Technical Blog

The weather dataset was obtained from Kaggle (https://www.kaggle.com/selfishgene/historical-hourly-weather-data/data), which originally acquired from OpenWeatherMap website. Data on 36 cities and their weather attributes for 2012 to 2017, were contained in 7 csv files:

- city_attributes
- humidity
- pressure
- temperature
- weather_description
- wind_direction

Each weather attribute csv file contain the datetime and the value of attribute for each city.

Example (Humidity.csv):

datetime	Vancouver	Portland	San Francisco	Seattle	Los Angeles	San Diego	Las Vegas	Phoenix	Albuquerqu
16:00:00									
2012-10-02 17:00:00	76.0	75.0	94.0	58.0	19.0	15.0	18.0		
2012-10-02 18:00:00	67.0	52.0	70.0	63.0	24.0	57.0	9.0	9.0	23.0
2012-10-02 19:00:00	58.0	60.0	83.0	48.0	15.0	57.0	9.0		
2012-10-02 20:00:00	53.0	56.0	75.0	43.0	12.0	35.0	9.0	13.0	22.0
2012-10-02 21:00:00	47.0	34.0	75.0	43.0	14.0	69.0	8.0		21.0

There was several rows that contained nan values for some of the cities. It would be unwise to remove the rows of the weather attributes because the data was still relevant to the city. Since all the data related to a particular city were scattered among the 7 files, the data was merged based on the city, creating a file for each of the cities (36 files).

datetime	humidity	pressure	temp_celsius	description	wind_speed	wind_direction
01/10/2012 13:00	93	1001	12.68	overcast clouds	4	230
01/10/2012 14:00	91	986	12.68464995	sky is clear	4	230
01/10/2012 15:00	87	945	12.69778954	sky is clear	4	231
01/10/2012 16:00	84	904	12.71092912	sky is clear	4	233
01/10/2012 17:00	80	863	12.72406871	sky is clear	3	234
01/10/2012 18:00	76	822	12.73720829	sky is clear	3	236
01/10/2012 19:00	72	822	12.75034788	sky is clear	3	237
01/10/2012 20:00	68	822	12.76348747	sky is clear	3	238
01/10/2012 21:00	64	822	12.77662705	sky is clear	3	240
01/10/2012 22:00	61	822	12.78976664	sky is clear	3	241
01/10/2012 23:00	57	822	12.80290622	sky is clear	2	243
02/10/2012 0:00	53	822	12.81604581	sky is clear	2	244
02/10/2012 1:00	49	822	12.82918539	sky is clear	2	245

The merge was based on an outer join with the datetime column. Outer join was necessary to capture all weather attributes, not just those that had a common datetime.

For each file, it was easier to remove nan's. Nan rows were removed, but decided to backfill values since values do not change immensely from hour to hour. Since some of the columns had many missing values, backfill was limited to 8 rows and forward fill was performed and also limited to 8 rows. This minimized the number of nan values in the dataset.

Exploring the data, temperature was found to be in Kelvin. Since we Canadians are more familiar with temperature in Celsius, the temperature column was changed to Celsius degrees (formula: C=K-273.15)

At this time of the project, it was not determined how the weather was going to be utilized. It may have been useful to determine which part of the day, activities would occur within the 24 hour cycle. A procedure is included to identify the time of day (every 6 hours: AM1, AM2, PM1, PM2). Another procedure is included to identify the cardinal direction of the wind, instead of wind degree. Unfortunately, these procedure were not used for the purpose of the project. Each of the city file were saved for next stage of processing

Looking at predicting the number of bike rides based on Montreal weather, Bixi dataset set was found on Kaggle (https://www.kaggle.com/aubertsigouin/biximtl/data). Data comprised of 2014 to 2017 bixi rides and its associated station locations.

Bixi data was very clean, without any or very little nan values. Since I prefer to see the duration in minutes instead of seconds, I converted the value by dividing by 60. Montreal weather data was merged with bixi ride 2017 data based on the datetime, and saved as a processed csv file called bixiWeather.csv. Due to time limitation, I did not include 2014, 2015 and 2016 bixi rides for analysis.

Business Blog

Weather conditions plays an important role and has a large impact on different areas of our human lives. It can affect our economy, society, agriculture, transportation system, and even individual human aspects. Although there are many interesting studies to look how weather correlates to a specific subject matter, I will be exploring the use of bikes from a bike sharing system and predicting the number of bikes rides for a given day based on its weather condition.

