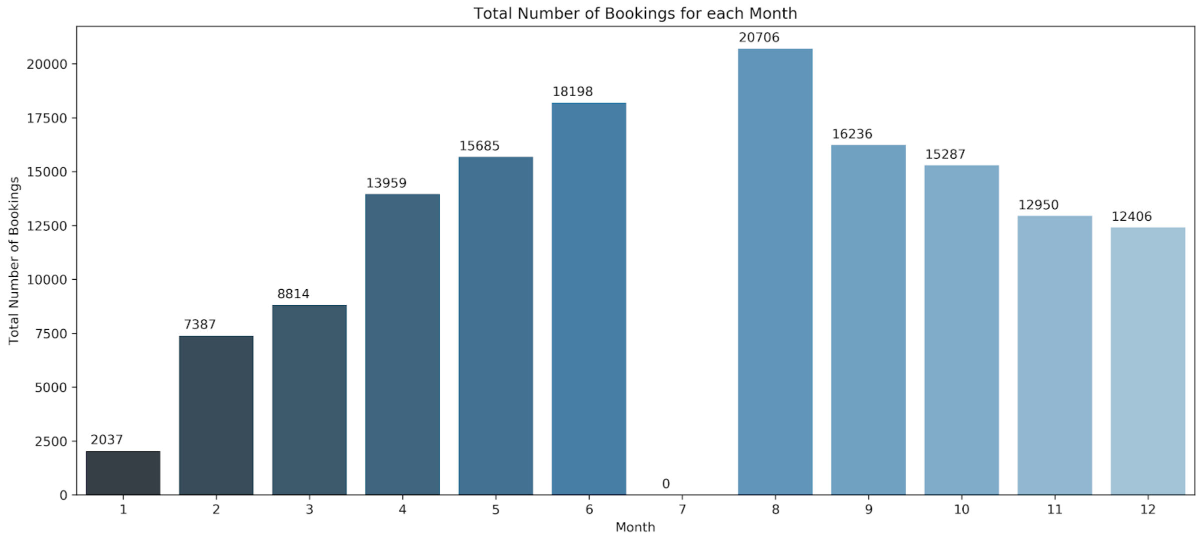


Figure 5: Total number of bookings per month 2019

Afterwards, we analyzed the number of bookings by days of the week (figure 6). The most bookings occur on Fridays and the fewest on Sundays.

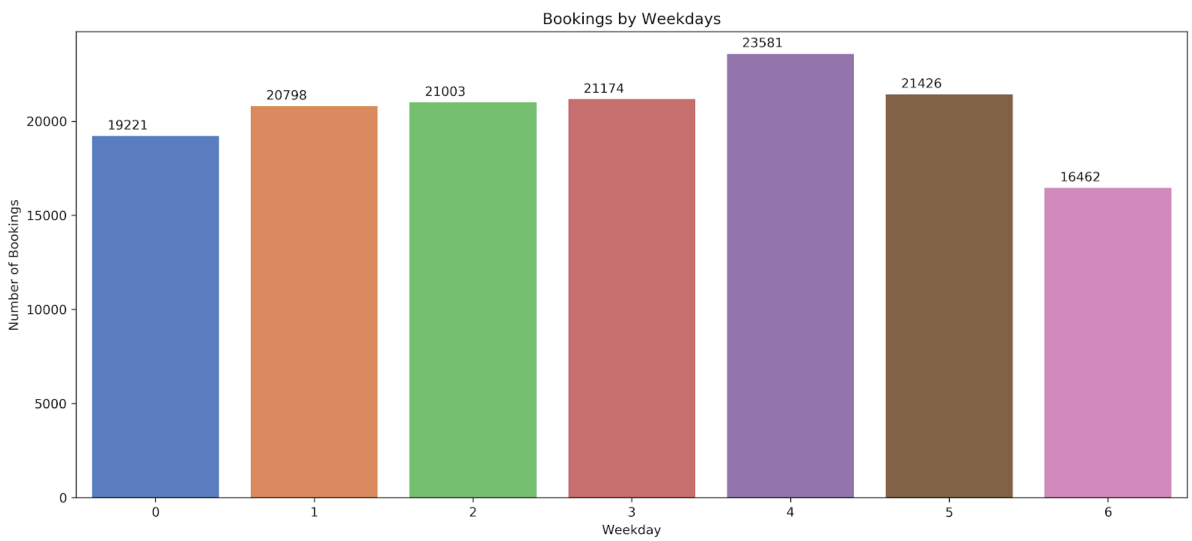


Figure 6: Total bookings by weekdays for 2019

We enhanced the weekly analysis by looking at bookings by the time of the day of weekday/weekend (figure 7). For the weekdays, we can identify a rush hour from 6 am to 7 am and 3 pm to 4 pm. Compared to the weekend, we generally have more bookings during the week and slightly fewer bookings during the night.

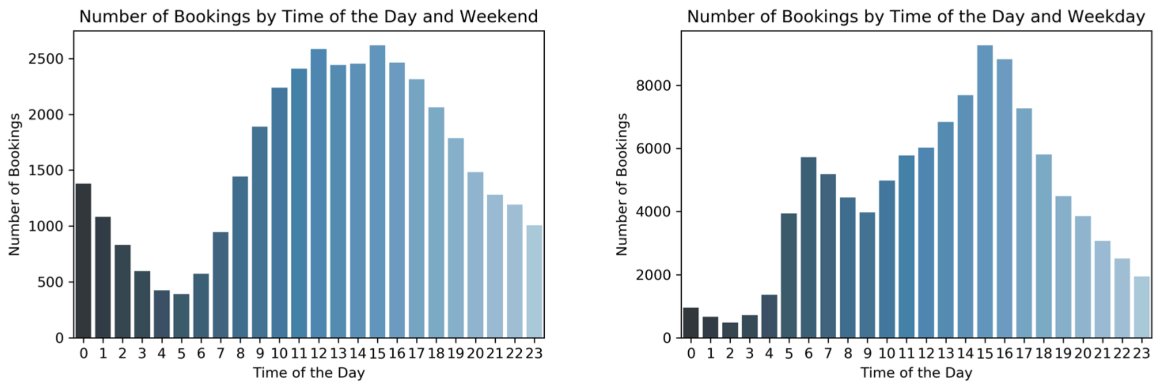


Figure 7: Number of bookings by time of the day and weekend(left) and weekday (right)

As a next step, we conducted a situational analysis. Table 3 shows the most popular routes. As most of the trips start at Hauptbahnhof/Übersee Museum (5,413), Weser-Kurier/PRessehaus (1,891), GOP/Steigenberger (1,347), Uni Bremen (988), those stations are represented more often.

	start_name	end_name	duration_min						
			count	min	max	mean	median	std	var
799	Hauptbahnhof / Übersee Museum	Hauptbahnhof / Übersee Museum	777	8.0	1418.0	98.720721	58.0	114.105657	13020.101026
1618	WESER-KURIER | Pressehaus	WESER-KURIER | Pressehaus	250	8.0	707.0	96.760000	46.5	116.025160	13461.837751
517	GOP / Steigenberger	GOP / Steigenberger	216	8.0	714.0	121.814815	62.5	139.679577	19510.384152
1294	Park Hotel	Park Hotel	164	8.0	850.0	139.890244	95.5	139.636250	19498.282358
524	GOP / Steigenberger	Hauptbahnhof / Übersee Museum	125	7.0	555.0	38.368000	10.0	97.703678	9546.008645
825	Hauptbahnhof / Übersee Museum	WESER-KURIER | Pressehaus	123	5.0	1328.0	45.569106	8.0	166.661190	27775.952152
1503	Uni Bremen	Uni Bremen	114	8.0	615.0	131.412281	93.5	118.018284	13928.315246
1567	Universum Bremen	Universum Bremen	109	8.0	1335.0	193.431193	112.0	211.798269	44858.506796
465	GEWOBA | Kurt-Schumacher 11	GEWOBA | Kurt-Schumacher 11	104	8.0	839.0	176.153846	114.0	199.557120	39823.044063
1592	WESER-KURIER | Pressehaus	Hauptbahnhof / Übersee Museum	96	5.0	723.0	38.593750	8.0	99.522817	9904.791118
1172	Mondelez Deutschland Services GmbH	Hauptbahnhof / Übersee Museum	92	11.0	58.0	16.586957	15.0	6.082134	36.992355
83	Anne Conway / Wohnheim	Anne Conway / Wohnheim	82	8.0	1141.0	176.170732	115.0	206.089325	42472.809997
789	Hauptbahnhof / Übersee Museum	GOP / Steigenberger	79	8.0	1070.0	47.151899	10.0	184.691294	34110.874067
963	Konsul-Smidt-Str.	Konsul-Smidt-Str.	73	8.0	570.0	145.287671	89.0	148.572137	22073.679985
710	Haltestelle Wartburgstraße	Flughafen Bremen	71	19.0	36.0	27.718310	27.0	2.752573	7.576660

Table 3: Most popular routes computed for trip duration

4.3 Exploring Started Trips per Postcode Region

We visualize the number of started trips per postcode region for both the whole year 2019 and August 2019. A comparison shows that there is no significant difference. In figure 8 you can see that most trips, 4,328 in total, start around the center of Bremen (postcode: 28195). In more remote areas (light yellow) only less than 541 trips start. Due to the fact that one postcode region itself comprises areas that differ in terms of demand patterns, we visualize the amount of trip starts in august 2019 per hexagon (resolution 9, using h3[3]) as well. For instance, figure 9 shows that there are hotspots within a postcode region, where most trips start from. Additionally, a hotspot hexagon can also be located within several postcode regions.

