Description of the problem and the data

Cab Booking Evaluation System

The automated process of renting a cab through an app is known as a "cab booking system". Utilizing this app people can order a cab from one location to another.

Objective

- The goal of this project is to combine historical usage patterns with publicly available data sources, such as weather data, to predict if most people will book a cab in a city.
- Based on this the company could determine whether to deploy more or fewer cabs on that specific day.
- Understanding cab supply and demand might improve service effectiveness for the cab booking app firm and improve user experience by reducing waiting times.

Motivation

My friend has breathing problem in India, doctors have suggested that he must carry an inhaler to breathe due to such air pollution. But when we came to Boston, I noticed that he had stopped using it since the air quality is so good that he no longer needs an inhaler. That prompted me to consider how I may lessen this pollution.

In India, air pollution is a major contributor to several ailments, and one of the biggest sources of this pollution is smoke from moving automobiles. The amount of gas emitted in turn increases the number of pollutants released, and which is further increased with the number of vehicles on the road. As a solution, I developed a system to lessen air pollution by lowering the number of cars on the road.

Dataset

	datetime	season	holiday	workingday	weather	temp	atemp	humidity	windspeed	Total_booking
0	5/2/2012 19:00	Summer	0	1	Clear + Few clouds	22.14	25.760	77.0	16.9979	504
1	9/5/2012 4:00	Fall	0	1	Clear + Few clouds	28.70	33.335	79.0	19.0012	5
2	1/13/2011 9:00	Spring	0	1	Clear + Few clouds	5.74	6.060	50.0	22.0028	139
3	11/18/2011 16:00	Winter	0	1	Clear + Few clouds	13.94	16.665	29.0	8.9981	209
4	9/13/2011 13:00	Fall	0	1	Clear + Few clouds	30.34	33.335	51.0	19.0012	184

- This dataset was provided to me by Edukreka.com to work on some test problem on Regression.
- This dataset was used to predict the total number of bookings based on certain features and performs well for Regression task.
- I cannot create this dataset without getting help from some Cob Booking Company like Uber.
- So, I have made some changes to the dataset like changing features for the need of my project.

Descriptions of the columns present in the dataset

- **datetime -** hourly date + timestamp of the booking
- **season -** one of the four periods (spring, summer, fall, and winter) into which the year is commonly divided.
- **holiday -** whether the day is considered a holiday
- **workingday** - when the day is neither a weekend nor holiday
- **temp –** temperature of a day in Celsius
- **weather** – Represents Clear , Cloudy, Light Rain, Heavy
- **atemp -** "feels like" temperature in Celsius
- **humidity -** relative humidity (amount of water vapor in the air)
- **windspeed** - wind speed
- **Total_booking -** number of total bookings on a particular day

A row represents what were the features present at that instant of time and Total Bookings show how many bookings were made at that instant of time.

Example- Like how many people booked a cab at 7 am in the morning on some date.

Creating target Variable for Classification

- “Book" is equal to 1 representing majority of people will book a cab.
- If total bookings are more than median, then Book will be 1 otherwise 0

	datetime	season	holiday	workingday	weather	temp	atemp	humidity	windspeed	Total_booking	Book
0	5/2/2012 19:00	Summer	0	1	Clear + Few clouds	22.14	25.760	77.0	16.9979	504	1
1	9/5/2012 4:00	Fall	0	1	Clear + Few clouds	28.70	33.335	79.0	19.0012	5	0
2	1/13/2011 9:00	Spring	0	1	Clear + Few clouds	5.74	6.060	50.0	22.0028	139	0
3	11/18/2011 16:00	Winter	0	1	Clear + Few clouds	13.94	16.665	29.0	8.9981	209	1
4	9/13/2011 13:00	Fall	0	1	Clear + Few clouds	30.34	33.335	51.0	19.0012	184	1

Based on the following characteristics, the model will determine if the majority of people will order a cab.

- **datetime -** Time of day heavily depends on bookings made as people tend to make less bookings during midnight.
- **season -** People may use cabs even for shorter distances in winter to save themselves from cold.

- **holiday -** May decrease number of bookings as persons who use cabs for offices or colleges on daily basis will not go
- **workingday -** May increase number of bookings opposite of holiday
- **temp -** May decrease it, as more people would like to stay at home as temperature rises.
- **weather -** People may use more cabs in the rainy season to save them from rain or snow.
- **humidity -** People may tend to stay at home due to higher humidity.
- **windspeed -** High wind speed may increase the number of cab bookings.

Analyses of the Dataset

1. I have used a separate dataset for training and testing. Furthermore, I have divided the training set into train and validation set.
2. Training Data contains 8708 records with 10 columns
3. Test Data contains 2718 records with 10 columns
4. Dataset contains both numerical and categorical data type
5. There are no duplicated values in the dataset
6. The data has following data types-

```
datetime          object
season            object
holiday            int64
workingday         int64
weather           object
temp             float64
atemp            float64
humidity         float64
windspeed        float64
Total_booking      int64
Book               int64
dtype: object
```

Checking for missing values

- Two values of humidity were missing,
- No other features have missing values.

Correcting missing values

- Used the mean of humidity based on seasons to fill the missing values.

```
season
Fall      63.877057
Spring    55.979859
Summer    60.811561
Winter    65.818390
Name: humidity, dtype: float64
```

Feature Engineering

- Cannot use datetime feature .
- Extracting date hours, minutes, and weekday from Datetime columns
- Retrieving relevant information and removing datetime feature.

	datetime	Hour	Weekday	Month	Date
0	5/2/2012 19:00	19	Wednesday	5	2012-05-02
1	9/5/2012 4:00	4	Wednesday	9	2012-09-05
2	1/13/2011 9:00	9	Thursday	1	2011-01-13
3	11/18/2011 16:00	16	Friday	11	2011-11-18
4	9/13/2011 13:00	13	Tuesday	9	2011-09-13

Converting object type to Categorical Variables-

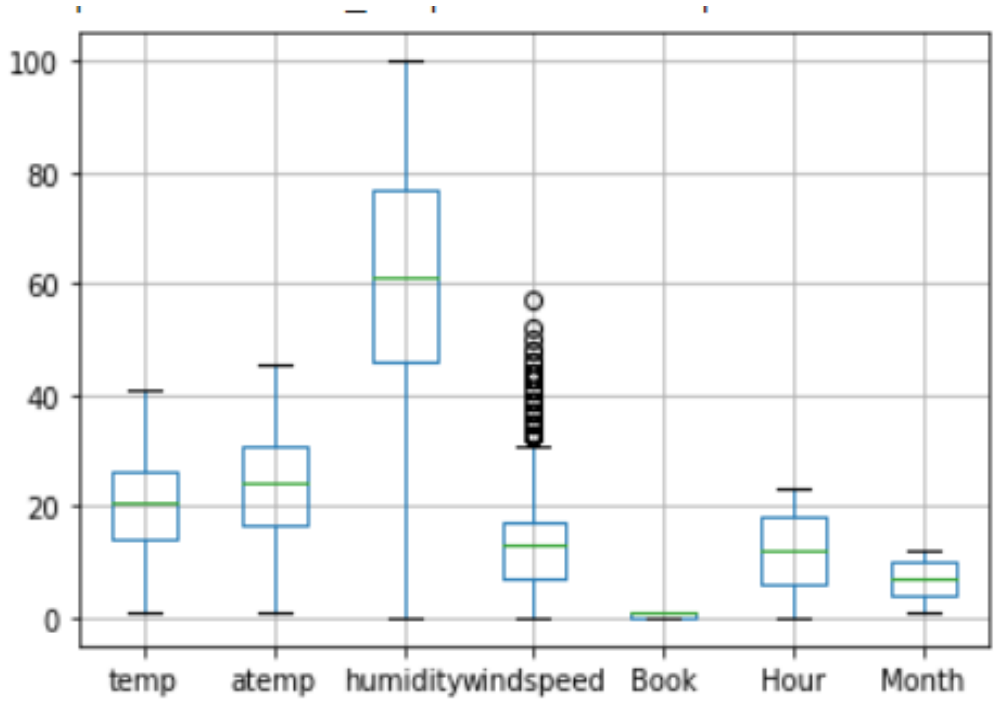
so that numerical operation cannot be performed on them.

```
datetime           object
season           category
holiday          category
workingday       category
weather          category
temp              float64
atemp             float64
humidity          float64
windspeed         float64
Total_booking       int64
Book                int64
Hour                int64
Month               int64
Weekday          category
Date               object
dtype: object
```

Data Cleaning and Visualization

Outlier Analysis

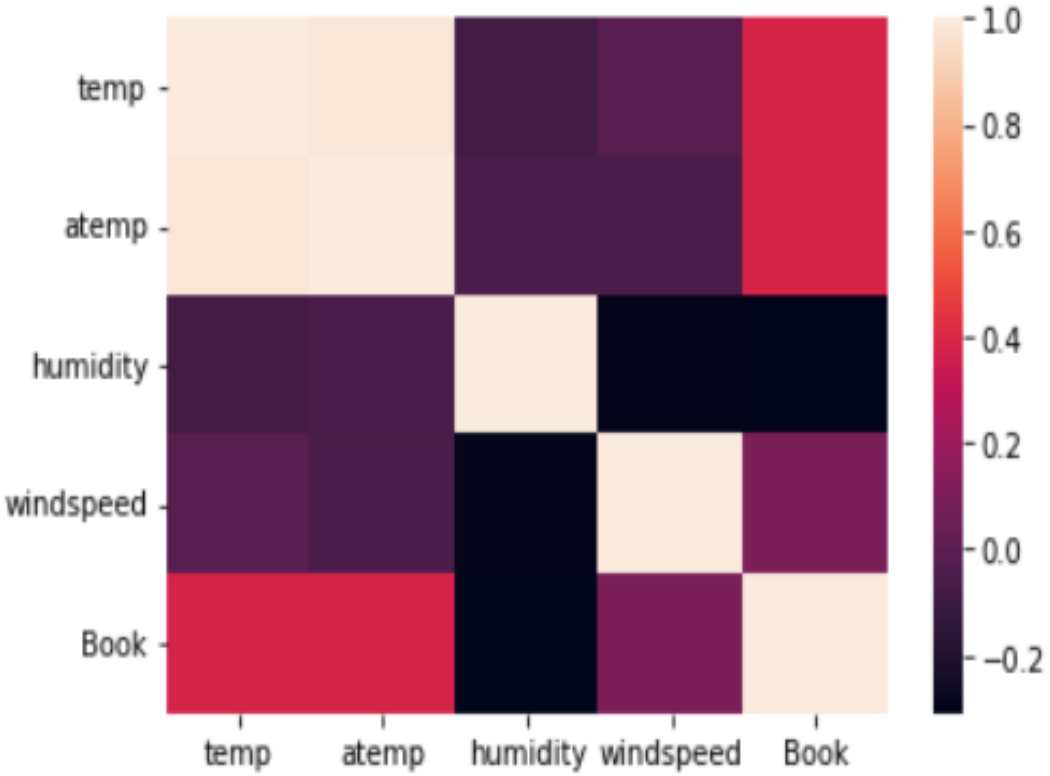
- Only windspeed has outliers.
- Removing outlier by finding-
- Interquartile Range Formula= Quartile3 - Quartile1
- Finding which points are below (q1 - IQR*1.5) and
- which points are above (q1 + IQR*1.5)



