Assignment 1

A. Exploratory Data Analysis:

The Seoul Bike dataset contains information regarding number of bikes rented on an hourly basis for a year and includes prevailing weather conditions. It is important to analyze every variable to understand the dataset better. EDA will provide insights on the distributions of variables, outliers points and, correlation between variables.

Summary of Numerical Variables:

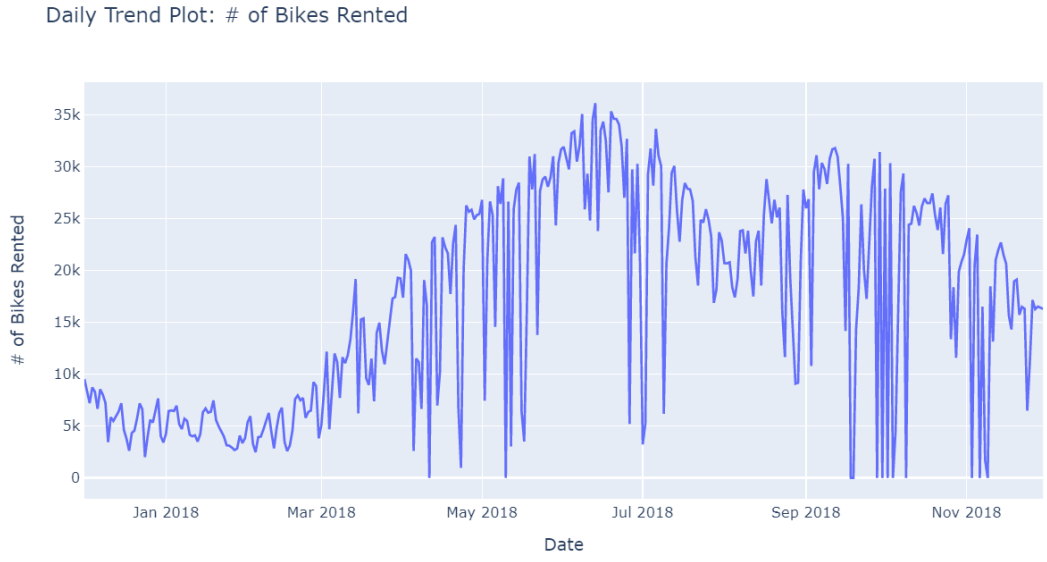
	count	mean	std	min	25%	50%	75%	max
bike_count	8760.0	704.602055	644.997468	0.0	191.0	504.50	1065.25	3556.00
temperature	8760.0	12.882922	11.944825	-17.8	3.5	13.70	22.50	39.40
humidity	8760.0	58.226256	20.362413	0.0	42.0	57.00	74.00	98.00
wind_speed	8760.0	1.724909	1.036300	0.0	0.9	1.50	2.30	7.40
visibility	8760.0	1436.825799	608.298712	27.0	940.0	1698.00	2000.00	2000.00
dew_point_temp	8760.0	4.073813	13.060369	-30.6	-4.7	5.10	14.80	27.20
solar_radiation	8760.0	0.569111	0.868746	0.0	0.0	0.01	0.93	3.52
rainfall	8760.0	0.148687	1.128193	0.0	0.0	0.00	0.00	35.00
snowfall	8760.0	0.075068	0.436746	0.0	0.0	0.00	0.00	8.80

Summary of Categorical Variables:

	count	unique	top	freq
hour	8760	24	23	365
season	8760	4	Summer	2208
holiday	8760	2	No Holiday	8328
functioning_day	8760	2	Yes	8465

Implication: The 'hour' variable is considered as a categorical variable and its explanation is given in the next section.

A1. Date and Time Variables:

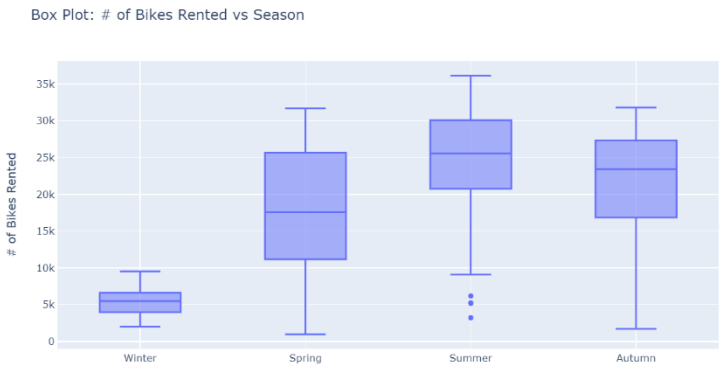


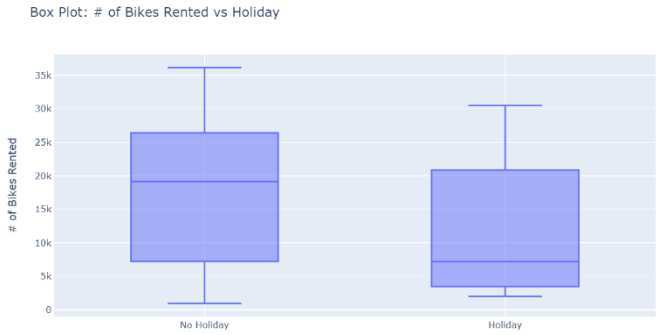
The above trend-plot shows the number of bikes rented on a particular day. As shown in the plot, there are certain days when no bike is rented. By looking at functioning day variable, one can see that on non-functioning days, bike count is 0. The list of days on the right correspond to the non-functioning dates.

```
131   2018-04-11
160   2018-05-10
291   2018-09-18
292   2018-09-19
301   2018-09-28
303   2018-09-30
305   2018-10-02
307   2018-10-04
309   2018-10-06
312   2018-10-09
337   2018-11-03
340   2018-11-06
343   2018-11-09
```

Implication: We do not need to include 'functioning_day' while training our models. On those days we do not to predict the number of bikes rented.

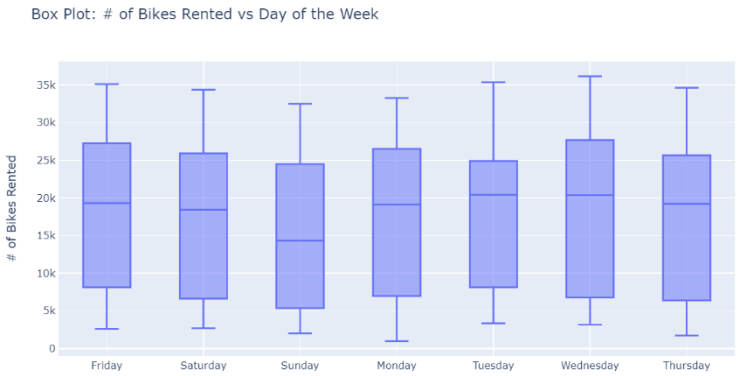
Another observation from the trend plot is that colder weather conditions (entire Winter, early Spring, and late Autumn) hampers bike rentals. This fact is also corroborated from the below set of boxplots between season and bike count.

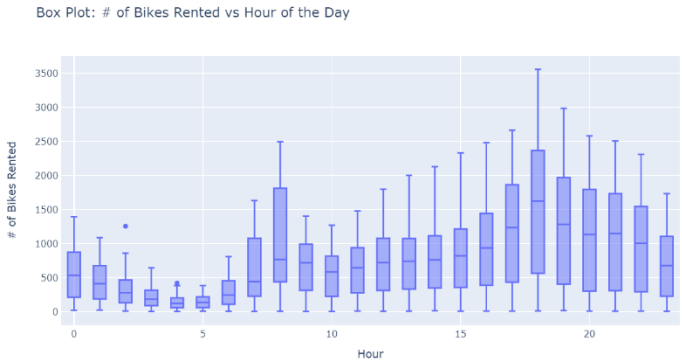




There were total of 18 holidays in a yearly calendar in Seoul (may or may not include weekend days). It is interesting to see that there are lesser bikes rented on holidays compared to working days as shown in the below set of box-plots. It shows that people generally rent bikes to commute and most of them prefer to stay at home during holidays. However, this correlation does not imply causal relationship because pupils who rent bike are not a sample of entire population.

Since holidays may or may not fall on weekends, analysis on number of bikes rented with respect to the days of a week can provide crucial insights. The median values are lower on the weekends compared to the weekdays, but one needs to test the significance of the expected value which is beyond the scope of this report.