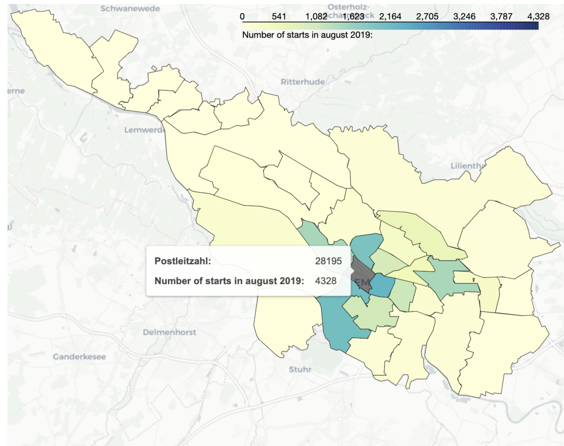


Figure 8: Map of Bremen visualizing the number of starts in august 2019 per postcode

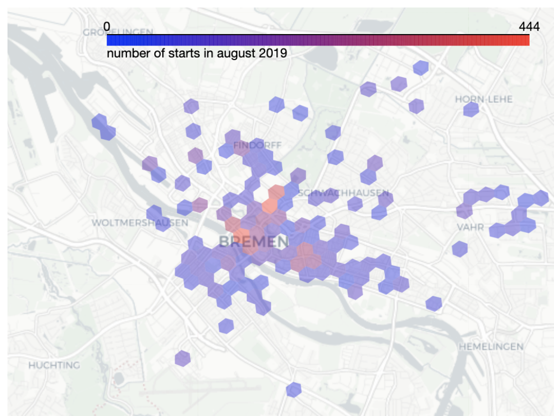


Figure 9: Map of Bremen visualizing the number of starts in august 2019 per hexagon (resolution 9)

4.4 Exploring Number of Bikes at Fixed Stations

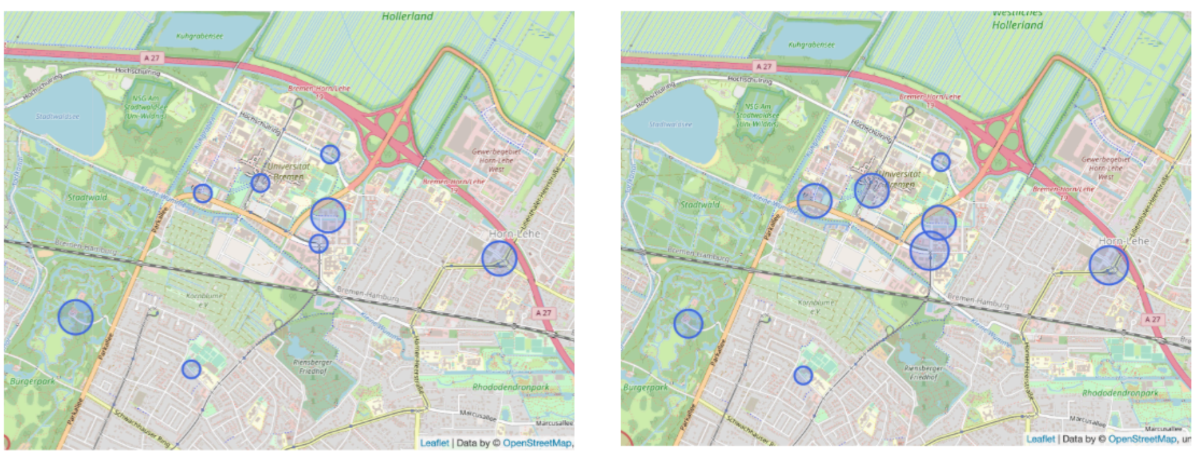
The station capacity ranges from 4 (hanseWasser Bremen GmbH (11119973)) to 8 (Park Hotel (8806546), GOP/Steigenberger (8806578)) bikes per station. Figure 10 shows an interactive bubble map (for the 20.09.19 12:00) with the radius of the bubbles indicating the number of bikes available at the station. By clicking on a bubble an information box pops up showing the station ID, station name and number of bikes available at this station. In this example, we are looking at “hanseWasser Bremen GmbH” with three out of four bikes available. Notebook #4-visualize-#bikes-at-stations provides a function that takes a timestamp and returns a data frame containing the corresponding bike availability per station.

---

[3] https://eng.uber.com/h3/

Figure 10: Bubble map showing the availability of bikes at fixed stations at the 20.09.19 12pm

As this type of visualization becomes more interesting when combining it with extra information we also looked at the bike availability at the University of Bremen when there was a concert (15th of June). Figure 11 clearly shows an increase of bike availability at NextBike stations located around the campus between 5pm and 6pm. This fits to the information that the concert started at 6pm.



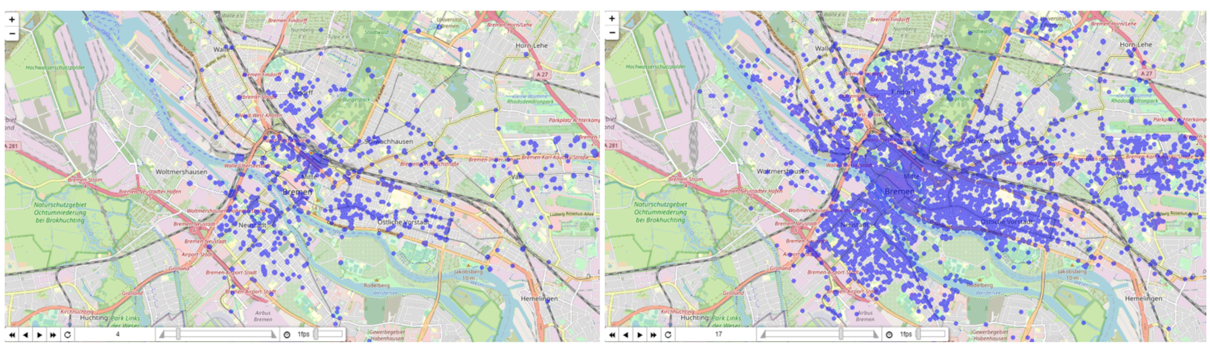
A: Bike availability at 5pm B: Bike availability at 6pm

Figure 11: Bubble maps showing the availability of bikes around the campus between 5pm (A) and 6pm(B)

4.5 Heatmap Visualizations for Dates of Interest

To find interesting points in time within our data set, we carried out a detailed analysis of days with the highest number of bike bookings (peaks) of each respective month. For each peak day, we performed a date-restricted google search for relevant news and major events. In case a major event could be determined, associated geographic coordinates were marked within corresponding heatmaps. This chapter will go into detail for a few selected interesting dates associated with major events. Besides, heatmaps were created for the entire period 2019, which provide interesting insights into the distribution of trips in hourly, daily and monthly temporal resolution. All mentioned (dynamic) heatmaps can be accessed in the exploration notebook. Trip start or end locations are visualized by blue dots which turn red in case of a trip duration above the 80th percentile.

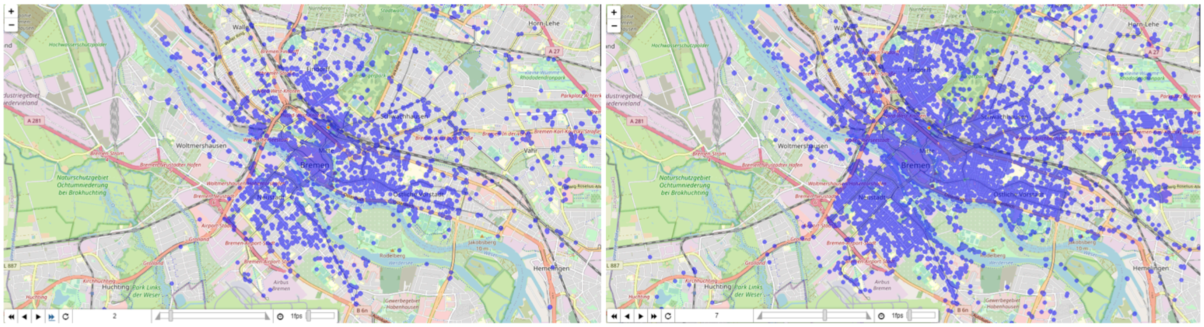
Figure 12 visualizes the hourly trip starts in the period 2019. The heatmap shows only a few starts during early mornings at 4 am (figure 12a) and becomes significantly denser during rush hours at 5 pm (figure 12b).