Inference

1. Only windspeed contains outliers in the datafile.
2. All outliers were present above third quantile.
3. A total of 182 records were found as outliers.
4. After removing outliers, 8526 rows were left.
5. Training with outlier causes training accuracy 79%.
6. Training without outliers slightly increases training accuracy to 80%.

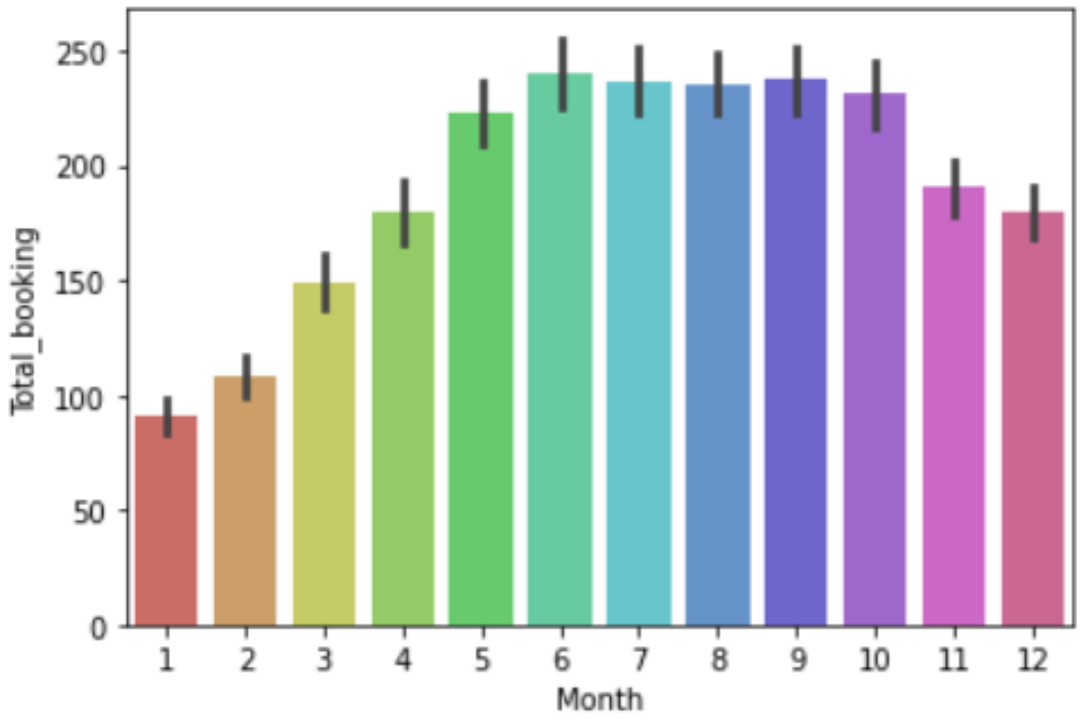
Correlation



1. temp and humidity moderately affect Total_bookings.
2. windspeed slightly influences Total_bookings.
3. temp and atemp are highly correlated so need to remove one to reduce redundancy.
4. Don't need to use holiday and working day as they give basically the same information.

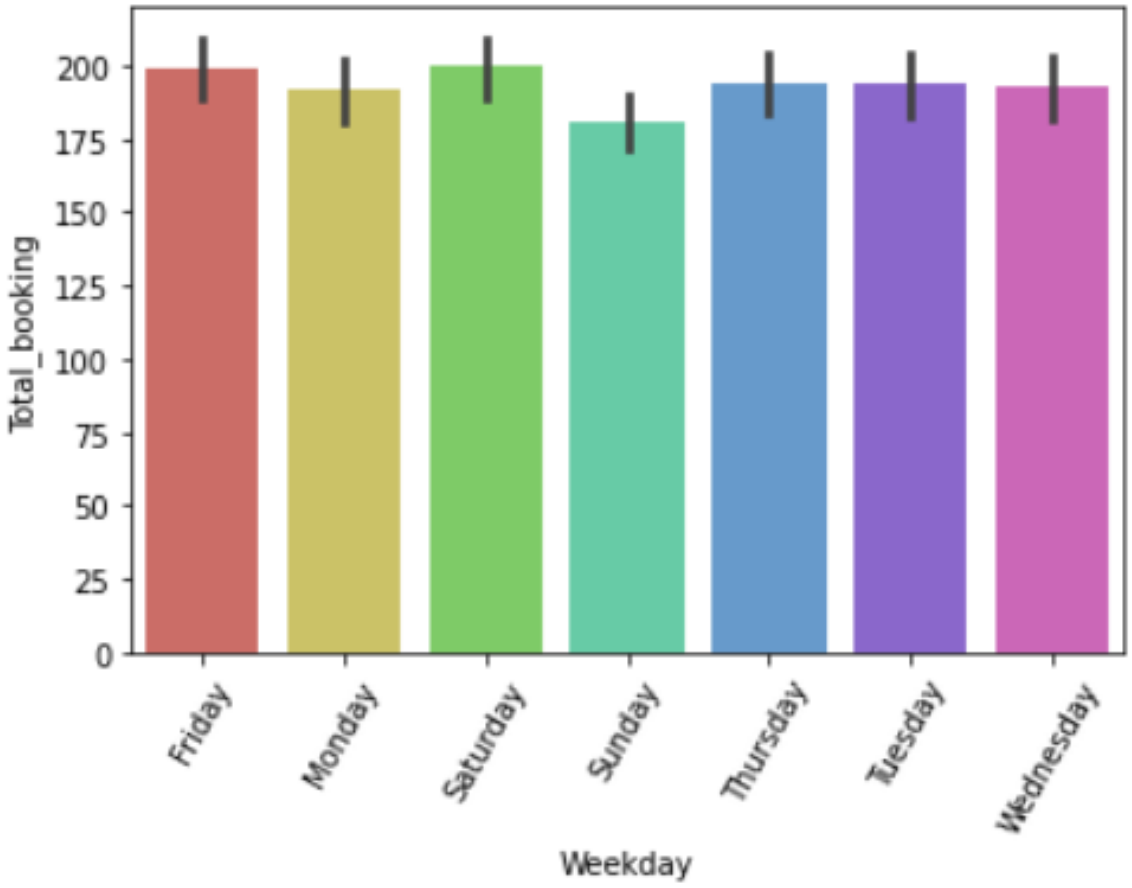
Data Visualization

Total number of Bookings vs Month



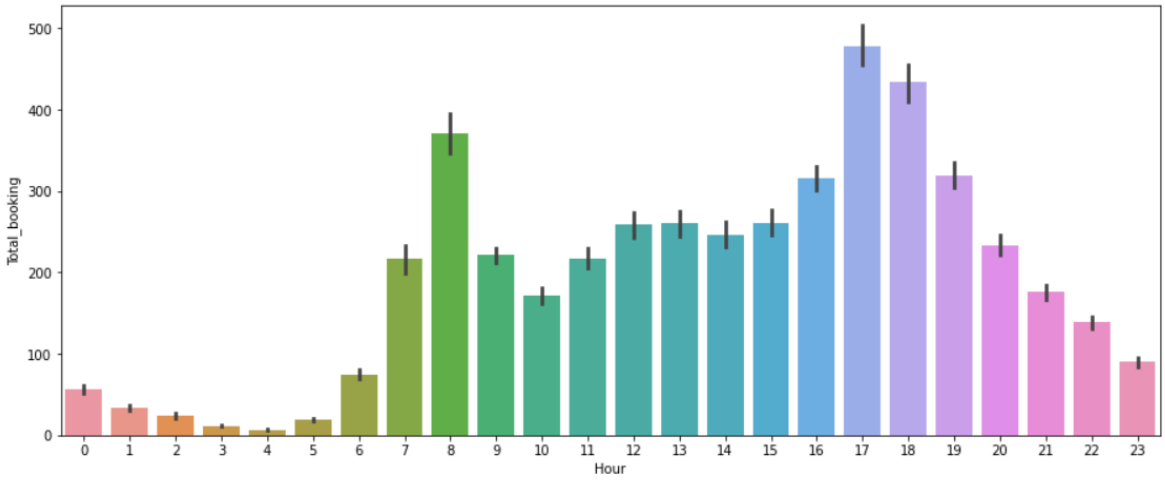
- January and February have very a smaller number of bookings
- July to October have approximately the same number of bookings.
- There is a slight decrease in number of bookings after October