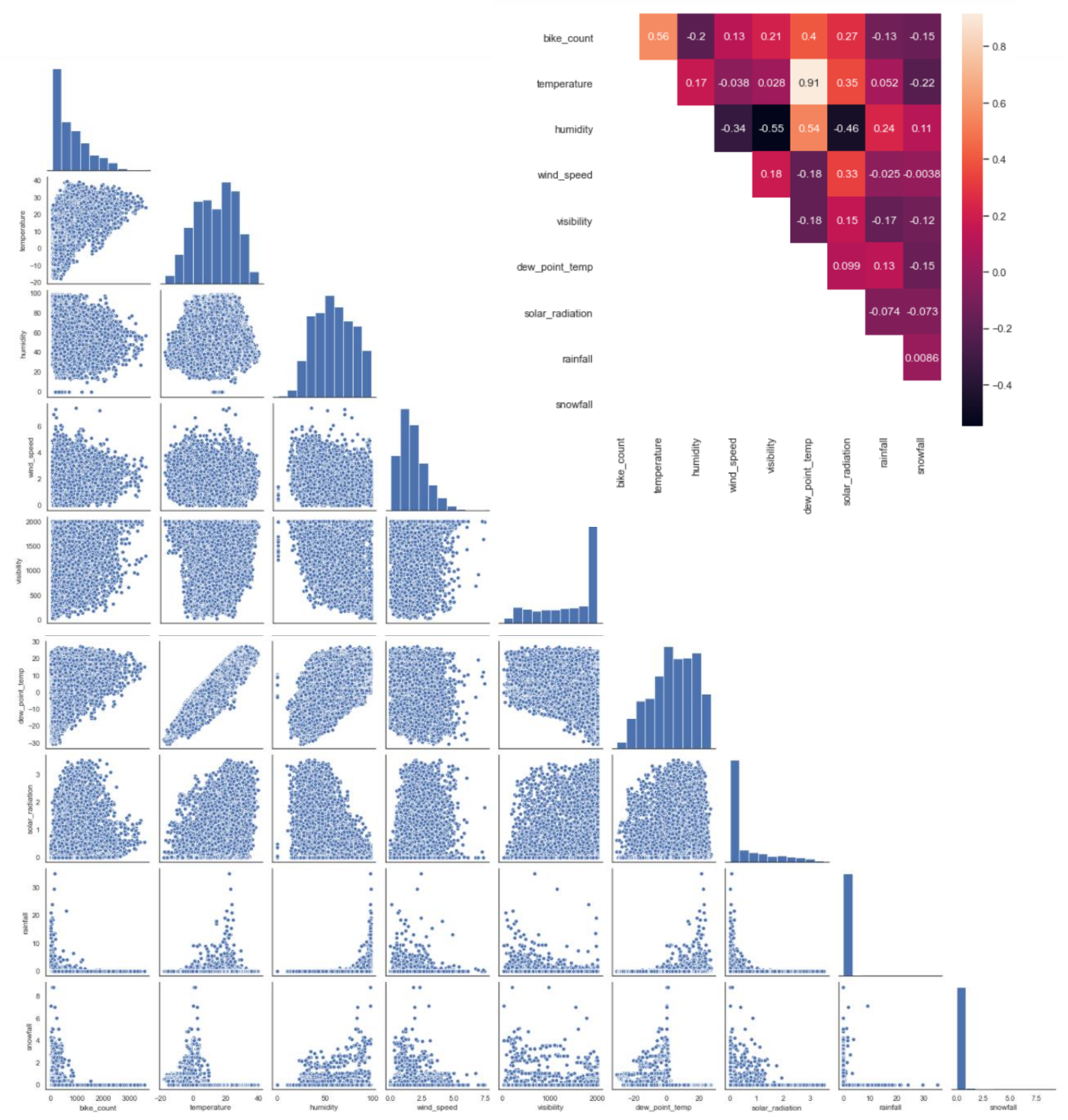


The dataset contains hourly information of bike counts. Since bikes are speculated to be rented for daily commutes (especially around office hours), hourly distribution of bike count can shed some light on the effect of peak commute hours on number of bikes rented. From the set of box-plots below, the bike sharing is predominant from 5 p.m. to 10 p.m.

Implication: The dataset does not contain day of week as variable. Hence a new variable is created to include in machine learning models.

A2. Weather Conditions:

Correlation heatmap between weather characteristics and bike count are shown below.



Implication: Dew point temperature is highly correlated with temperature (0.91) and including both these variables might lead to multi-collinearity issues while modelling. In addition, the variance inflation factor of all variables is less than 10 after removing dew-point temperature. Previously, by including that variable, two variables namely – temperature (87) and humidity (20) have VIF greater than 10 in addition to dew point temperature (116). Hence, dew point temperature is not included in the model building.

B. Data Preprocessing:

B1. Final Dataset:

The final dataset that is used for building linear and logistic models contains 12 variables (included: [‘day’], excluded: [‘date’, ‘functioning_day’, ‘dew_point_temp’]) with 8465 records (which excludes the non-functioning days).

	bike_count	hour	temperature	humidity	wind_speed	visibility	solar_radiation	rainfall	snowfall	season	holiday	day
0	254	0	-5.2	37	2.2	2000	0.0	0.0	0.0	Winter	No Holiday	Friday
1	204	1	-5.5	38	0.8	2000	0.0	0.0	0.0	Winter	No Holiday	Friday
2	173	2	-6.0	39	1.0	2000	0.0	0.0	0.0	Winter	No Holiday	Friday
3	107	3	-6.2	40	0.9	2000	0.0	0.0	0.0	Winter	No Holiday	Friday
4	78	4	-6.0	36	2.3	2000	0.0	0.0	0.0	Winter	No Holiday	Friday
...	...	...	...	...	...	...	...	...	...	...	...	...
8460	1003	19	4.2	34	2.6	1894	0.0	0.0	0.0	Autumn	No Holiday	Friday
8461	764	20	3.4	37	2.3	2000	0.0	0.0	0.0	Autumn	No Holiday	Friday
8462	694	21	2.6	39	0.3	1968	0.0	0.0	0.0	Autumn	No Holiday	Friday
8463	712	22	2.1	41	1.0	1859	0.0	0.0	0.0	Autumn	No Holiday	Friday
8464	584	23	1.9	43	1.3	1909	0.0	0.0	0.0	Autumn	No Holiday	Friday

The numeric features are normalized, and the categorical variables are one-hot encoded as shown below.