A: 4 am B: 5 pm

Figure 12: Hourly trip starts in the year of 2019 at 4 am (a) and at 5pm (b)

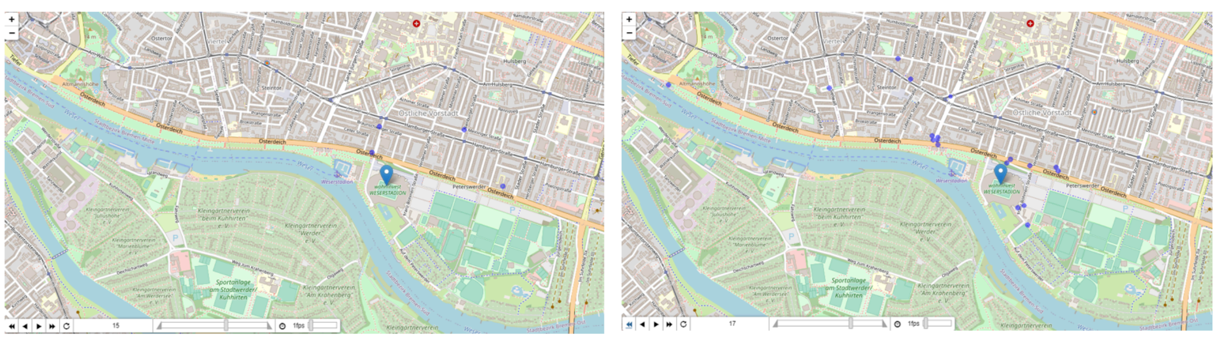
A similar observation can be made when visualizing trips starting in a monthly temporal resolution. Figure 13 shows a much denser heatmap in the summer month of August (figure 13b), compared to the winter month of February (figure 13a).



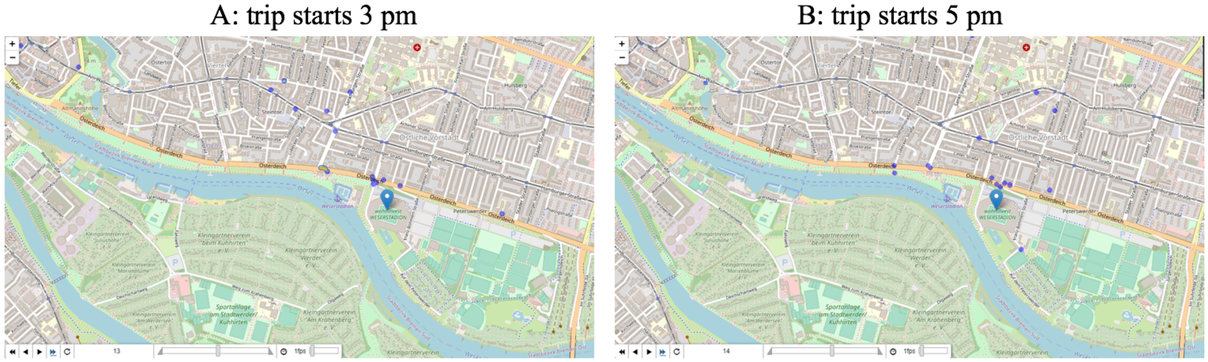
A: February B: August

Figure 13 Monthly trip starts in the year of 2019 for February (left) and August (right)

With 520 bookings, the 30th was the second-highest amount of bookings within the month of March. On that Saturday, a major football match took place at Weser-Stadion, which is marked in Figure 14. The match began at 3:30 pm and ended around 5:30 pm. We can see a few trips starting in surrounding areas of Weser stadium starting from 3 pm (14a), which are presumably people who live close by. After 5 pm we can see lots of trips starting in close areas around Weser stadium (14c), as the people start heading home. Furthermore, we can see a relatively high amount of trips ending from 1 pm until 3 pm around Weser stadium (14b,d), which suggests some fans took a NextBike to reach the location.



A: trip starts 3 pm B: trip starts 5 pm



C: trip ends 1 pm D: trip ends 2 pm

Figure 14: Trips at Weser-Stadion for certain points in time

With 681 bookings, **the 15th** was one of the highest amounts of bookings within the month of June. On that Saturday, three festivals took place. Two of them were located near the city center and one at the University campus as marked in both figures. The figure 15 depicts a heatmap with all trip endings which occurred on that Saturday. We can see hotspots near the marked festivals.

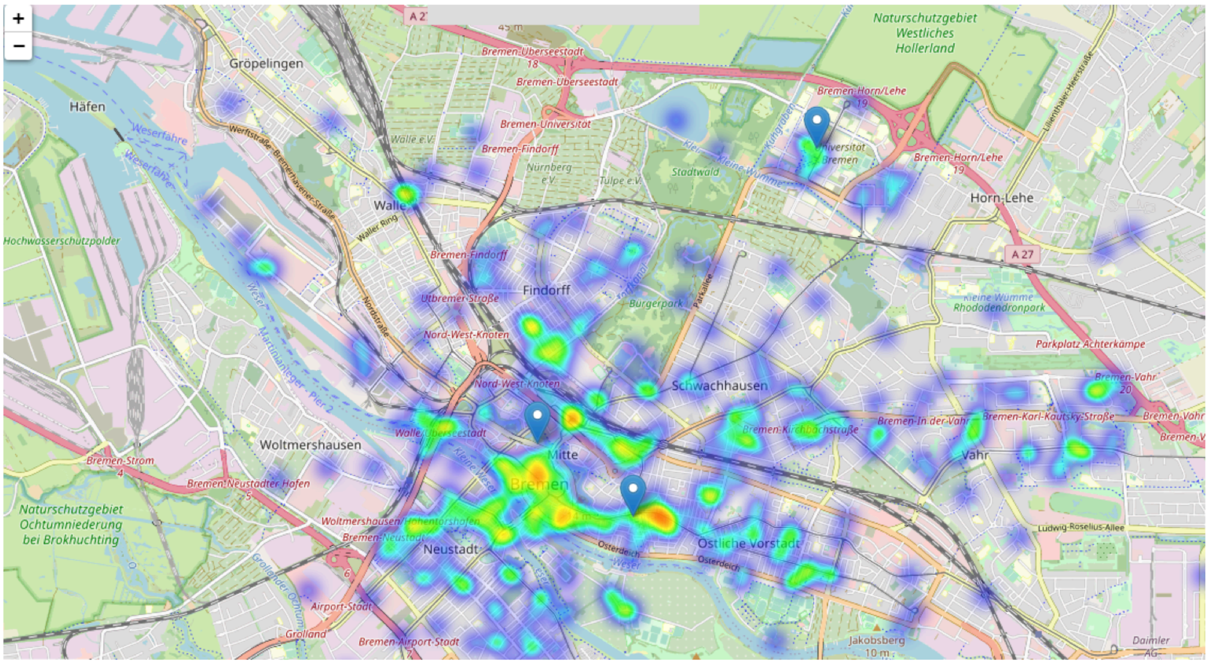
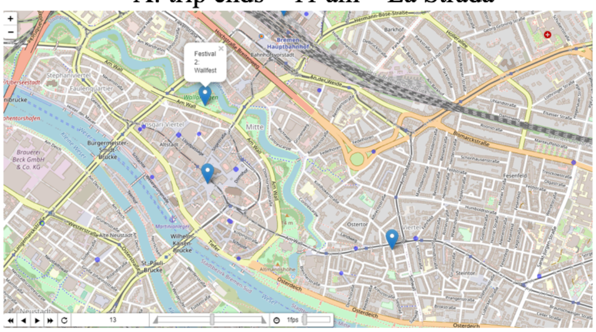
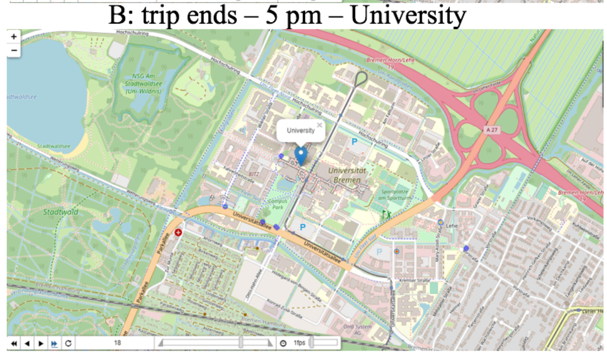


Figure 15: Trips to festival at 15th of June, 2019

Figure 16 depicts traffic near the festivals for interesting points in time. The "La Strada" (festival 1) and the "Wallfest" (festival 2) attracted a high number of trip endings starting from 11 am until 6 pm (16 a,c). Since these are street festivals, the markers do not indicate an exact location, rather than providing an indication. The "Open Campus" (festival 3) at the University of Bremen has a visible increase in trip endings shortly before the beginning of its main event - a concert. As the concert at festival 3 started at 7 pm, we can see several trip endings nearby at 6 pm 16b), a visible contrast to zero trip ends near the University at 5 pm (16d).