Total number of Bookings vs Weekday



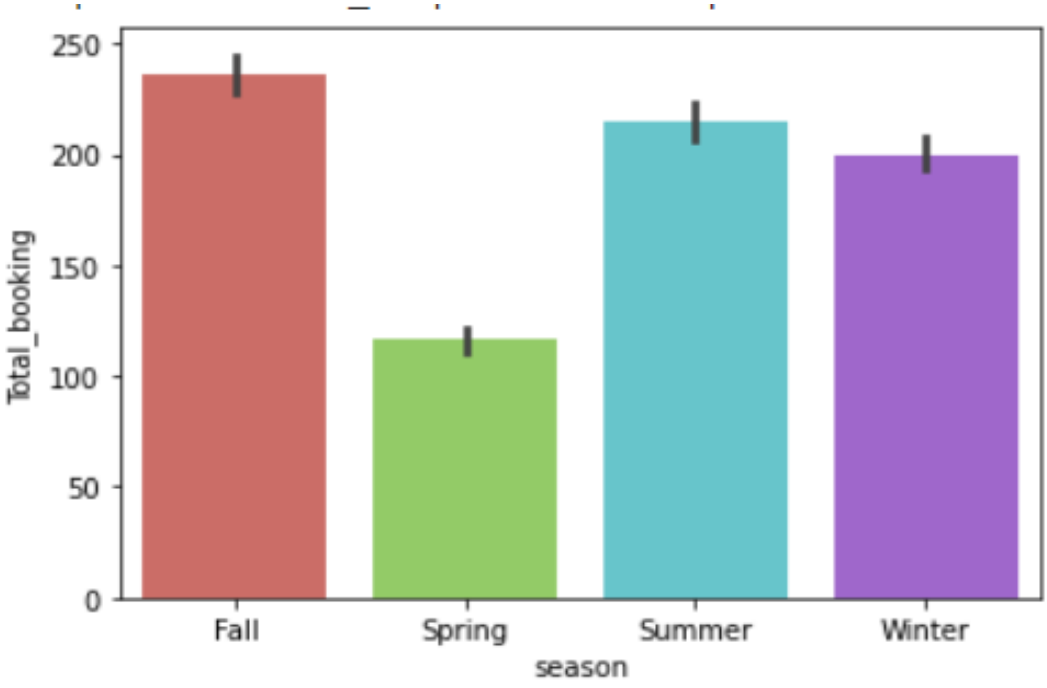
- Weekdays has approximately no effect on number of bookings
- Sunday has relatively a smaller number of bookings
- May be because of holiday.

Total number of Bookings vs Hour



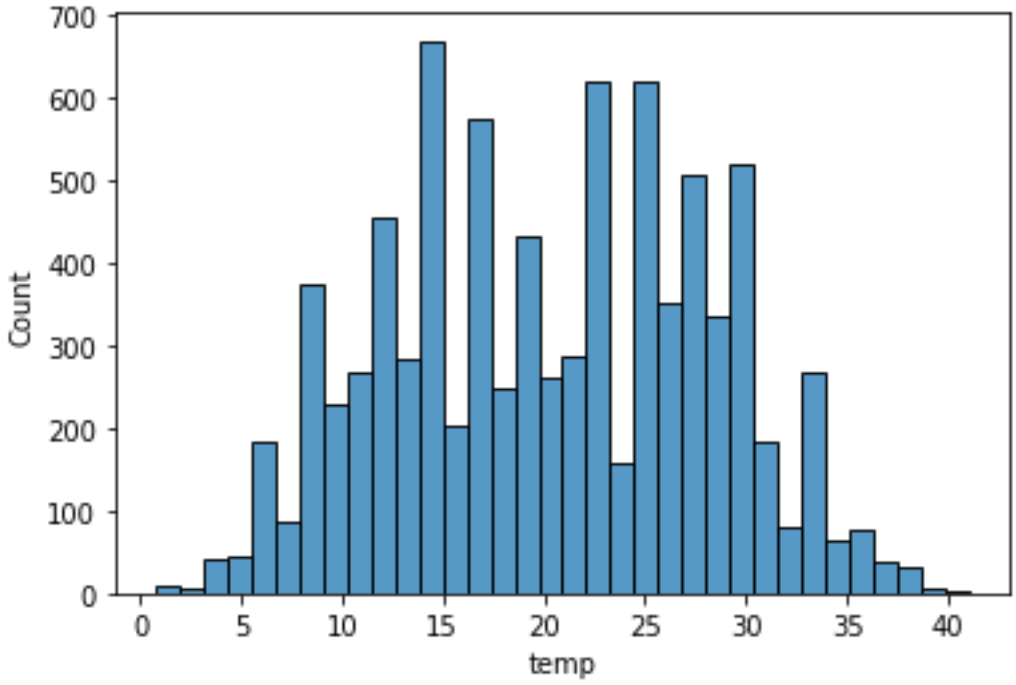
- People generally don't book cabs late at night.
- Bookings are high in the morning around 8 and at noon around 6 am. Maybe due to office hours.

Total number of Bookings vs Season



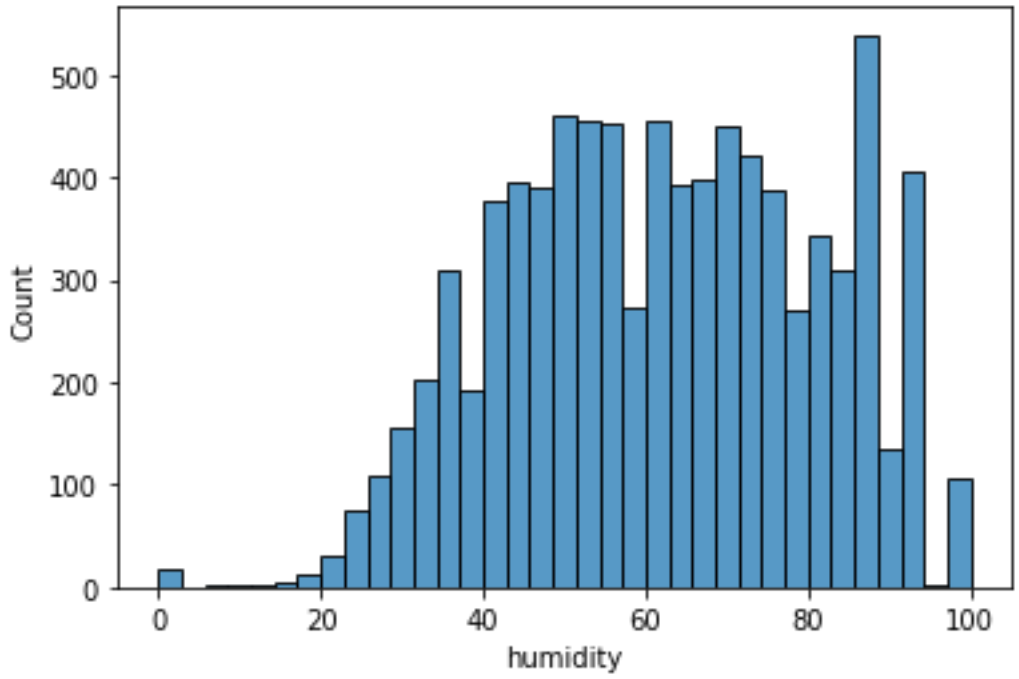
- People use less cabs in spring season.
- Maybe they like to walk and enjoy weather
- Fall has the maximum number of bookings.

Distribution of temperature



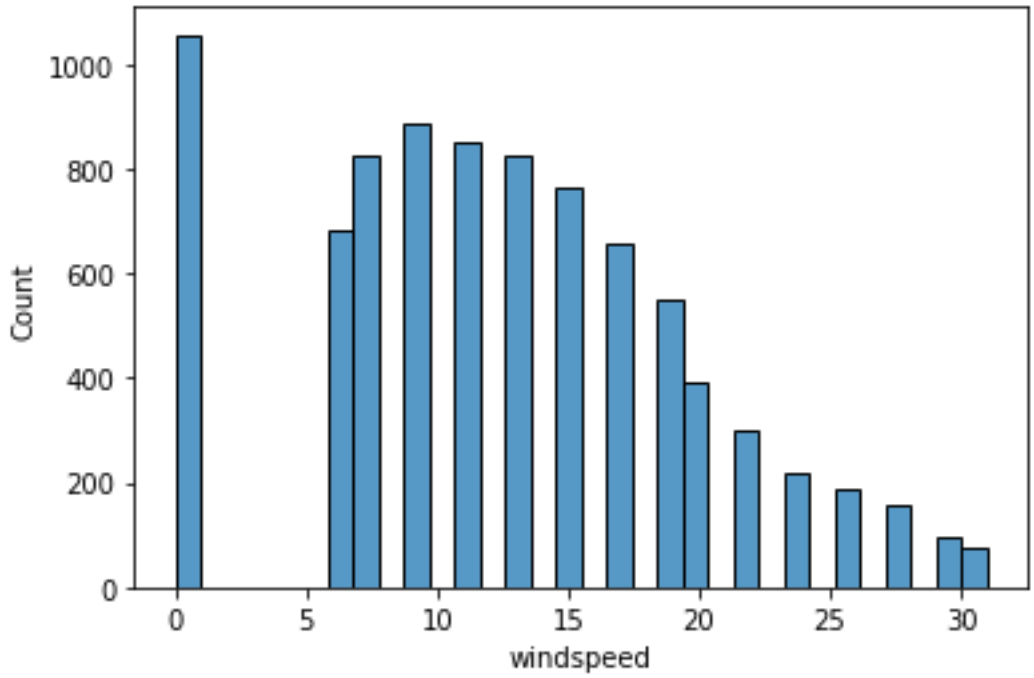
Somewhat normally distributed in range 0-40 Celsius.

Distribution of humidity



Slightly skewedly distributed in the range 0-100.

Distribution of windspeed



Skewedly distributed in the range 0-30.