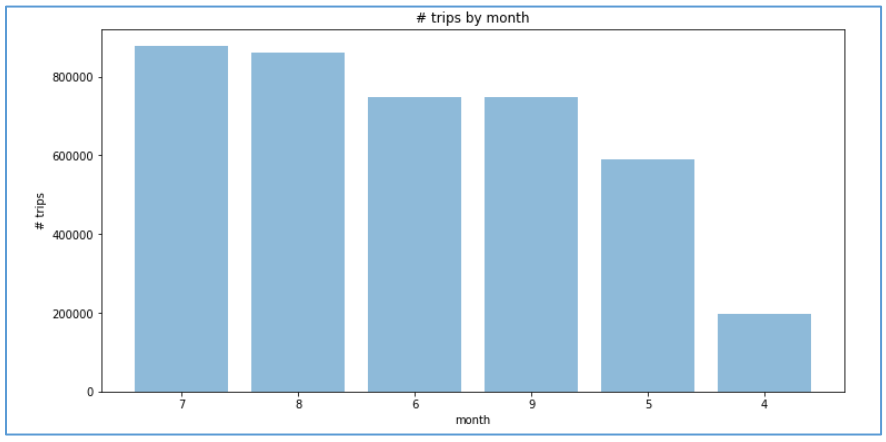
The focus will be on bixi, a bike sharing system, located in the city of Montreal, Quebec. Bikes are located at a specific location or station throughout Montreal. The rider buys a one-day pass, one-way trip, or a 10- one-way package with a credit card. The rider reads and accepts the user contract before he/she can undock the bike from the docking station. If the rider plans to use bixi bikes more frequently in the future, he/she can become a member to take advantage of cost savings. The bikes can return to any of the 540 bixi stations. If the destination station is full, another 15 minutes can be request, free of charge, to allow the rider sufficient time to return the bike at the next closest available station. The bike sharing system is in service from mid-April to mid-November each year. All stations are removed when bixi is not in service.

From a business perspective, data analysis can help provide insight to bike ridership in the city and bring added value to the bike sharing industry

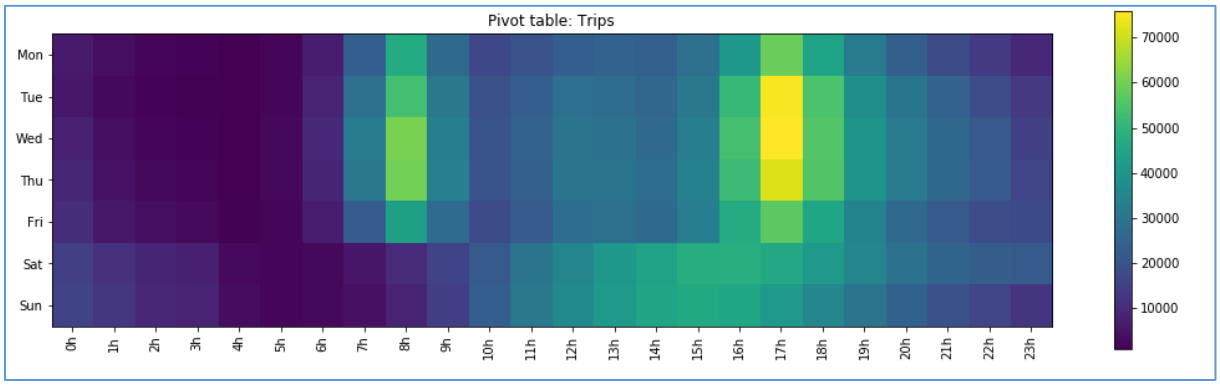
- Ridership and station statistics (current number of members and riders, frequency of each station, average duration of a ride, number of rides per month, time of use, frequency by members vs non-members, etc)
- The weather conditions during the riders' bixi rides
- Given the weather conditions and the time of day, can we predict the number riders?
- Ridership demographics can help increase revenue. If riders are tourists, can more bike stations can be added to heavy tourist areas?
- City infrastructure to support a bike sharing program and reduce traffic congestion during biking season

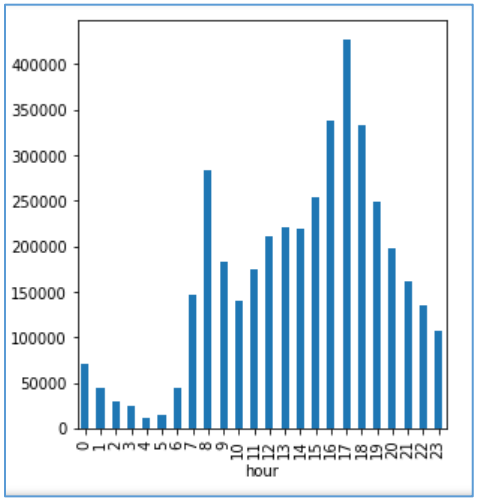
As the results show,

- duration of the majority rides occur in less than 15 minutes
- temperature of majority rides are between 15 and 25 Celsius
- Members ride more often, especially from June to September
- Non-members ride most often during the month of July
- Rides peak during July and August;