A: trip ends – 11 am – La Strada   B: trip ends – 5 pm – University

C: trip ends – 1 pm – Wallfest   D: trip ends – 6 pm – University

Figure 16: Traffic near festivals for certain point in time at 15th of July, 2019

4.6 Shortly Exploring the Weather Dataset

As one of our tasks is to predict the trip duration, we have thought about possible additional features that could increase our predictive accuracy. Therefore, we fetched weather data, as the choice of riding a bike presumably depends on good weather conditions. Please refer to table in in the appendix to see chosen features (see Appendix C).

First, we decided to have a look at the temperature. Figure 17 shows a scatter plot of the average daily temperature with a seasonal trend.

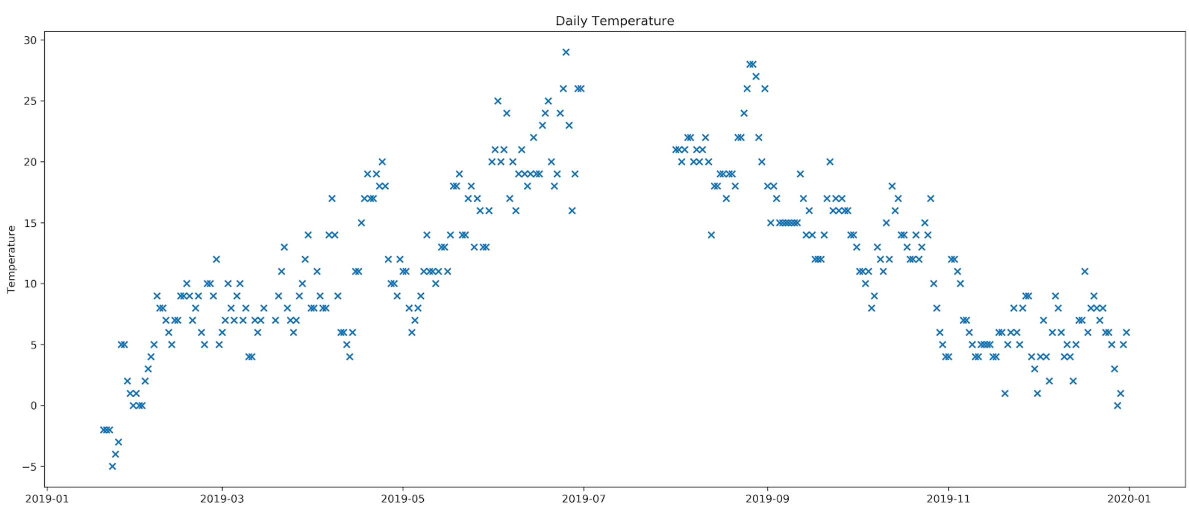


Figure 17: Scatter plot of average daily temperature for 2019

Afterward, we had a look at the scatter plot of average daily temperature and average trip duration (figure 18). Here, we can observe at most a weak correlation between average trip duration and average daily temperature. For the prediction, we need to have a closer look at the correlation matrix between weather data and trip duration, because average trip duration does not seem very promising.

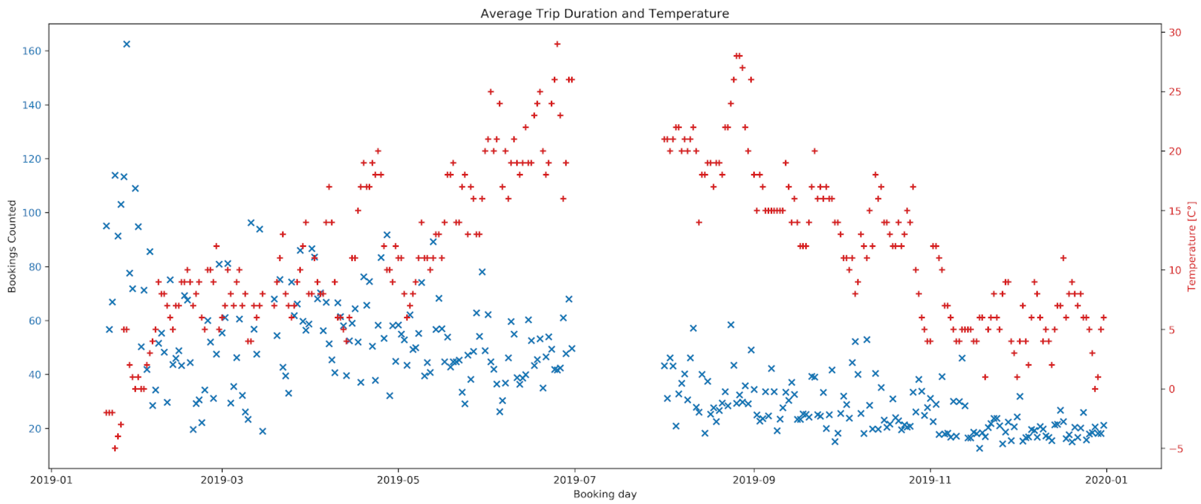


Figure 18: Scatter plot of average daily temperature for 2019 (red) and daily average trip duration for 2019 (blue)

Due to the fact that trip duration and temperature did not correlate well, we looked at the daily total rented minutes, which depict a seasonal trend as mentioned above. The scatter plot showed a positive correlation between daily total rented minutes and daily average temperature (see Appendix D). However, most promising was the scatter plot for average daily temperature and average daily demand (figure 19), which depicts a strong positive correlation. As this indicates a promising predictive analysis, we decided to predict the trip demand, next to the trip duration and direction, as well.

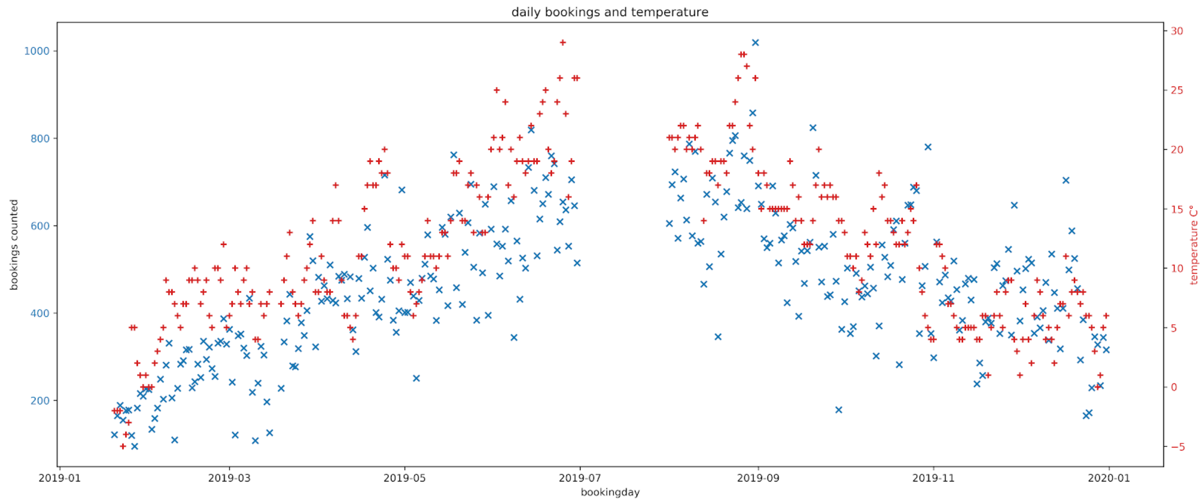


Figure 19: Average daily bookings (blue) and average daily temperature (red) for 2019

4.7 Preprocessor

In order to automate the findings from the exploration part about preprocessing and cleaning of the data we wrote the preprocessor. It starts by cleaning the data and filtering for the city of Bremen. Then trips are generated as described in Chapter 3. Further, trips again are cleaned by dropping duplicates, any trips longer than 1 day, exact round trips, round trips shorter than 8 minutes, round trips over three standard deviations from the mean and trips which duplicate over start and end time. Deeper insights about trip cleaning can also be found in Chapter 3. Finally, weather data for Bremen is fetched, cleaned and merged on the trip data based on the start time of each trip.

The final data collection of trips is in a universal format for further prediction. Features which are unique to each model are generated prior to prediction in the corresponding training method.