Feature Scaling

	temp	humidity	windspeed	Total_booking	Book	Hour	Month
count	8526.000000	8526.000000	8526.000000	8526.00000	8526.000000	8526.000000	8526.000000
mean	20.290528	62.005613	12.342809	193.02639	0.500469	11.537180	6.578231
std	7.810630	19.005977	7.491369	181.64990	0.500029	6.910977	3.430779
min	0.820000	0.000000	0.000000	1.00000	0.000000	0.000000	1.000000
25%	13.940000	47.000000	7.001500	43.00000	0.000000	6.000000	4.000000
50%	20.500000	62.000000	11.001400	148.00000	1.000000	12.000000	7.000000
75%	26.240000	77.000000	16.997900	286.00000	1.000000	18.000000	10.000000
max	41.000000	100.000000	31.000900	977.00000	1.000000	23.000000	12.000000

- Humidity- high spread
- Temp- medium spread
- Windspeed- medium spread
- Hours and Month- low spread

One hot Encoder

- Cannot use features like Season, Weekday, and weather directly.
- Converted in numerical form using one hot encoding.
- Encode categorical features as a one-hot numeric array.
- It ensures that model does not assume that higher numbers are more important

	season	Summer	Fall	Spring	Winter
0	Summer	1.0	0.0	0.0	0.0
1	Fall	0.0	1.0	0.0	0.0
2	Spring	0.0	0.0	1.0	0.0
3	Winter	0.0	0.0	0.0	1.0
4	Fall	0.0	1.0	0.0	0.0

- Similarly for weekdays and weather

	holiday	temp	humidity	windspeed	Total_booking	Book	Hour	Month	Date	Clear + Few clouds	...	Spring	Summer	Winter	Friday	Monday	Saturday	Sunday	Thursday	Tuesday	Wednesday
0	0	0.236803	0.788976	0.621430	504	1	19	5	2012-05-02	1.0	...	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0
1	0	1.076733	0.894213	0.888860	5	0	4	9	2012-09-05	1.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0
2	0	-1.863023	-0.631713	1.289558	139	0	9	1	2011-01-13	1.0	...	1.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0
3	0	-0.813110	-1.736693	-0.446501	209	1	16	11	2011-11-18	1.0	...	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0	1.286716	-0.579095	0.888860	184	1	13	9	2011-09-13	1.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0

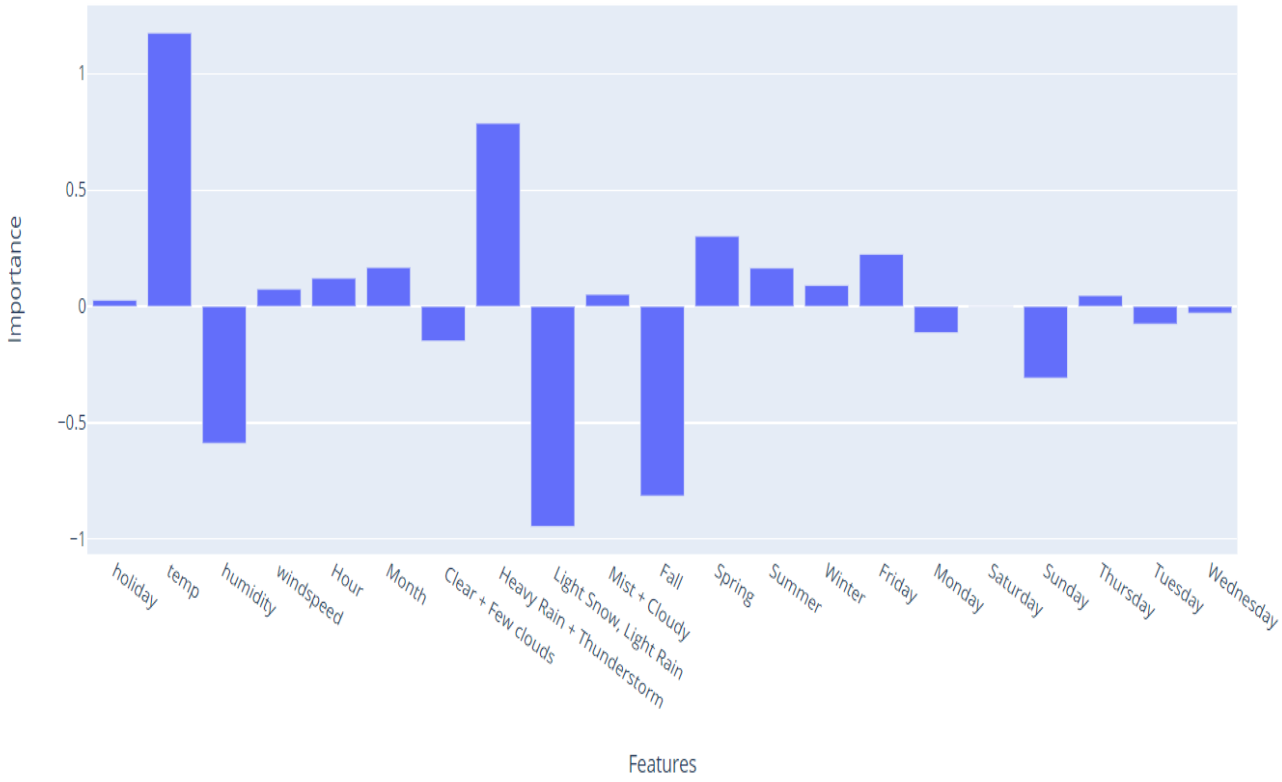
5 rows × 24 columns

Data Preprocessing

- Spit data into training, validating, and testing
- Train shape- (5968, 21)
- Valid shape- (2558, 21)
- Test shape- (2178, 21)

Features vs their importance in the model

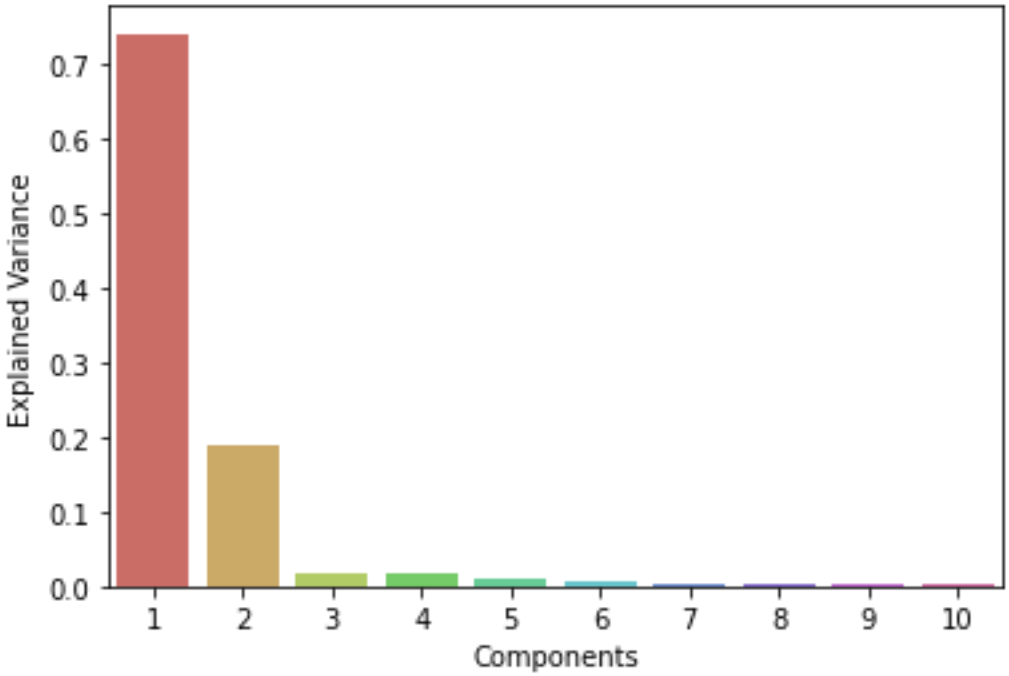
The coefficients of the Logistic Regression model are as follows



- The larger the absolute value of the coefficients the greater is their importance in predicting.
- Larger value of temp, humidity, and weather shows that these features are more important.

PCA

Too many features, so need to reduce the number of features to increase the speed of learning, applying Principal Component Analysis.



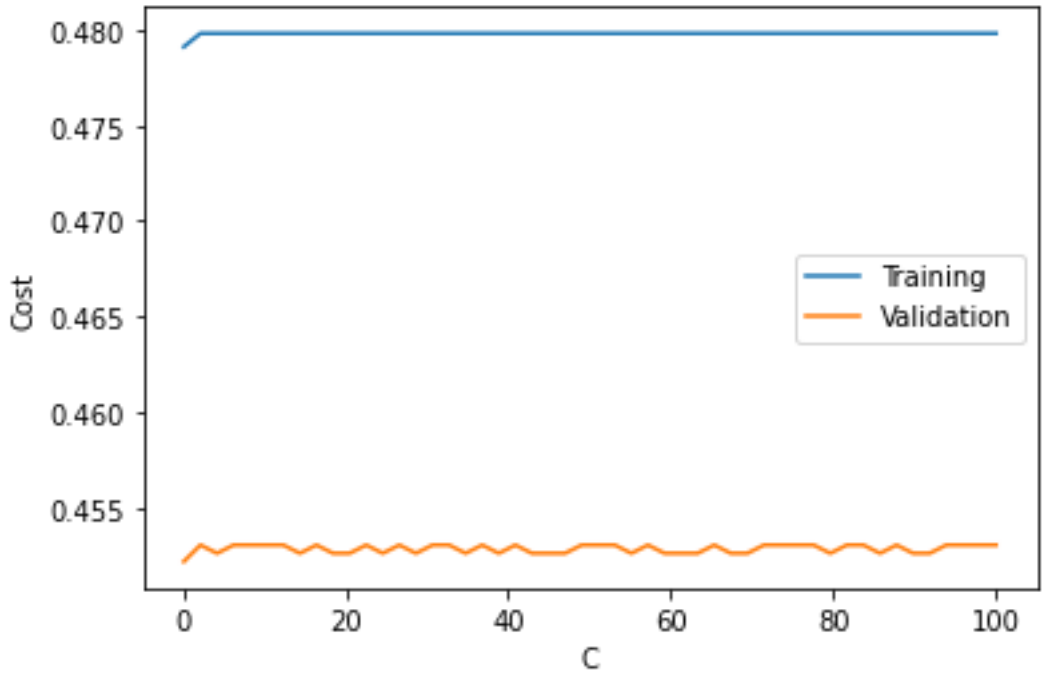
- Principal component Analysis to reduce the number of features.
- First two PCA components explained about 92% of the variance
- First five PCA components explained about 97.5% of the variance
- So, using the first five components for our model.

