BIKE SHARING DEMAND PREDICTION USING MACHINE LEARNING - *KANNA LOKESH*

DATE - 1/05/2023

1. Problem Statement:

The Bike Sharing Demand Prediction is to predict the number of bikes that will be rented in a given time period from a bike sharing system. This problem is important for bike sharing companies to optimize their operations and provide better services to their customers.

The demand for bike rentals can be effected by various factors such as weather conditions, time of day, day of the week, season of the year, holidays, special events and other factors.

Therefore the goal of bike sharing demand prediction is to build a machine learning model that can accurately predict the demand for bike rentals based on these factors.

2. Market/Customer/Business Need Assessment:

Bike Sharing Demand Prediction can help bike sharing companies optimize their operation by predicting the demand and ensuring that bikes are available at the right locations at the right times.

By optimizing the operations, bike sharing companies can increase efficiency and profitability, which can benefit the market by providing a more sustainable and cost-effective transportation option.

For customers, the bike sharing demand prediction is more convenient for customers, as they can find bikes more easily and avoid situations where they arrive at a bike station and find that no bikes are available.

By providing better Services, bike sharing demand prediction can increase revenue for businesses by attracting more frequent bike rentals.

3. Target Specifications and Characterization:

The most common target specification for bike sharing demand prediction is the number of bike rentals in a given time.

The popular locations also can be targeted, because the most of the people needs rental bikes their.

The characteristics of the rental bike prediction is depends on the needs of the bike sharing company and the goals of the projects.

4. External Search (information sources/references):

The [Dataset](https://www.kaggle.com/competitions/bike-sharing-demand/data) can be found at kaggle. Here provided hourly rental data spanning two years. For this Project, the training set is comprised of the first 19 days of each month, while the test set is the 20th to the end of the month. We must predict the total count of bikes rented during each hour covered by the test set, using only information available prior to the rental period.

Dataset Origin:

https://www.kaggle.com/competitions/bike-sharing-demand/data

Let's view our dataset:

```
In [83]: data.sample(20)
```

Out[83]:

	id	year	hour	season	holiday	workingday	weather	temp	atemp	humidity	windspeed	count
4041	5740	2012	19	3	0	1	2	25.42	29.545	73	7.0015	486
7286	10329	2012	6	1	0	1	1	8.20	9.090	75	19.0012	76
7239	10265	2011	0	4	0	0	1	18.86	22.725	44	8.9981	107
6450	9159	2011	14	2	0	0	3	13.12	15.150	76	22.0028	190
5404	7673	2011	10	1	0	1	2	8.20	10.605	51	8.9981	51
2704	3857	2011	11	1	0	1	1	9.02	11.365	51	11.0014	64
2254	3247	2011	3	2	0	1	1	22.14	25.760	49	15.0013	5
7124	10103	2011	12	1	0	1	1	8.20	9.850	44	16.9979	61
6345	8996	2011	10	2	0	0	2	14.76	17.425	57	11.0014	154
2905	4130	2011	15	3	0	0	1	31.16	34.090	45	22.0028	428
6612	9381	2012	2	4	0	1	1	11.48	15.150	65	6.0032	3
34	51	2011	8	1	0	0	1	5.74	7.575	86	8.9981	30
1763	2535	2011	6	4	0	1	1	9.84	10.605	65	19.0012	97
3358	4794	2011	21	2	0	1	1	16.40	20.455	62	11.0014	150
5710	8111	2012	4	2	0	1	1	26.24	29.545	78	8.9981	7
2273	3271	2012	7	2	0	0	1	20.50	24.240	68	7.0015	35
4849	6881	2012	12	2	0	1	1	21.32	25.000	25	0.0000	331
4773	6779	2012	5	3	0	1	2	26.24	28.790	89	22.0028	48
1611	2314	2011	14	4	0	0	1	24.60	31.060	40	19.9995	462
3639	5190	2012	0	3	0	1	1	27.88	12.120	57	11.0014	88