5 Predictive Modeling and In-Sample Evaluation

Based on lessons learned from previous tasks, we will predict the trip duration, the direction of the journey (depending on two locations of interest) as well as trip demand. The modelling procedure and evaluation are the same for each prediction target. To validate our developed models later on, we split our data set into training [40%] and test [60%] subsets initially, and further split the test subset [60%] into validation [24%] and test [36%] subsets. First, we calculated and assessed correlations of all possible features to our target variable (e.g. demand, duration, direction). Based on this initial assessment, we evaluated possible relevant features using a correlation matrix to determine the final features for each prediction model. Lastly, we evaluated and compared our models performance by calculating the r-squared (R^2) score and mean absolute error (MAE). The R^2 metric provides us comparability between our different models, while the MAE gives us an easy to interpret error value for respective prediction goals. In the case of the direction classification, our metric of interest differs from those of duration and demand prediction. Here, we will focus on the accuracy metric for direction predictions.

5.1 Predicting the Duration of Trips

The goal of the first prediction task was to predict the duration of each trip in minutes based on its starting conditions. To establish which features contain informational value we need to calculate the correlation of each feature to the target variable (duration_min) (see figure 20).

```
duration_min           1.000000
duration_sec           1.000000
start_plz              0.032302
start_place            0.016068
start_lat              0.014783
max_mean_m/s           0.011748
max_m/s                0.011627
mean_speed_h/s         0.011497
min_mean_m/s           0.010279
direction_degree_y     0.002629
direction_degree_x     0.002391
day_of_month           0.001005
start_lng             -0.002108
humidity_2m           -0.003819
min                   -0.004548
temp_2m               -0.008943
max_at_2m             -0.009031
weekdays              -0.012378
dew_point_2m          -0.014040
hour                  -0.018113
month                 -0.029515
dtype: float64
```

Figure 20: Correlation of features to target value duration of trips

We found that no feature has a higher correlation than start_plz ($r = .03$). This is supported by our findings in 4.2 and the noisy, non-normal distribution of trip lengths in each month. Thus, we could already see that trip duration in Bremen is not predictable and seems to be independent of external factors. Nonetheless, we tried to run a linear regression as well as Random Forest Regression to see if the features still provide some predictive power.

The features start_plz, start_place and max_mean_m/s (see Appendix C) were chosen due to having a correlation of $r \geq .01$ to the target variable and being fairly independent of each other ($r \leq .15$).

A search for the polynomial degree showed that the model performs "best" with a degree of 1. The results were still insignificant with a score of R^2 = **.001** for the train set and R^2 = **.001** for the validation set. The Random Forest did not improve results by any large significance either but we chose to integrate it into the package due to slightly better performance metrics in the validation set (R^2 = **.004**).

Our assumption, that the informational value to predict trip durations in Bremen based on the given features is non-existent, holds.

In future research data like the traffic light information could help to generate some informational value and maybe predict durations with some confidence.

5.2 Predicting the Trip Direction

Feature	**Mean Decrease Impurity (to_uni_bool)**	**Correlation (to_uni_bool)**
start_lng	0.1836	-.13
start_lat	0.172	-.2
humidity_2m	0.1489	-.01
dew_point_2m	0.1404	-.01
max_mean_m/s	0.1338	.01
hour	0.0964	.01

Table 4: Mean Decrease Impurity scores and Pearson correlations of features used for predicting to_uni_bool.

For the development of a model that predicts whether a trip will be towards the University of Bremen or not we had to create a target variable first. Therefore, we calculated the difference of the distances between each start and end location for each trip. If the result was positive, the trip was towards the university (to_uni_bool = 1, else to_uni_bool = 0). A Random Forest Classifier (criterion='entropy', class_weight='balanced_subsample') (RFC) proved to perform best on this task. We selected features by checking correlations and calculating Mean Decrease Impurity (MDI) scores. This resulted in choosing features with an MDI >0.1 shown in table 4.

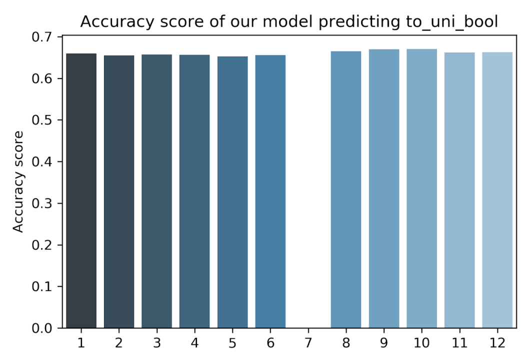


Figure 21: Bar chart plotting the accuracy scores of Random Forest Classifier predicting to_uni_bool for each month.

To make sure that our model generalizes well (thus avoiding overfitting) we optimized the model's hyperparameters by performing a grid search. This led to an optimum of 256 estimators with a maximum depth of seven. The evaluation of this model showed that it generalizes very well as the accuracy score of all three training, validation and test sets are about 67%. Figure 21 supports this by showing the results of a k-fold cross-validation (k = 12) where we choose each month of the year 2019 to be the test set once when training on all others. This also proves how well the model generalizes as the accuracy is equal for each month (no data for July).

As the main station of Bremen is the area with most trips generally, we trained another model equivalent to the previous that predicts if a trip will be towards the main station or not. The two models differ in terms of their most important features (see Appendix E) and that the model performs slightly better in terms of accuracy (68%). Additionally, we trained a Logistic Regression model for the classification which performed worse than the RFC so we stuck with our first approach. Two further models (Polynomial Regression) that predict the distance a trip moves in direction to the uni/main station (in kilometers) did not show good results as well (to_uni: $R^2 = .246$, $RMSE = 1.233$, to_main_station: $R^2 = .251$, $RMSE = 1.366$).

5.3 Predicting the Number of Bookings for Given Temporal Resolution

As the number of bookings indicate the demand for NextBike bikes, we applied a polynomial regression model to the data. The models were optimized regarding their hyperparameter (polynomial degree) for the corresponding temporal resolution of one, six, twelve and 24 hours. First, we assessed the feature selection by analyzing the correlation of possible features with our target variable "number_bookings" within a 24-hour temporal resolution. The target variable was created by simply adding up individual incidents for given temporal resolutions. Furthermore, we assessed the correlation between variables of temporal resolution as well as the correlation between weather features. We consider "month" ($r = .33$), "hour" ($r = .28$), "temp_2m" ($r = .72$) and "min" ($r = -.42$) as relevant features for predicting "number_bookings" (see Appendix G).

The following table 5 pictures the suggested polynomial degree for a given temporal resolution obtained by the training set (1), the validation set performance (R^2 and MAE) of said polynomial model (2), the performance metrics when fitting on the combined training and validation set (3), and finally the test set performance metrics (4). The mean number of bookings for given temporal resolution are provided as well for reference (5). Our number of bookings prediction models achieved relatively high performance on 12-hour and 6-hour temporal resolutions in the in-sample evaluation. We can see that the predictions with a temporal resolution of 12 hours achieved the highest R^2 score of $R^2 = .773$ with an MAE = 36.75. This is a rather surprising result, since the features were selected within a 24-hour temporal resolution. The worst performance on the test set occurred with a temporal resolution of one hour with an score of $R^2 = .597$ and an MAE = 7.09. To achieve better results in the 24-hour time span, the feature "hour" could be dropped, as it does not provide valuable information for when predicting daily demands. We decided to keep it, as we expected the feature to be relevant for smaller temporal resolutions. This assumption has been confirmed. To achieve better performance within an hourly resolution, the feature selection process could be repeated by analyzing the correlations of the specific hourly data set. This suggestion applies to every temporal resolution.

temporal resolution	**24 hours**	**12 hours**	**6 hours**	**1 hour**
(1) hyperparameter deg. (polynomial degree)	1	2	3	5
(2) validation R^2	.794	.807	.727	.602
(2) validation MAE	62.57	38.89	30.58	7.30
(3) merged R^2	.736	.807	.768	.610
(3) merged MAE	68.61	36.42	28.18	7.12
(4) test R^2	.649	.773	.759	.597
(4) test MAE	71.18	36.75	28.61	7.09
(5) mean number bookings (per temporal resolution)	460.46	230.28	115.41	19.76

Table 5: Performance metrics of modelling and in-sample evaluation process

6 Package description

To facilitate the easy and reproducible application of transforming and applying various machine-learning models to raw NextBike data, we created a Python package that can easily be installed by the end-user. In addition to the package itself, the repository[4] contains various files and folders that are useful when working with the NextBike package. Thus, when speaking of "the repository" from here on, not only the NextBike package itself, but everything that comes shipped with it, is meant.