Parameter tuning with charts

Logistic Regression

Cost VS C curve

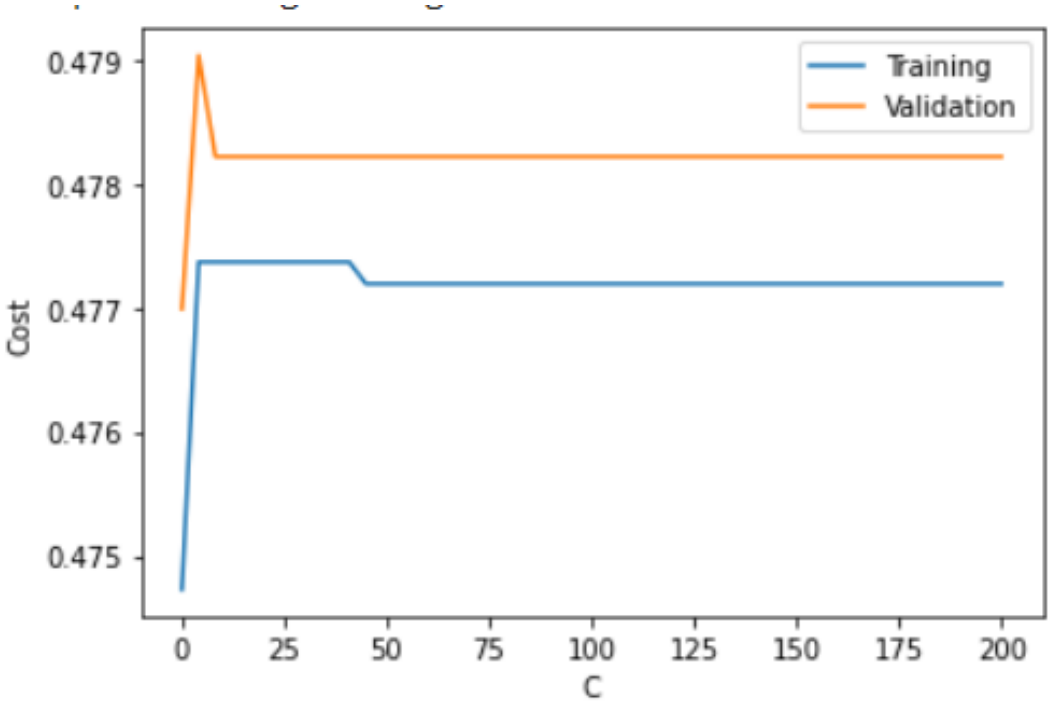
For L1 regularization



- Adjusting C to more than 0.1 increases cost
- Increasing C to more than 1 does not affect our model
- Minimum Cost of 0.455 achieved at C=0.1

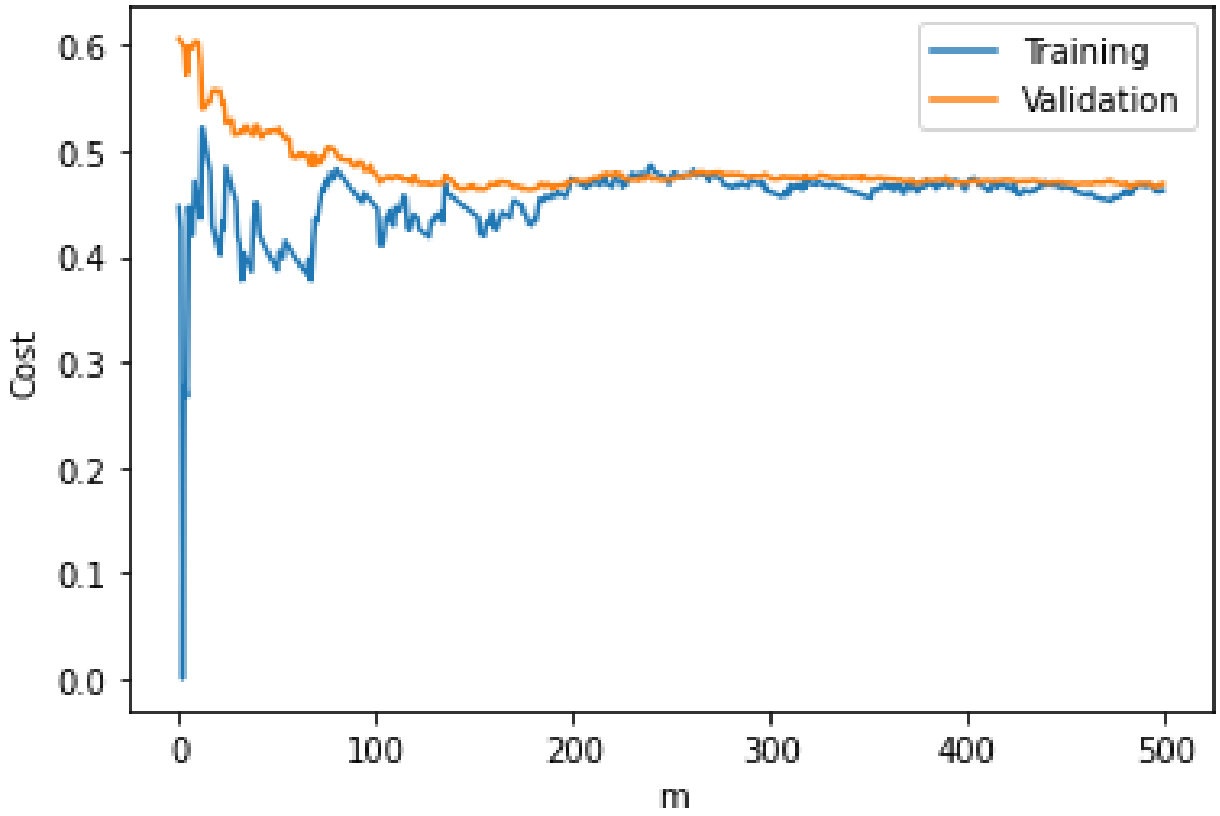
Cost VS C curve

For L2 regularization



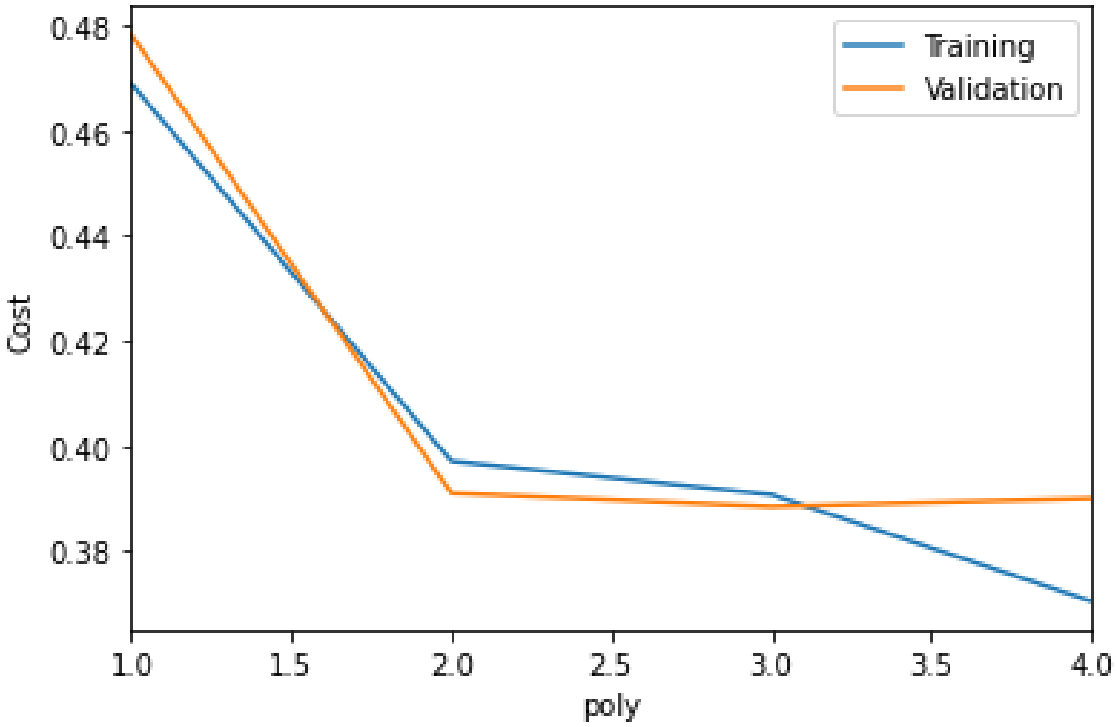
- C- Inverse of regularization strength; Like in support vector machines, smaller values specify stronger regularization.
- Adjusting C to more than 0.1 increases cost
- Increasing C to more than 1 does not affect our model
- Minimum Cost of 0.451 achieved at C=0.1 slightly better than L1 regularization