	intercept	temperature	humidity	wind_speed	visibility	solar_radiation	rainfall	snowfall	h_1	h_2	...	d_Monday	d_Saturday	d_Sunday
0	1.0	-1.484675	-1.032334	0.458402	0.929522	-0.654041	-0.132487	-0.17494	0	0	...	0	0	0
1	1.0	-1.509459	-0.983517	-0.895195	0.929522	-0.654041	-0.132487	-0.17494	1	0	...	0	0	0
2	1.0	-1.550766	-0.934701	-0.701824	0.929522	-0.654041	-0.132487	-0.17494	0	1	...	0	0	0
3	1.0	-1.567289	-0.885884	-0.798509	0.929522	-0.654041	-0.132487	-0.17494	0	0	...	0	0	0
4	1.0	-1.550766	-1.081151	0.555088	0.929522	-0.654041	-0.132487	-0.17494	0	0	...	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
8460	1.0	-0.708096	-1.178784	0.845144	0.755481	-0.654041	-0.132487	-0.17494	0	0	...	0	0	0
8461	1.0	-0.774188	-1.032334	0.555088	0.929522	-0.654041	-0.132487	-0.17494	0	0	...	0	0	0
8462	1.0	-0.840279	-0.934701	-1.378622	0.876981	-0.654041	-0.132487	-0.17494	0	0	...	0	0	0
8463	1.0	-0.881587	-0.837068	-0.701824	0.698014	-0.654041	-0.132487	-0.17494	0	0	...	0	0	0
8464	1.0	-0.898110	-0.739434	-0.411767	0.780109	-0.654041	-0.132487	-0.17494	0	0	...	0	0	0

B2. Randomly Selected Features:

8 random features are selected for building models and compare the training and test errors with the full model. The selected 8 features are: ['day', 'temperature', 'hour', 'visibility', 'holiday', 'rainfall', 'solar_radiation', 'season']

B3. Important Features:

Top 8 features are selected using random forest classifier. The default method to compute variable importance is the mean decrease in impurity (or Gini importance) mechanism: At each split in each tree, the improvement in the split-criterion is the importance measure attributed to the splitting variable and is accumulated over all the trees in the forest separately for each variable. Below table shows the decreasing order of feature importance.

feature	importance
temperature	0.154601
humidity	0.144303
hour	0.144045
wind_speed	0.142666
visibility	0.132103
day	0.125397
solar_radiation	0.080190
season	0.042071
rainfall	0.012886
snowfall	0.011428
holiday	0.010311
intercept	0.000000

Selected features are: ['temperature', 'humidity', 'wind_speed', 'hour', 'visibility', 'day', 'solar_radiation', 'season']

B4. Train and Test Sets:

The datasets are randomly split in 70/30 ratio to create training and test sets.

C. Linear Regression:

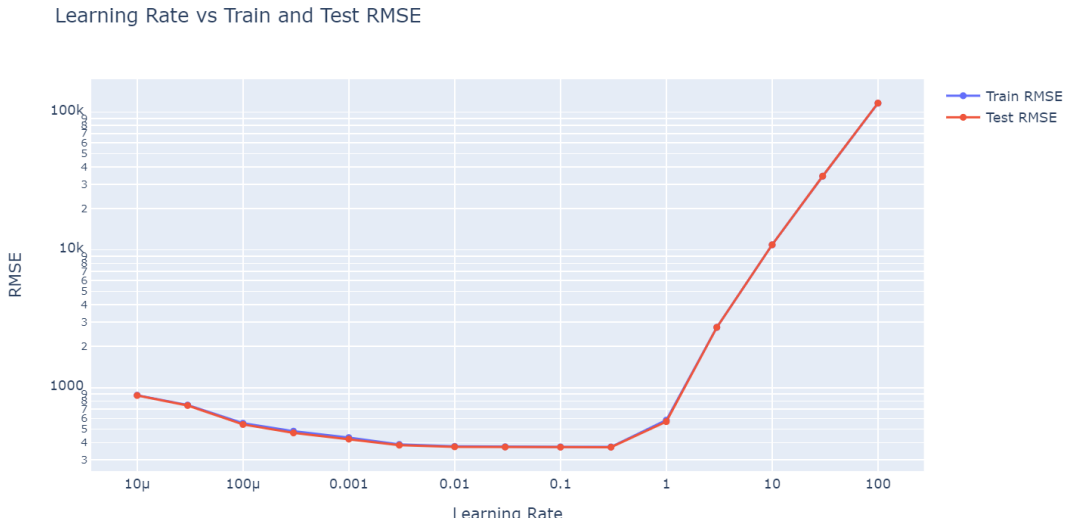
Maximum number of iterations are capped at 10000.

Threshold determines the minimum percentage change in cost function required at each update. If the change is lower than threshold, gradient descent comes out of loop and reports the cost function and thetas at that point (convergence).