6.1 Repository Structure

The repository contains a README file that informs the user on functionalities and installation procedure, a setup.py file that enables the user to install the NextBike package using the same, easy method as installing packages from the Python Package Index (pip), and an environment.yml file that enables the user to work in an isolated conda environment that comes pre-shipped with all necessary dependencies. Next to these three files, there is the NextBike package, a data folder, a models folder, and a notebooks folder.

The data and models folders are also used by the NextBike package, with data having four sub-folders raw, external, processed, and predicted. Raw NextBike dumps belong into the raw folder,

[4] https://github.com/janiksudo/PDS_Project/

and without the user putting files there, the NextBike package is not able to work. External data like weather data, administrative boundaries of cities where NextBike operates and GeoJSON files that describe the service zones of NextBike cities, that employ a free-floating model, need to be put into the external folder. The NextBike repository comes pre-shipped with external data for the city of Bremen. When using the transformation and cleaning functionalities of the package, intermediate data dumps are stored in the processed folder. When applying machine-learning models on NextBike data, predictions are saved for further evaluation into the predicted folder. Pre-trained machine-learning models are stored in the models directory.
The notebooks directory contains some interactive Jupyter Notebooks that were used for data exploration of the city of Bremen and show how to create meaningful visualizations. Additionally, we did the hyperparameter search and tuning for the machine-learning models here. These notebooks can be used by the user to retrace the thought process involved with creating the final NextBike package. Moreover, the user can create interactive notebooks in this directory as well and easily import the top level NextBike package for development and further evaluation of predicted data.

6.2 Package structure and functionalities

The NextBike package itself is structured fairly simple, with only three submodules: io, model and preprocessing, as well as the entry point for the user-friendly command line interface: cli.py.

Everything that reads and writes files is found in the io module, transformation and cleaning related is found in the preprocessing module, and everything that enables the user to apply machine-learning is found in the model module. Every module implements a class (except for the io module, which provides its functionality directly through importable functions) that can be instantiated from anywhere within the NextBike package and then used to call its public methods. In addition to that, classes might have methods (denoted by a name with a leading underscore) that are not exposed to other classes, but can be called directly if necessary for development purposes[5].

[5] This follows PEP-8 standards: https://www.python.org/dev/peps/pep-0008/#method-names-and-instance-variables

Note: A more detailed and tech-focused look into the functionalities and the command line interface is available in the README of the repository.

7 Out-of-Sample Model Evaluation

To evaluate our models out-of-sample performance, we were provided with a test data set consisting of previously missing data from the month of July. The following subsections provide further detail of each model's evaluation.

7.1 Duration Prediction Model

The duration prediction was, matching our expectations, not possible or accurate on the new data. With an $R^2 = .007$ and a mean absolute error of 47 minutes per trip the model is not fit for deployment and should not be used in practice. To improve the models performance, additional features which contain informational and predictive power are necessary.

We assume that the non-normal distribution of the outcome variable and the lack of informational features contribute to this non-existing predictive power (see chapter 4.2), too. If NextBike could collect additional features (as mentioned in chapter 7.1) in the future, we could repeat this analysis with features of higher correlation and possibly different prediction models.

7.2 Journey Prediction Model

The evaluation of our models predicting the direction of a trip (to university or to the main station) showed that the models generalize well. Accuracy, precision and recall scores around 67% match the scores of prior performance tests on validation and test data (in-sample) with a difference of less than 3%. Regarding the magnitude of the accuracy the main limitation was not finding enough features that correlate sufficiently with the target variable. In order to take the next steps and improve the accuracy of the models, we propose to continue working on feature engineering. Classification reports containing the precis results of all related performance metrics can be found in the Appendix E and Appendix F.

7.3 Number of Bookings Prediction

The models which predict demand on various hourly temporal resolutions underperformed expectations (to various degrees depending on the resolution). The mean absolute error is only slightly higher in the out-of-sample test while the R^2-score gets worse with higher time frames. This means that the prediction of values is generalizable but the variance is not explained very well in higher temporal resolutions.

temporal resolution	24 hours	12 hours	6 hours	1 hour
IS test R^2	.649	.773	.759	.597
IS test MAE	71.18	36.75	28.61	7.09
OOS test R^2	.063	.642	.672	.513
OOS test MAE	77.41	41.89	30.35	7.82
mean number bookings	460.46	230.28	115.41	19.76

Table 6: Performance comparison OOS/IS

The out-of-sample predictions of the 24-hour model (R^2=.063) performed significantly worse than during in-sample predictions (R^2=.649), as seen in table 6. One reason could be that our 24-hour model was trained on a lower number of incidents in comparison to our other models. Another performance limitation of the 24-hour model stems from the feature "hour" which had one of the lowest correlation to "number_bookings" (r =.28), as it does not provide valuable information in a daily temporal resolution. We kept that feature because we expected it to be relevant for smaller temporal resolutions.

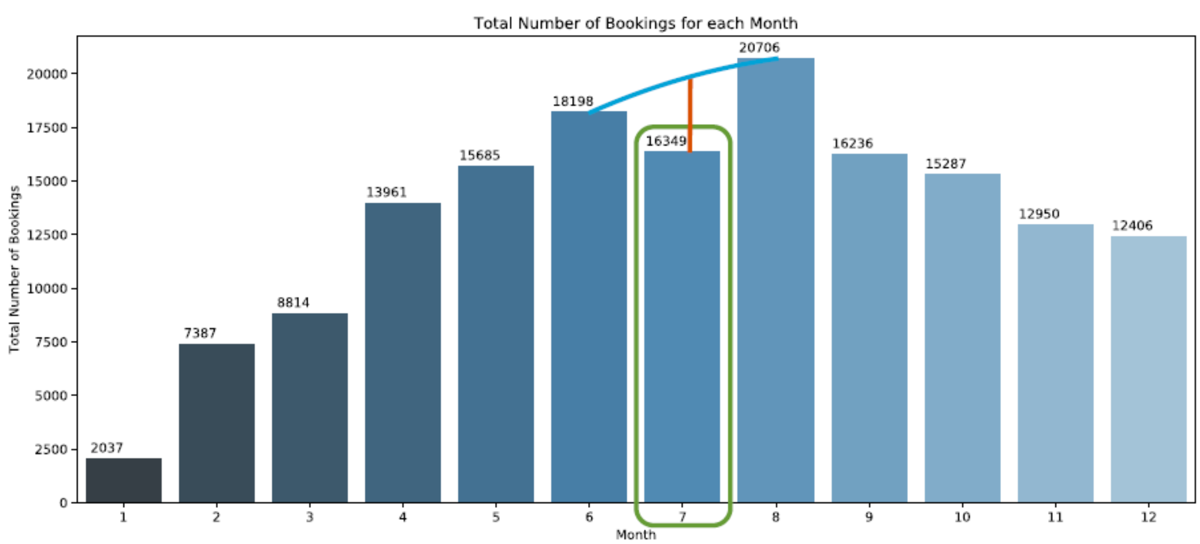


Figure 22: Annotated total number of bookings per month for 2019 for evaluation reference

A possible reason for the models' underperformance, in general, could be the out of the ordinary (lower) number of bookings in July as outlined with the red line in figure 21. A possible explanation for this phenomenon could be the beginning of the summer break starting from 4th of July, which is a highly preferred vacation period for Germans. For this reason, there may have been a slump in demand for bikes. The blue line could represent our models expectations for the month of July, while the red line symbolizes resulting prediction error. An evidence for this theory is that our models overall have a positive bias and overpredict the number of bookings (see Appendix H).

8 Business Recommendation

We assume NextBike has four main business areas of interest: Gaining new customers (1), retaining existing customers (2), as well as optimizing the bike availability across spatial resolutions (3), in order to provide the optimal number of bikes for the predicted amount of bookings and, thus, save costs. Lastly, we assume the goal of profit maximization (4).

To gain and retain customers (1,2), we would suggest to collect supplementary data for every reservation, specifically user-related data. This could comprise a status of the customer (e.g. work commuter, student, WK card owner), the used tariff (Basic, Comfort, Day Pass), or for instance the driven distance in kilometers provided by the location data. This information could be clustered and analyzed, to design an optimized and attractive pricing system for individual customer segments and, thus, attract more customers and maximize the profit.

Based on our analysis it was possible to precisely compute the number of bikes for each postal zone. Correctly allocated, the number of required bikes can be reduced (3), even if the total number of bookings could increase in the near future. To achieve optimal bike allocation, a prediction model for bike demand within certain spatial resolutions could be developed. Unused bikes could be located at rental stations with a higher predicted demand, which results in cost reductions and efficiency gains.

Furthermore, our analysis concluded a remarkable low bike utilization during winter months (Figure 5), and on Sundays in general (refer to chapter 4.2 Figure 6). NextBike could offer special discounts for winter periods and Sundays. In order to maximize profits and demand (4), adapting the pricing mechanisms with reference to the day of the week can be considered as well.