Cost VS Number of Training examples



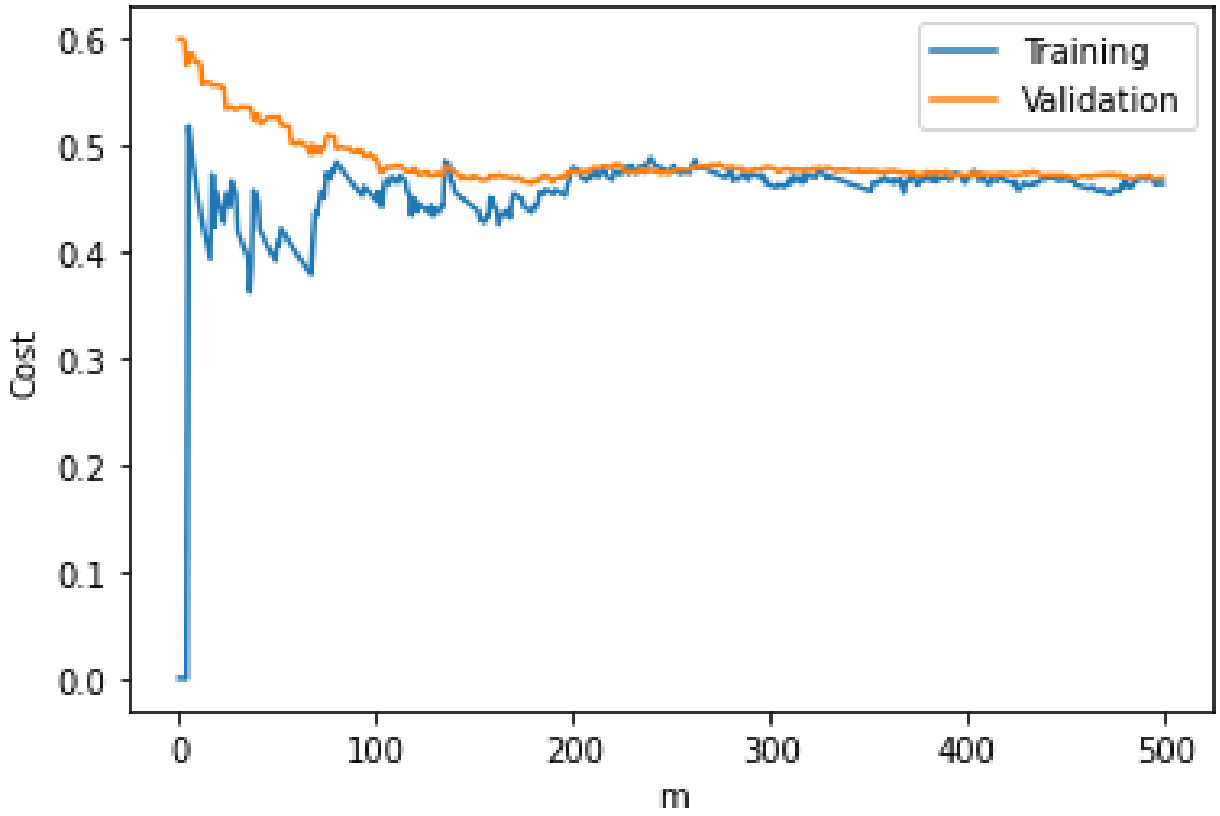
- At starting graph is little distorted due to bias toward towards single target value
- After using more than 200 training examples model starts to perform well
- Cost stabilizes around 0.45 and slowly decreases as we increase training examples.
- Slightly underfitting

Cost VS Polynomial Feature



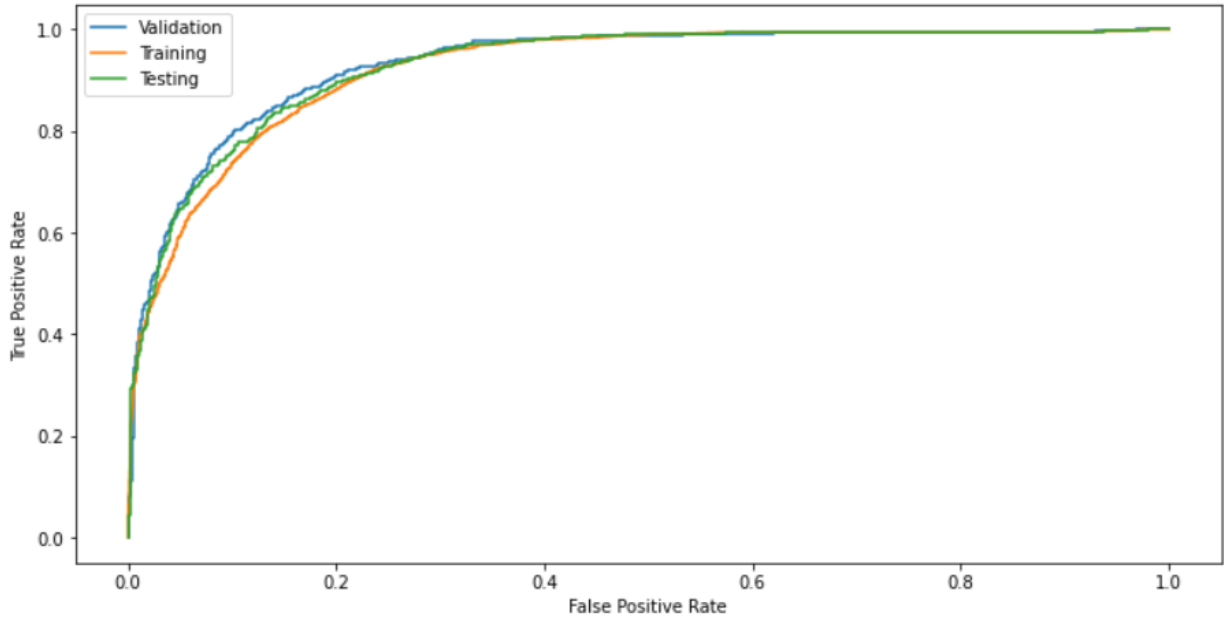
- Using polynomial features to decrease cost.
- After increasing polynomial features more than 2 degrees, model starts to overfit.
- Though training cost is less on three degrees polynomial features it does not decrease validation cost that much.

Cost vs Number of training data



- With 2 degrees of polynomial
- Cost keeps on decreasing slowly
- With full training set it comes down to 0.39

ROC curve

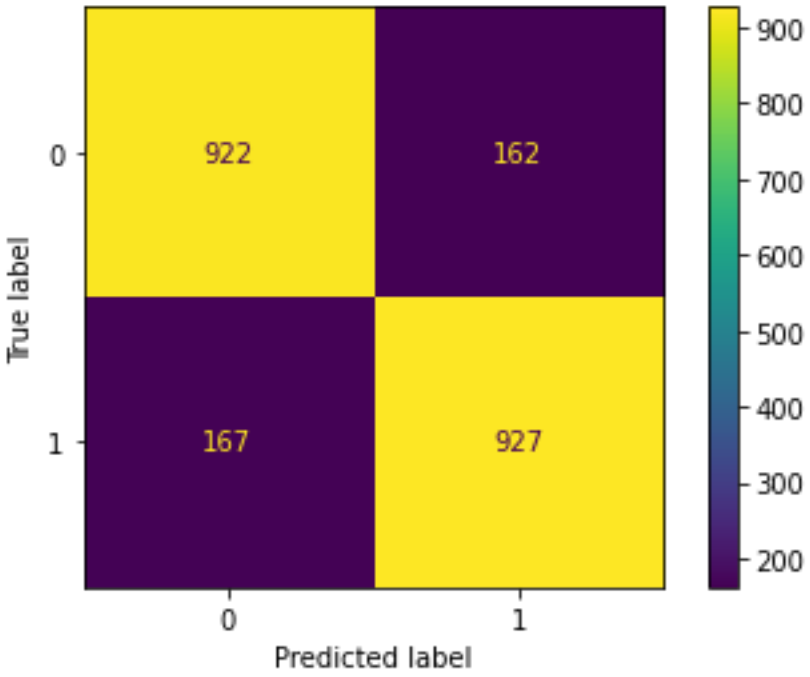


- AUC score- 0.8473
- Minimum validation Cost 0.39
- Model performance is good and somewhat similar in all test cases.

Classification and Confusion Matrix

	precision	recall	f1-score	support
0	0.84	0.85	0.85	1084
1	0.85	0.84	0.85	1094
accuracy			0.85	2178
macro avg	0.85	0.85	0.85	2178
weighted avg	0.85	0.85	0.85	2178

- Reached AUC score 85 on test data.

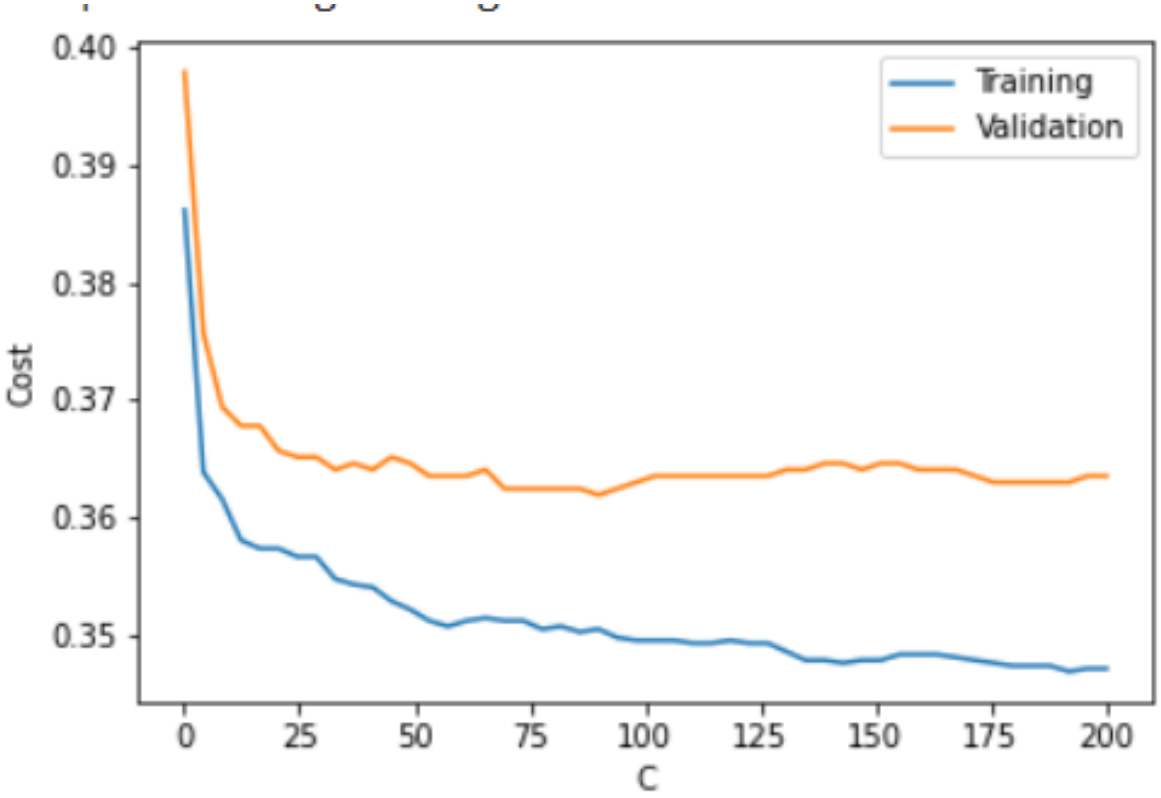


- Model predicted 162 false positive.
- Model predicted 167 false negative.
- Models perform equally for both classes.