C1. Experiment 1:

For threshold = 0.001

	learning_rate	converging_iteration	train_rmse	test_rmse
0	100.00000	1	116083.853594	116274.242687
1	30.00000	1	34260.136163	34313.067235
2	10.00000	1	10890.041227	10903.277327
3	3.00000	1	2746.731318	2744.268895
4	1.00000	3	582.646878	568.464602
5	0.30000	435	372.199252	370.665470
6	0.10000	1022	372.514622	370.863688
7	0.03000	2534	373.443047	371.520436
8	0.01000	5597	375.690491	373.213588
9	0.00300	10000	388.240869	383.398296
10	0.00100	10000	435.065775	424.302269
11	0.00030	10000	484.541685	470.722002
12	0.00010	10000	552.993489	541.764324
13	0.00003	10000	749.988650	744.042925
14	0.00001	10000	883.534153	879.716272



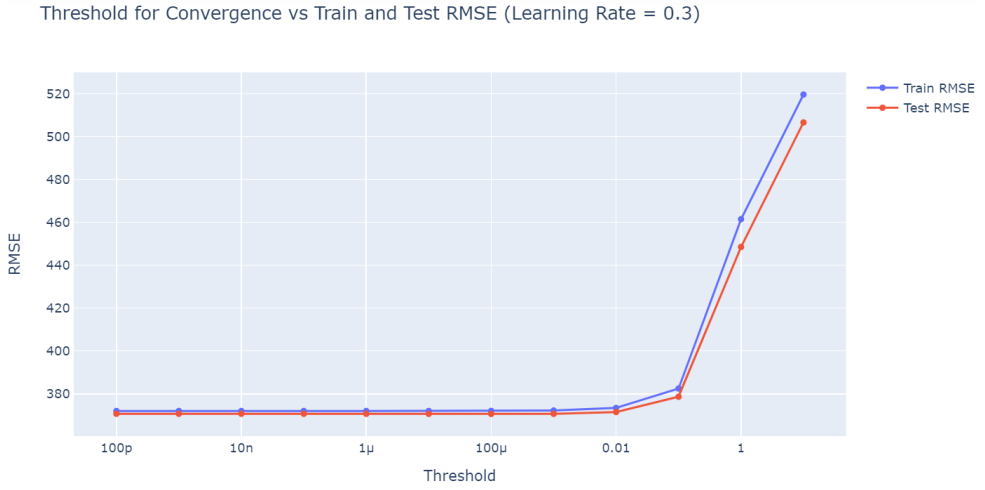
For higher learning rates, the gradient descent does not converge at minimum and it overshoots. As a result, both train and test RMSE are very high and converging iteration is less than 5. On the other hand, when the learning rate is very low, the gradient descent requires more iterations to converge. Since the maximum iteration is capped at 10000, it has not converged to global minimum and therefore the train and test RMSE are comparatively higher that at the optimum learning rate.

The learning rate with minimum train and test RMSE is 0.3.

C2. Experiment 2:

For learning rate = 0.3

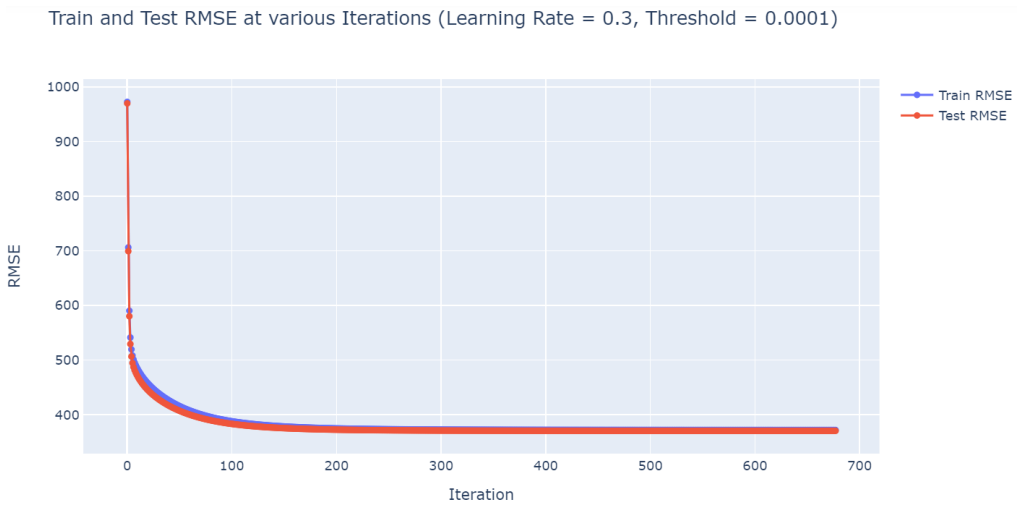
	threshold	converging_iteration	train_rmse	test_rmse
0	1.000000e+01	4	519.635345	506.604443
1	1.000000e+00	18	461.485471	448.541979
2	1.000000e-01	123	382.488304	378.644793
3	1.000000e-02	254	373.415856	371.503301
4	1.000000e-03	435	372.199252	370.665470
5	1.000000e-04	677	372.031792	370.609310
6	1.000000e-05	971	372.011382	370.638199
7	1.000000e-06	1655	372.007824	370.665349
8	1.000000e-07	4106	372.006077	370.681649
9	1.000000e-08	6594	372.005896	370.686741
10	1.000000e-09	9081	372.005878	370.688390
11	1.000000e-10	10000	372.005877	370.688657



Threshold determines the minimum cut-off percentage change in cost function. Higher thresholds might lead to gradient descent not converging at global minimum. As the threshold decreases, there is not much change in the cost function. However, very small thresholds would take a longer time to converge (lot of gradient descent iterations). The train and test RMSE do not change much by decreasing the threshold any further than optimum threshold.