9 Conclusion

As data collection and business analytics are evolving to some of the most important business activities, it is just as important to expand the scope of data as to accurately analyze data. An appropriate data analysis provides valuable insights. From a business perspective, it allows NextBike to increase revenue streams by retaining existing and gaining new customers through an individualized and optimized pricing system. Simultaneously it enables NextBike to decrease its operating costs and maximize its profits by accurately predicting bike demand, bike availability and customer journey direction. Modern visualization techniques like discretizing maps with hexagons or interactive heatmaps enabled us to identify patterns of great importance for all rental service stakeholders. Moreover, out of an environmental perspective, an increasing customer base and attractive bike rental prices automatically lead to a decrease in use of environmentally burdening mobility services.

Appendix

.

Appendix A. Distribution of bookings of first and last

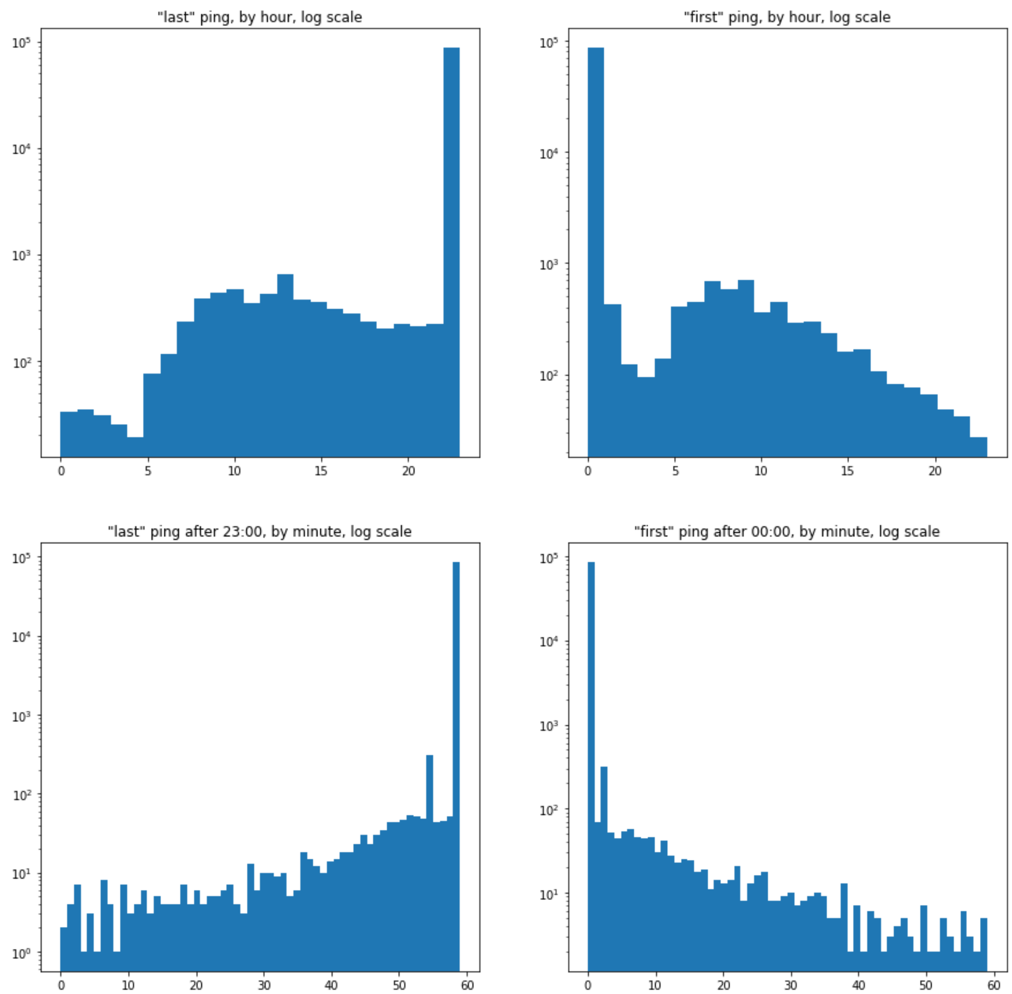


Figure 23 Distribution of bookings of first and last

Most bikes have their last "ping" at exactly 23:59 and their first "ping" at exactly 00:00. We can assume, these are just status pings that bikes/stations emit when a new day starts.

These can be safely omitted when creating trips. For all other occurrences, we can assume these are trips that started on one day and ended on the next.

Appendix B. Number of round trips per day

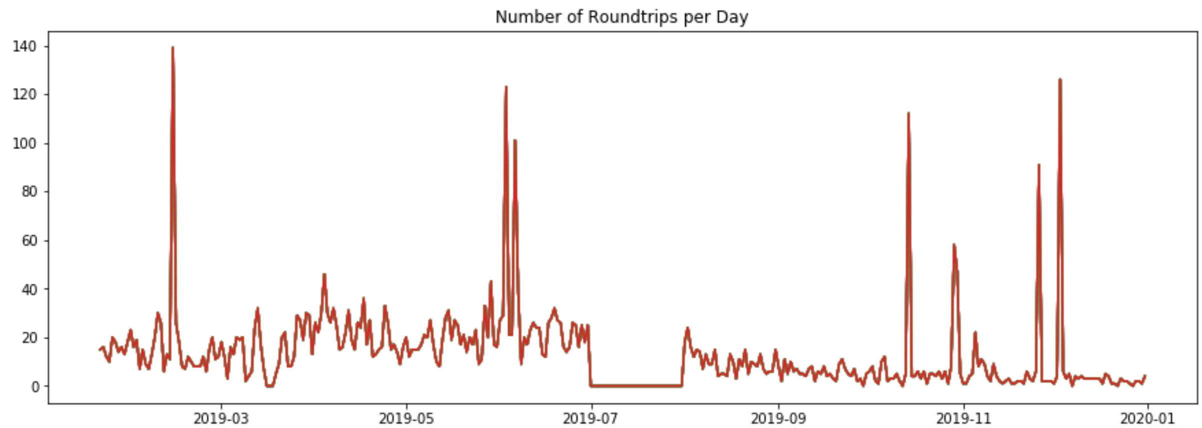


Figure 24: Number of round trips per day

Appendix C. Weather features table

Attribute name	**Attribute description**
temp_2m	air temperature measured at 2 meters height in Celsius
humidity_2m	relative humidity measured at 2 meters height in percentage
dew_point_2m	dew point temperature measured at 2 meters height in Celsius
max_at_2m	maximum air temperature measured at 2 meters height of last 10 minutes
mean_speed_h/s	mean of wind speed during the last 10 minutes in m/s
direction_degree_x	mean of wind direction during the last 10 minutes in degree
max_m/s	maximum wind gust of the last 10 minutes in m/s
min_mean_m/s	minimum 10-minute mean wind velocity in m/s
max_mean_m/s	maximum 10-minute mean wind velocity in m/s
direction_degree_y	direction of highest wind gust in degree
min	duration of precipitation within the last 10 minutes in minutes
mm	precipitation height of the last 10 minutes in mm

Table 7: Attributes from the weather data set

Appendix D. Scatter plot between rented minutes and temperature

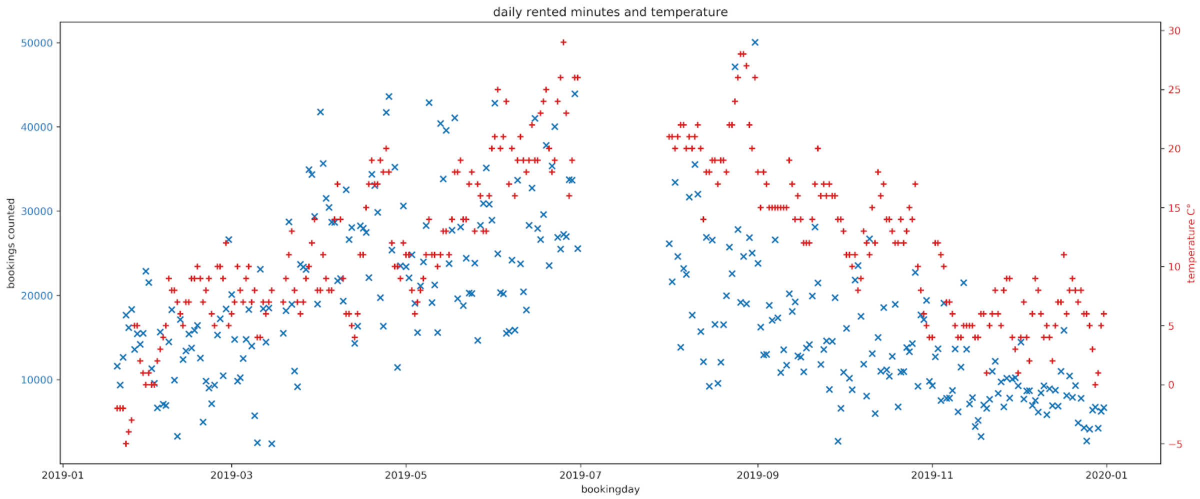


Figure 25: Scatter plot of daily total rented minutes (blue) and daily average temperature (red)