Generation and tuning of alternative models

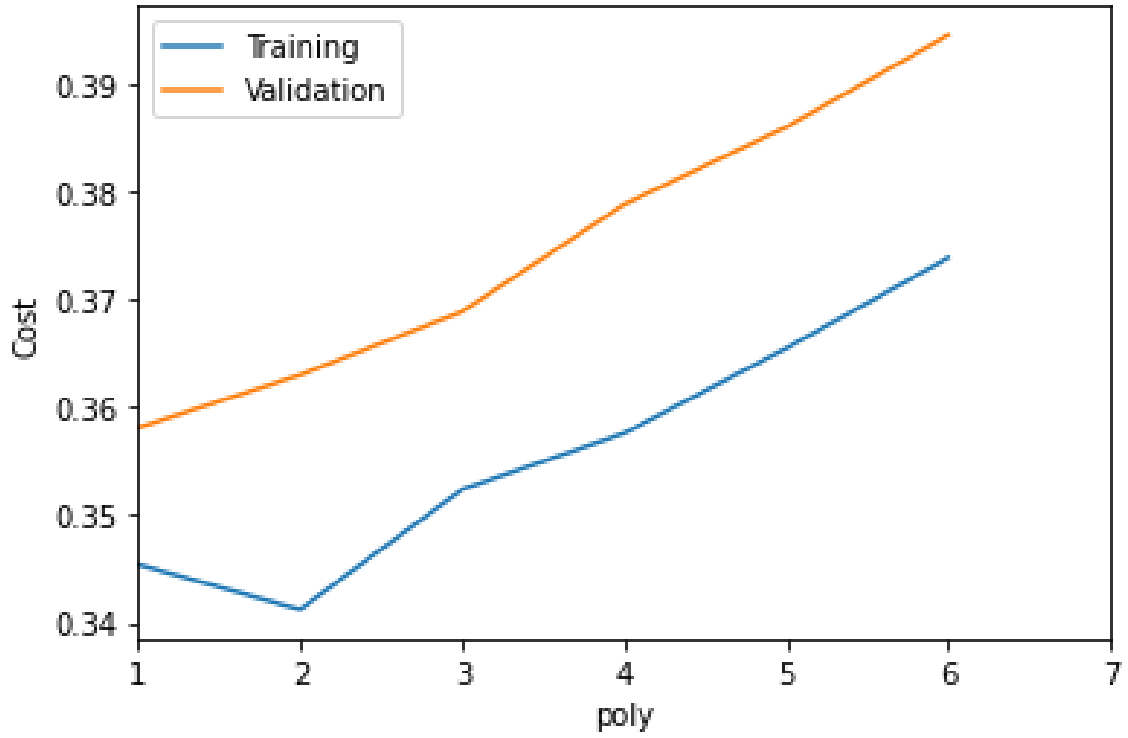
Support Vector Classifier

Cost vs C



- As C increases cost decreases.
- Elbow curve is formed around C=20
- Increasing the value of C to more than 50 does not decrease the cost for validation data.
- The cost is lowest at Regularization C value 50.

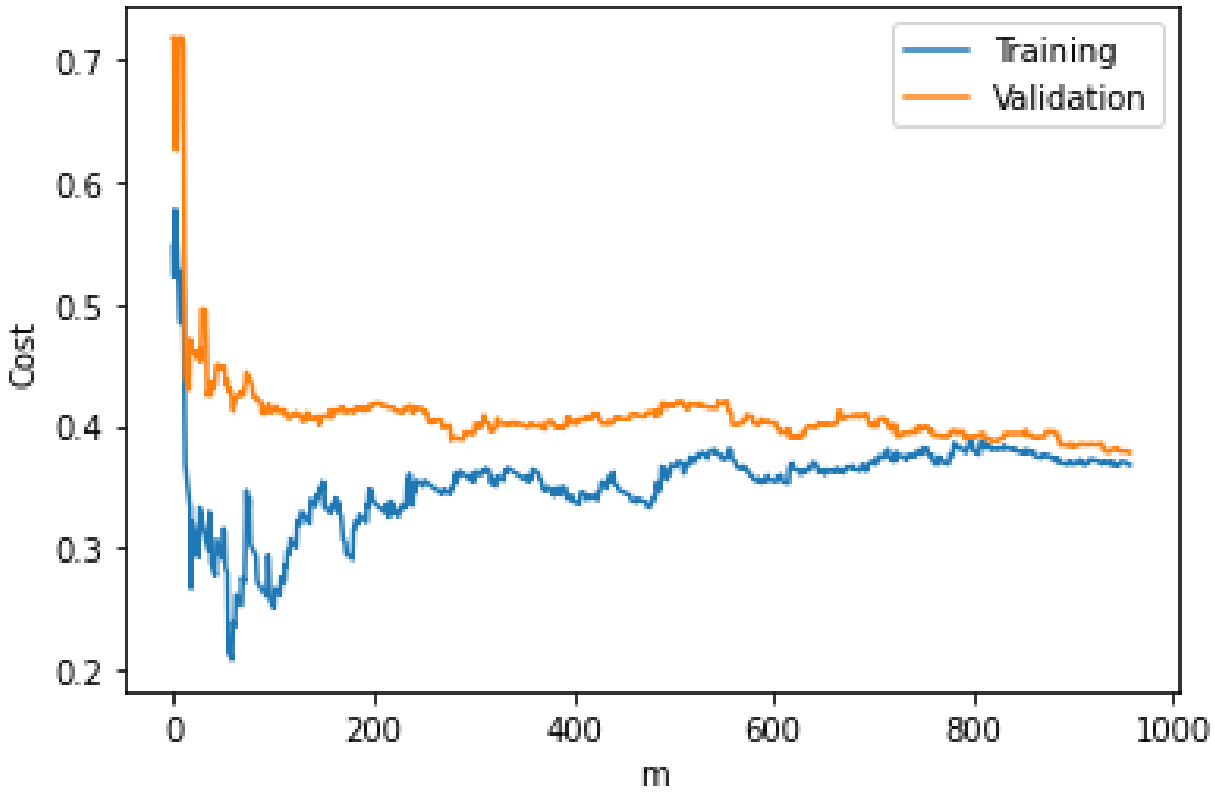
Cost vs degrees of Polynomials



- Though training Cost is decreased when I used 2 degree of polynomial, validation cost is increased, a case of overfitting.
- Training cost also starts increasing when we go beyond 2 degrees of polynomial.
- So, no polynomial features are used for SVC model.

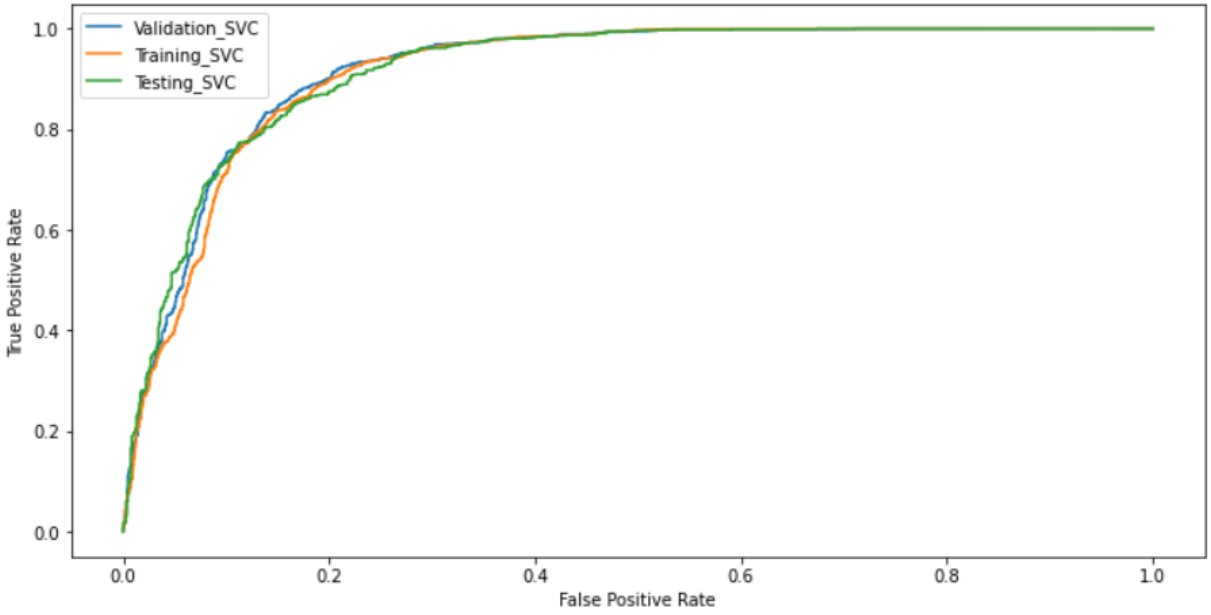
Learning curve analysis

Cost vs Number of training data



- The optimal value of cost is reached when we use only 800 training data.
- However, Cost keeps on decreasing as we increase training data
- Perform better than Logistic Regression