See some information about our dataset:

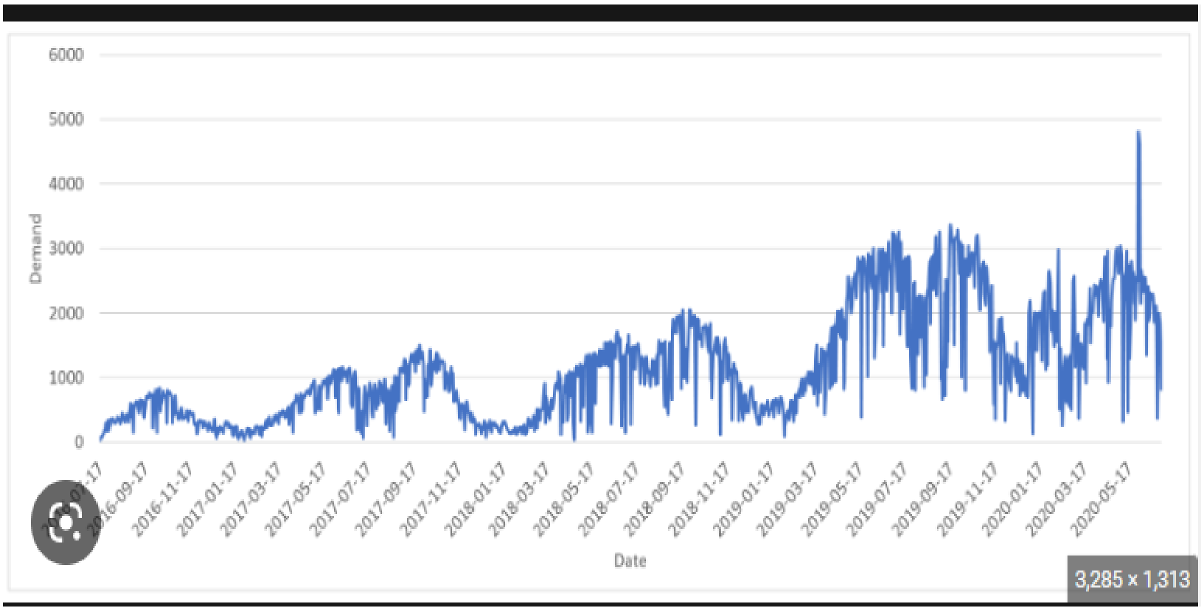
```
In [86]: data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7689 entries, 0 to 7688
Data columns (total 11 columns):
year          7689 non-null int64
hour          7689 non-null int64
season        7689 non-null int64
holiday       7689 non-null int64
workingday    7689 non-null int64
weather       7689 non-null int64
temp          7689 non-null float64
atemp         7689 non-null float64
humidity      7689 non-null int64
windspeed     7689 non-null float64
count         7689 non-null int64
dtypes: float64(3), int64(8)
memory usage: 660.9 KB
```