Appendix E. Direction classification In-Sample

	precision	recall	f1-score	support
0	0.65	0.64	0.65	23714
1	0.67	0.68	0.67	25046
accuracy			0.66	48760
macro avg	0.66	0.66	0.66	48760
weighted avg	0.66	0.66	0.66	48760

Table 8: Classification report of test set (in-sample) performance of to_uni_bool Random Forest Classifier

	precision	recall	f1-score	support
0	0.56	0.45	0.50	15917
1	0.56	0.66	0.61	16589
accuracy			0.56	32506
macro avg	0.56	0.56	0.55	32506
weighted avg	0.56	0.56	0.56	32506

Table 9: Classification report of test set (in-sample) performance of to_uni_bool Logistic Regression

	precision	recall	f1-score	support
0	0.69	0.65	0.67	24287
1	0.67	0.71	0.69	24473
accuracy			0.68	48760
macro avg	0.68	0.68	0.68	48760
weighted avg	0.68	0.68	0.68	48760

Table 10: Classification report of test set (in-sample) performance of to_main_station_bool Random Forest Classifier

Appendix F. Direction Classification Out-of-Sample

	precision	recall	f1-score	support
0	0.64	0.62	0.63	7915
1	0.65	0.68	0.67	8434
accuracy			0.65	16349
macro avg	0.65	0.65	0.65	16349
weighted avg	0.65	0.65	0.65	16349

Table 11: Classification report of test set (out-of-sample) performance of to_uni_bool Random Forest Classifier

	precision	recall	f1-score	support
0	0.70	0.65	0.68	8110
1	0.68	0.73	0.70	8239
accuracy			0.69	16349
macro avg	0.69	0.69	0.69	16349
weighted avg	0.69	0.69	0.69	16349

Table 12: Classification report of test set (out-of-sample) performance of to_main_station_bool Random Forest Classifier

Appendix G. Correlation matrix for predictive analysis of demand

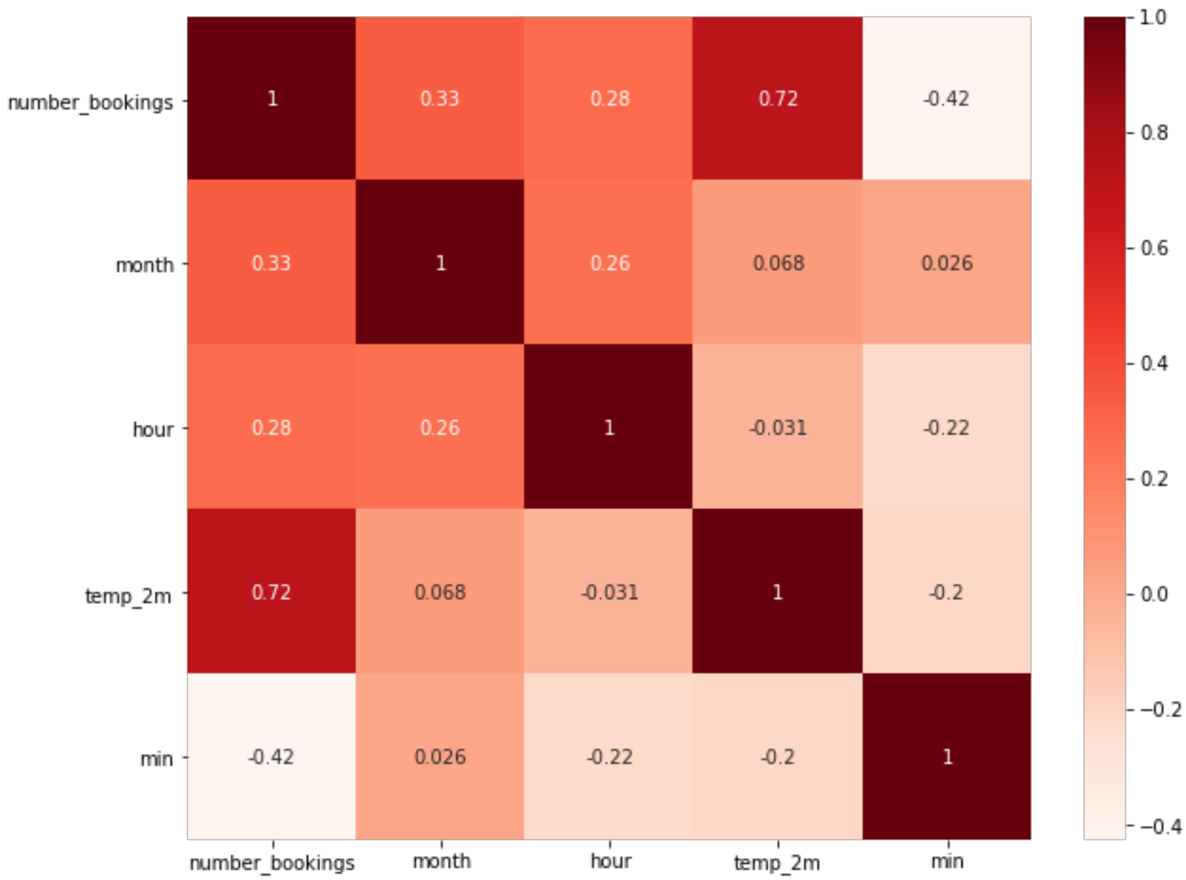


Figure 26: Correlation matrix of number of bookings prediction model features

Appendix H. Plotting the actual demand vs. predicted demand for 6h temporal resolution

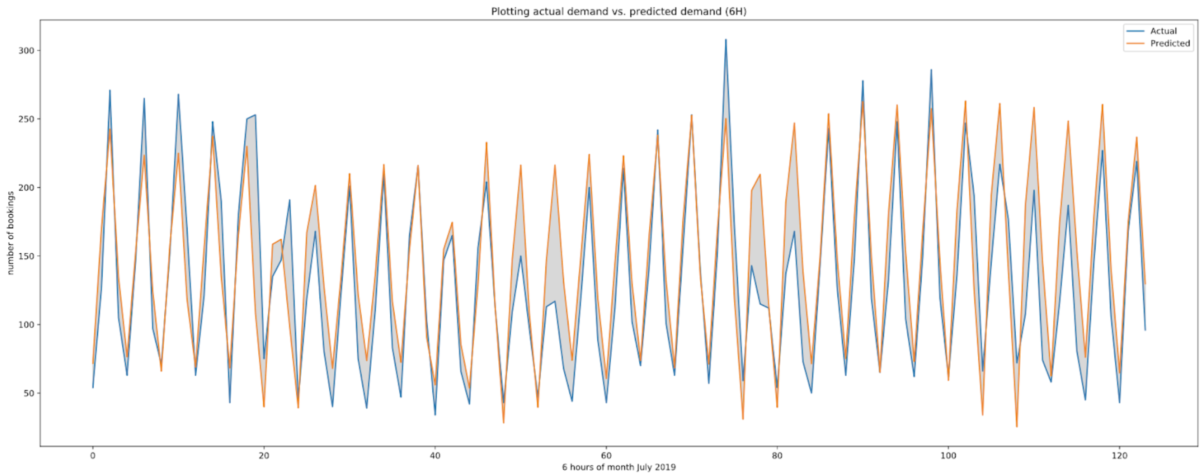


Figure 27: Line plot of actual (blue) and predicted (orange) number of bookings of test set (out-of-sample).

Eidesstattliche Versicherung

Hiermit versichere ich an Eides statt, dass ich die vorliegende Arbeit selbstständig und ohne die Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten und nicht veröffentlichten Schriften entnommen wurden, sind als solche kenntlich gemacht. Die Arbeit ist in gleicher oder ähnlicher Form oder auszugsweise im Rahmen einer anderen Prüfung noch nicht vorgelegt worden. Ich versichere, dass die eingereichte elektronische Fassung der eingereichten Druckfassung vollständig entspricht.

Die Strafbarkeit einer falschen eidesstattlichen Versicherung ist mir bekannt, namentlich die Strafandrohung gemäß § 156 StGB bis zu drei Jahren Freiheitsstrafe oder Geldstrafe bei vorsätzlicher Begehung der Tat bzw. gemäß § 161 Abs. 1 StGB bis zu einem Jahr Freiheitsstrafe oder Geldstrafe bei fahrlässiger Begehung.

Köln, den 10.06.2020

J. Muires

Janik Muires

T.O.

Tim Ockenga

Tom Celig

Maximilian Kaiser

D.G.

Diana Gowalow