ROC curve

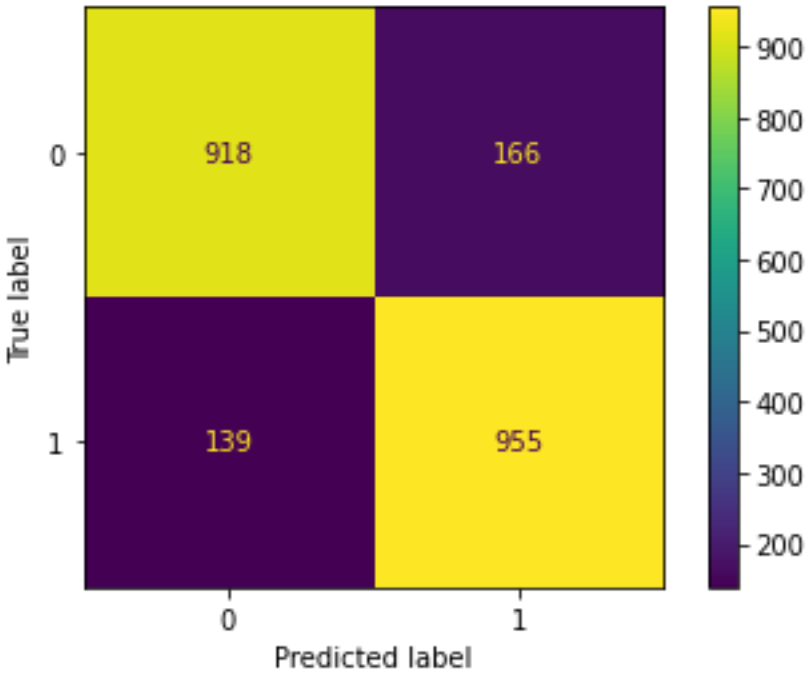


- AUC score of 0.873 is reached on test data
- Performs better than Logistic Regression

Classification and Confusion Matrix

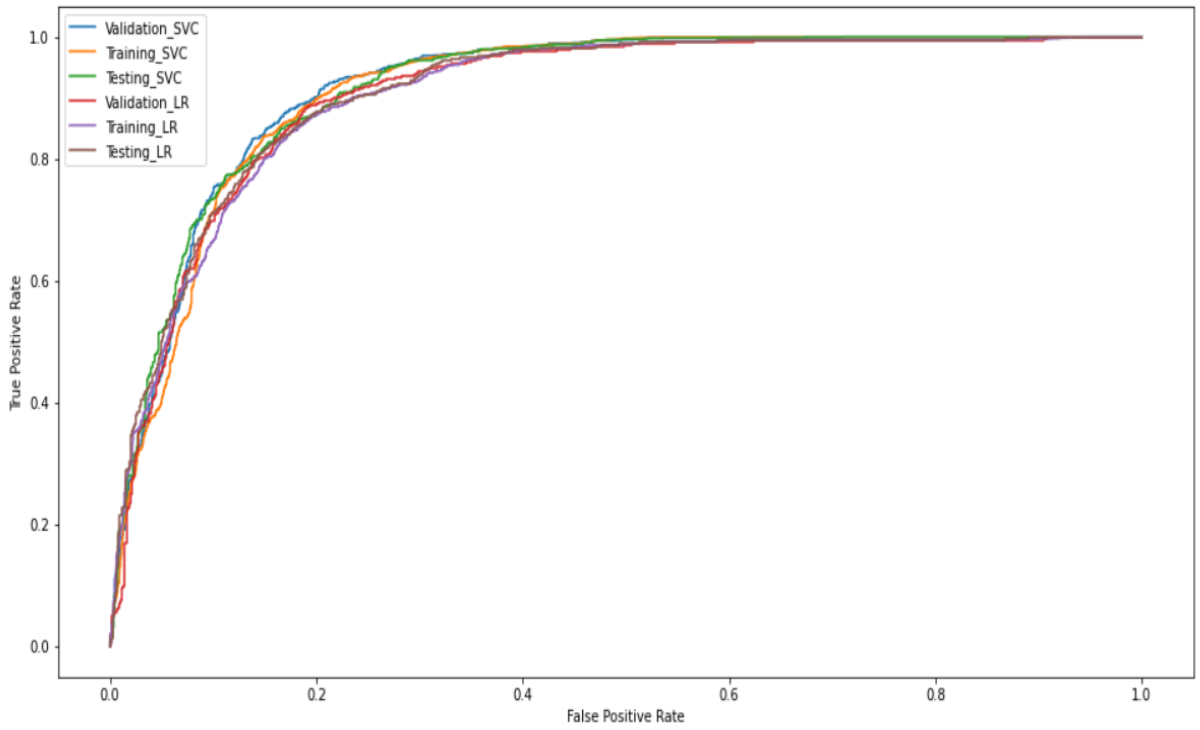
	precision	recall	f1-score	support
0	0.87	0.83	0.85	1084
1	0.84	0.88	0.86	1094
accuracy			0.86	2178
macro avg	0.86	0.86	0.86	2178
weighted avg	0.86	0.86	0.86	2178

- F1 score 86% is reached on test data
- Recall for prediction 0 is little less, maybe model is slightly biased towards prediction 1



- The model predicted 166 false positive.
- The model predicted 139 false negative.
- Model performs better for class 0.

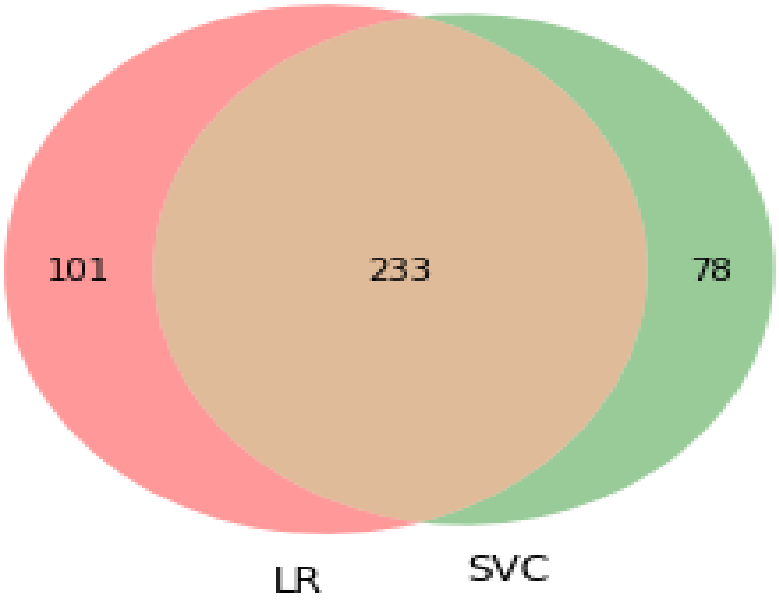
ROC Comparison



Both models perform somewhat similarly. However, SVC has an edge over Logistic Regression

Performance and error analysis

Venn diagram



- SVC performs better to some extent and misclassifies less test examples.
- 233 test data were wrongly classified by both models.

Comparison

Logistic Regression	Support Vector Classifier
Do need polynomial features	Do not need polynomial features
ROC score of 0.84	ROC score of 0.87
Wrongly classified 334 test data	Wrongly classified 311 test data
Test accuracy- 84%	Test accuracy- 86%
Lowest Validation Cost- 0.39	Lowest Validation Cost- 0.35

Misclassification

```
holiday                          0.000000
temp                             0.866751
humidity                        -0.421240
windspeed                        0.354895
Total_booking                  119.000000
Book                             0.000000
Hour                             8.000000
Month                            8.000000
 Clear + Few clouds              1.000000
 Heavy Rain + Thunderstorm       0.000000
 Light Snow, Light Rain          0.000000
 Mist + Cloudy                   0.000000
Fall                             1.000000
Spring                           0.000000
Summer                           0.000000
Winter                           0.000000
Friday                           0.000000
Monday                           0.000000
Saturday                         0.000000
Sunday                           1.000000
Thursday                         0.000000
Tuesday                          0.000000
Wednesday                        0.000000
Name: 33, dtype: float64
```

Most of the correct predictions for booking 0 contain temp around -0.45 (scaled) . For 75% of correct predictions temp is below 0.23.

Humidity for correct predictions is positive and around 0.40.

	temp	humidity	windspeed	Hour	Month	Clear + Few clouds	Heavy Rain + Thunderstorm	Light Snow, Light Rain	Mist + Cloudy	Fall
count	903.000000	903.000000	903.000000	903.000000	903.000000	903.000000	903.0	903.000000	903.000000	903.000000
mean	-0.468487	0.403868	-0.160288	7.354374	5.700997	0.634551	0.0	0.106312	0.259136	0.184939
std	0.903000	0.984002	1.025150	7.661535	3.737695	0.481823	0.0	0.308408	0.438403	0.388463
min	-2.177997	-3.262618	-1.647701	0.000000	1.000000	0.000000	0.0	0.000000	0.000000	0.000000
25%	-1.233075	-0.368622	-0.846305	2.000000	2.000000	0.000000	0.0	0.000000	0.000000	0.000000
50%	-0.603127	0.525886	-0.179071	4.000000	5.000000	1.000000	0.0	0.000000	0.000000	0.000000
75%	0.236803	1.315157	0.354895	11.000000	9.000000	1.000000	0.0	0.000000	1.000000	0.000000
max	1.706681	1.999193	4.225923	23.000000	12.000000	1.000000	0.0	1.000000	1.000000	1.000000

Correct Predictions