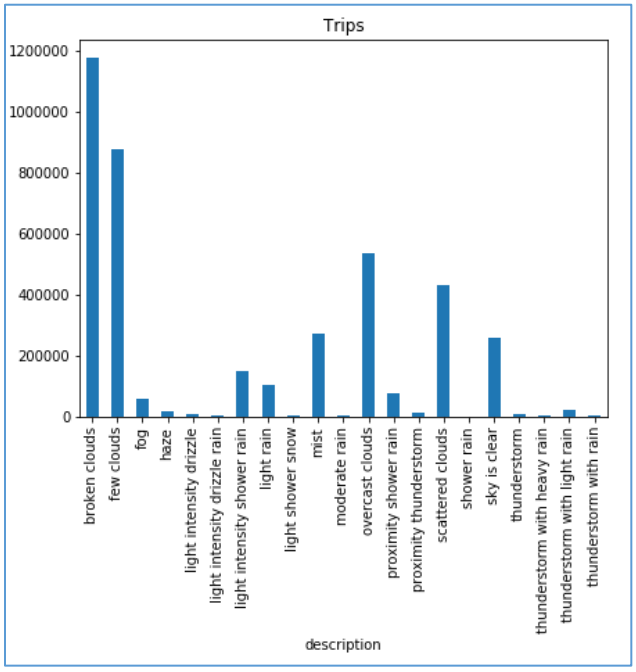


- Rides peak at 8am and between 4pm to 6pm Monday to Friday, which may indicate that many use bixi bikes to go and from work





- Riders do not appear to ride bixi bikes often on rainy or snowy days



With the 6 regression models used, the Random Forest Regressor models appear to have higher predictive accuracy than the other models. They show lower mean absolute errors (mae) and root mean square errors (rmse)

	model	mae	rmse
3	RandomForestRegressor100	154.183227	260.638500
1	RandomForestRegressor	160.633867	271.663691
2	RandomForestRegressor10	166.716872	277.785557
5	DecisionTreeRegressor	214.565271	387.548419
4	KNeighborsRegressor	330.619212	488.276417
0	LinearRegression	521.976636	673.659273

Technical Challenge Blog

Being a Windows user who is not proficient in python, technical challenges were encountered during the course of this project. I would love to learn use the linux environment given time and resource to do. With this project, I used juptyer notebook (anaconda) in Windows from my own desktop. In the classroom, I used linux in a virtual machine.

When trying to regression and clustering analytics, I had issues with installing python packages. I was not able to install pyproj package for the clustering analytics. I did, however manage to install findspark, but was not able to initiate it. So I decided to see if I can replicate the installation on the virtual box, unfortunately, I had problems there too. After a few hours, I decided to cease the investigation since it was taking a bit more time than I wanted to spend on it. I abandoned regression testing using Spark.

Going forward, I decided to use jupyter notebook in Windows so as to not hinder progress of the project.

Another technical challenge encountered was the lack of Python knowledge. As with any new language, learning and understanding python comes with practice. I had difficulties determining the proper technique and formula to group and display data. Although, I had some guidance by looking at the professor's code, I needed to formulate some of the techniques and answers based on my dataset. Thank goodness for google, I managed to get a few answers to understanding the python code.

For clustering, I used k-means model. Upon running it, the results took a long time to generate. Being unfamiliar with it, I was not sure if the model was working correctly and how to best reduce the time to run it to analyse the results. Changing the k value did not appear to have any impact to processing time.

For the prediction application, I had problems logging in Heroku and starting the sample that was provided. Once I had it working, I modified by HTML codes and ran the application:

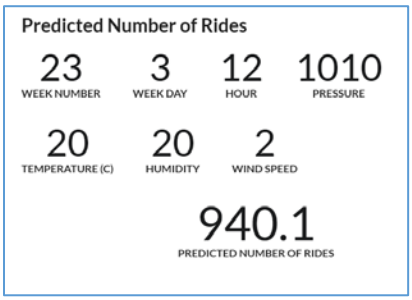
https://calm-journey-14126.herokuapp.com/predict_ride

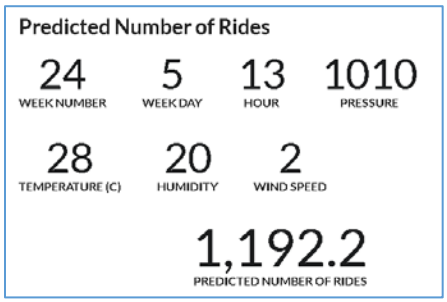
It didn't quite work out the way I expected with my prediction:

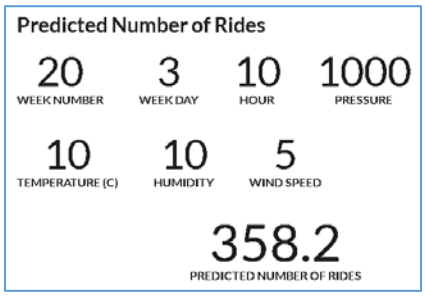
Bad Request

The browser (or proxy) sent a request that this server could not understand.

I was so ready to pack it in and hand it in as is. As I was reviewing what needed to push to github, I looked at my code once more, and made some corrections. It worked!







But I still have a lot to learn from this project.