The optimum threshold is 0.0001.

The below plot shows the train and test RMSE at every iteration for chosen learning rate of 0.3 and threshold of 0.0001.



The cost function decreases at every iteration so does the train and test RMSE. At 678th iteration, the gradient descent comes out of the loop with minimum cost function and reports the thetas at that point.

Thetas of our Linear Regression Model are compared to that of Scikit-Learn Linear Regression Model. Below is the table that summarizes the coefficients.

	Gradient Descent Model	Scikit-Learn Model
intercept	897.757102	892.029220
temperature	293.700559	293.283353
humidity	-141.868142	-141.458031
wind_speed	-3.911310	-3.718239
visibility	3.305475	3.028384
solar_radiation	58.861576	63.532383
rainfall	-69.704652	-69.348861
snowfall	16.719622	16.775978
hour:_1	-114.694774	-99.261872
hour:_2	-211.469206	-196.559326
hour:_3	-324.875268	-310.025063
hour:_4	-355.145167	-340.450310
hour:_5	-368.038054	-353.476735
hour:_6	-188.171530	-173.576352
hour:_7	143.289201	157.069390
hour:_8	492.947546	504.882751
hour:_9	30.052970	38.756510
hour:_10	-219.577522	-214.633625
hour:_11	-213.535800	-211.100737
hour:_12	-173.292292	-172.946550
hour:_13	-159.851573	-159.552745
hour:_14	-175.933212	-175.210211
hour:_15	-93.584638	-90.403959
hour:_16	57.145310	63.083312
hour:_17	316.795477	326.714060
hour:_18	803.459832	816.133342
hour:_19	542.180385	557.623382
hour:_20	468.632955	484.191405
hour:_21	432.983003	448.981008
hour:_22	343.788091	359.588174
hour:_23	106.103572	121.540974
day:_Monday	-25.958352	-29.739820
day:_Saturday	-69.804907	-74.210505
day:_Sunday	-125.013993	-128.926355
day:_Thursday	-24.151594	-28.144822
day:_Tuesday	2.061177	-1.856631
day:_Wednesday	7.149527	2.985237
season:_Spring	-165.191172	-166.915874
season:_Summer	-174.004557	-175.684399
season:_Winter	-348.093807	-347.898281
holiday:_Holiday	-127.809745	-128.321141

C3. Experiment 3 and 4:

For learning rate = 0.3 and threshold = 0.0001

```
Training RMSE (All variables): 372.03
Test RMSE (All variables): 370.61

Training RMSE (8 random variables): 381.69
Test RMSE (8 random variables): 378.48

Training RMSE (8 important variables): 379.64
Test RMSE (8 important variables): 377.48
```

The train and test RMSE of the model containing all features is the lowest of all, followed by the model containing 8 important variables. The model containing random features has the highest train and test RMSE. This can be attributed to the fact that features included in feature importance method explain the variation in target variable better than the one that were selected randomly. This proves the fact that by adding important variables, there is greater mean decrease in impurity (Gini) and better predictive power.

D. Logistic Regression:

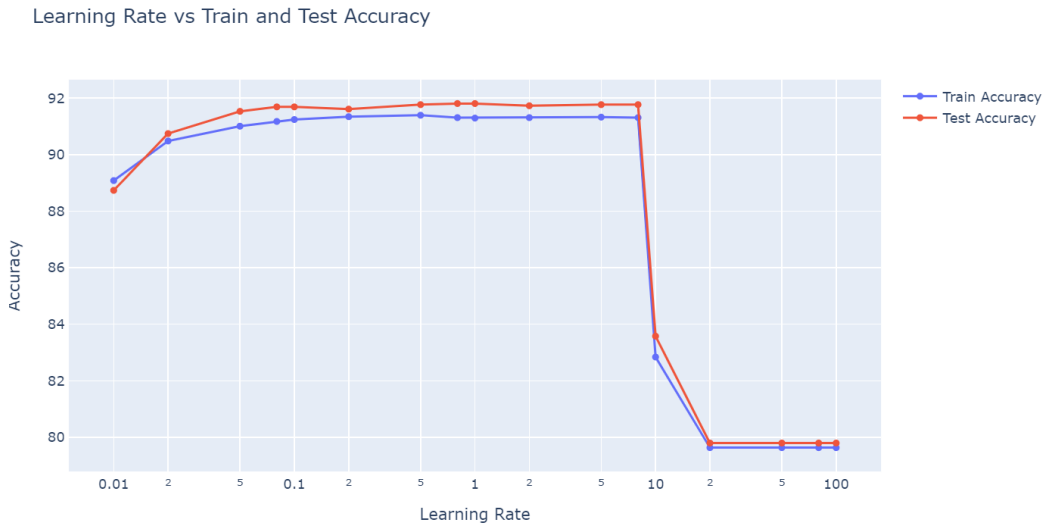
Maximum number of iterations are capped at 10000.

Threshold determines the minimum percentage change in cost function required at each update. If the change is lower than threshold, gradient descent comes out of loop and reports the cost function and thetas at that point (convergence).

D1. Experiment 1:

For threshold = 0.001

	learning_rate	converging_iteration	train_accuracy	test_accuracy
0	100.00	1	79.632130	79.795195
1	80.00	1	79.632130	79.795195
2	50.00	1	79.632130	79.795195
3	20.00	1	79.632130	79.795195
4	10.00	5	82.838340	83.576211
5	8.00	497	91.309484	91.768413
6	5.00	655	91.326358	91.768413
7	2.00	1139	91.309484	91.729027
8	1.00	1742	91.292609	91.807798
9	0.80	1996	91.309484	91.807798
10	0.50	2651	91.393858	91.768413
11	0.20	4526	91.343233	91.610870
12	0.10	6627	91.241984	91.689642
13	0.08	7452	91.174485	91.689642
14	0.05	9442	91.005737	91.532099
15	0.02	10000	90.482619	90.744388
16	0.01	10000	89.082011	88.735723



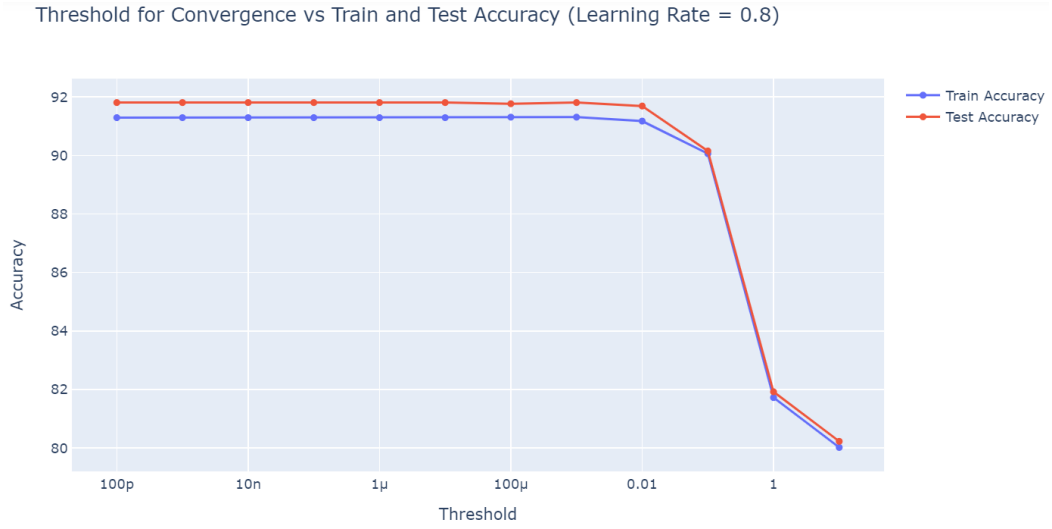
Similar to Linear Regression Model, the gradient descent does not converge at minimum and it overshoots for higher learning rates. As a result, both train and test Accuracy are very low and converging iteration is less than 10. On the other hand, since the maximum iteration is capped at 10000, the gradient descent has not converged to global minimum and therefore the train and test Accuracy are comparatively lower that at the optimum learning rate.

The learning rate with maximum train and test Accuracy is 0.8.

D2. Experiment 2:

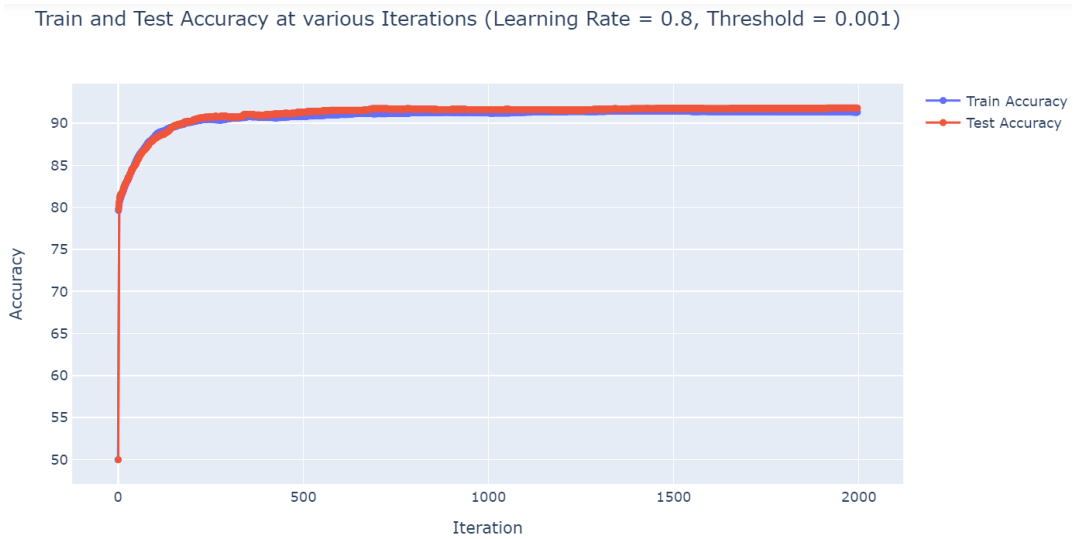
For learning rate = 0.8

	threshold	converging_iteration	train_accuracy	test_accuracy
0	1.000000e+01	2	80.020250	80.228436
1	1.000000e+00	12	81.724603	81.922017
2	1.000000e-01	185	90.060749	90.153604
3	1.000000e-02	745	91.174485	91.689642
4	1.000000e-03	1996	91.309484	91.807798
5	1.000000e-04	4981	91.309484	91.768413
6	1.000000e-05	10000	91.292609	91.807798
7	1.000000e-06	10000	91.292609	91.807798
8	1.000000e-07	10000	91.292609	91.807798
9	1.000000e-08	10000	91.292609	91.807798
10	1.000000e-09	10000	91.292609	91.807798
11	1.000000e-10	10000	91.292609	91.807798



Higher thresholds might lead to gradient descent not converging at global minimum. As the threshold decreases, there is not much change in the cost function. The train and test Accuracy do not change much by decreasing the threshold any further than optimum threshold.

The optimum threshold is 0.001.



The above plot shows the train and test Accuracy at every iteration for chosen learning rate of 0.8 and threshold of 0.001. At 1997th iteration, the gradient descent comes out of the loop with minimum cost function and reports the thetas at that point.

Thetas of our Logistic Regression Model are compared to that of Scikit-Learn Logistic Regression Model. Below is the table that summarizes the coefficients.

	Gradient Descent Model	Scikit-Learn Model
intercept	1.522990	1.667069
temperature	1.756802	1.732322
humidity	-0.836111	-0.821654
wind_speed	-0.199798	-0.196324
visibility	-0.129597	-0.122199
solar_radiation	0.602287	0.568179
rainfall	-2.342514	-2.259437
snowfall	-0.464561	-0.445532
hour:_1	-0.427301	-0.585325
hour:_2	-2.174072	-2.269357
hour:_3	-4.296914	-4.355338
hour:_4	-4.452676	-4.713926
hour:_5	-4.638258	-4.843519
hour:_6	-1.562477	-1.685495
hour:_7	0.474660	0.290248
hour:_8	2.349860	2.069126
hour:_9	1.091654	0.890519
hour:_10	-0.558942	-0.657446
hour:_11	-0.324325	-0.425582
hour:_12	-0.214955	-0.302473
hour:_13	-0.331585	-0.408105
hour:_14	-0.575896	-0.644568
hour:_15	0.249295	0.147573
hour:_16	0.386210	0.257045
hour:_17	1.327133	1.160650
hour:_18	3.140935	2.855433
hour:_19	1.996854	1.747598
hour:_20	1.750728	1.501989
hour:_21	1.568607	1.323797
hour:_22	1.616312	1.367790
hour:_23	1.146147	0.924092
day:_Monday	-0.351435	-0.393666
day:_Saturday	-0.394118	-0.432434
day:_Sunday	-1.102298	-1.125781
day:_Thursday	-0.215837	-0.259868
day:_Tuesday	0.070500	0.014255
day:_Wednesday	-0.205883	-0.254926
season:_Spring	-1.416430	-1.351365
season:_Summer	-0.842487	-0.795850
season:_Winter	-3.851036	-3.731133
holiday:_Holiday	-1.160382	-1.101342

D3. Experiment 3 and 4:

For learning rate = 0.8 and threshold = 0.001

```
Training Accuracy (All variables): 91.31
Test Accuracy (All variables): 91.81

Training Accuracy (8 random variables): 90.75
Test Accuracy (8 random variables): 91.49

Training Accuracy (8 important variables): 89.77
Test Accuracy (8 important variables): 90.78
```

The train and test Accuracy of the model containing all features is the highest of all, followed by the model containing 8 random variables. The model containing important features has the lowest train and test Accuracy. The variables selected using feature importance utilizes continuous target variable instead of binary target variable (required for Logistic Regression). Hence these variables not necessarily need to be important predictors for modelling such a target variable. The randomly selected features have higher explanatory power than the features selected using feature importance.

E. Questions:

What do you think matters the most for predicting the rented bike count?

The date and time categorical variables such as hour, season, day of the week and weather condition variables such as temperature, humidity windspeed and visibility are very important predictors. In fact, summer is the season when bike rentals are very high and, in a day, 5 p.m. to 10 p.m is when the business picks up. Commuters use bikes more often during in weekdays than weekends and holidays.

What other steps you could have taken with regards to modelling to get better results?

The 'bike_count' variable is highly skewed to the right. Running a linear regression model on such a target variable leads to heteroskedastic errors. By transforming the variable, we can take care of such errors. However, the interpretation of the coefficients change.

Another way to model this in a better way is to consider the dataset as time series model and run multi-variate time series forecasting models.