5. Benchmarking:

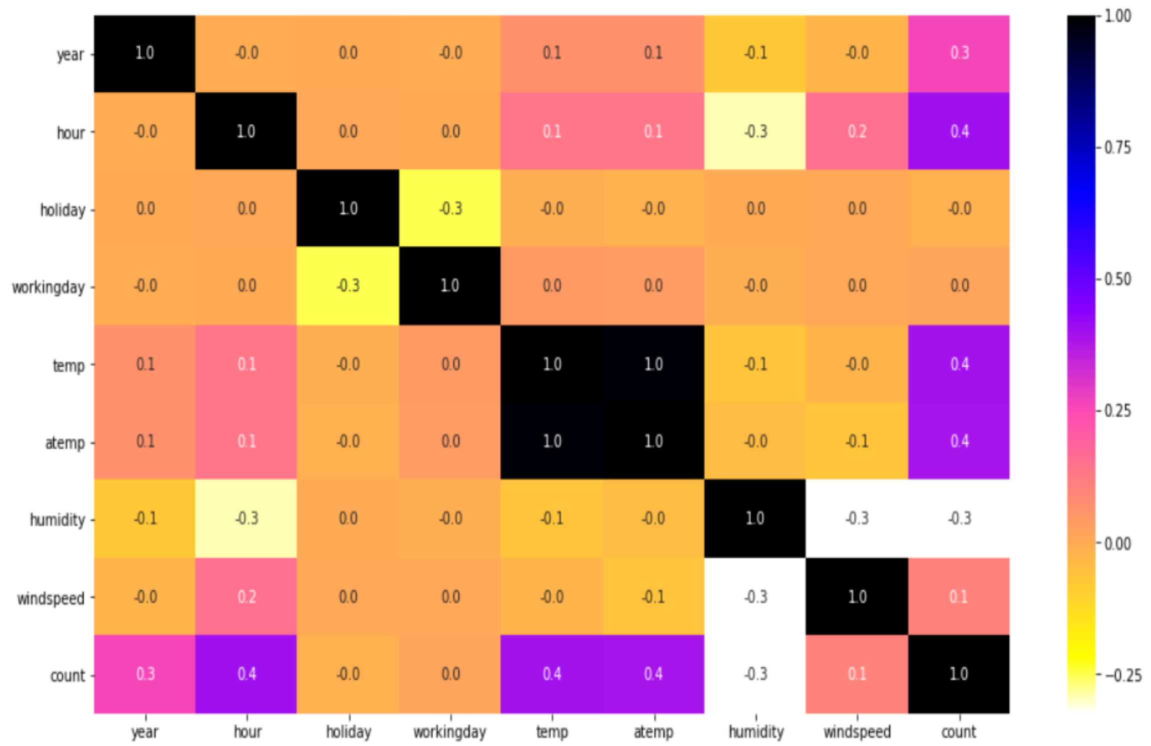
Benchmarking in Bike Demand Prediction involves comparing the performance of the machine learning model to the performance of other models or industry standards.

Bike sharing companies often have their own industry standards for bike demand prediction. These standards may be based on historical data, user surveys, or other factors.



Correlation Matrix For Rental Bikes Data

```
plt.figure(figsize= (16, 8))
sns.heatmap(data.corr(), annot = True, cmap= 'gnuplot2_r', fmt= '.1f');
```



In this, above fig., we find the same Correlation of all the columns. I use the matplotlib to resize the output of the image and using seaborn heatmap find a correlation between each of the columns.

7. Applicable Patents:

- Machine learning algorithms: The machine learning algorithms used for bike demand prediction may be subject to patents. For example, deep learning techniques, such as artificial nueral networks, may be patented.
- Data processing frameworks: Data processing frameworks used for bike demand prediction, such as Apache Spark or Hadoop, may have patents related to their use and implementation.
- Data visualization tools: Data visualization tools used to display bike demand predictions may be subject to patents related to their design and functionality.

8.Applicable Regulations:

- Bike demand prediction models may be subject to environmental regulations that promote the use of low-carbon transportation options. For example, regulations promoting the use of electric bikes or limiting carbon emissions from transportation may affect the operations and demand for bike sharing services

- Bike sharing services may be subject to traffic regulations, such as requirements for bike lanes and traffic signals, that may impact the demand for bike sharing services.

- Many cities and countries have regulations specific to bike sharing operations, such as permit requirements, insurance, and safety standards. These regulations may impact the operations and implementation of bike demand prediction models.

9.Applicable Constraints:

- The quality and quantity of historical data available for bike demand prediction may be limited .
- Bike demand prediction models may be limited in their geographic scope. Factors such as population density, traffic patterns, and bike infrastructure can vary widely by region and can affect the accuracy of the models.
- Legal constraints, such as regulations around bike sharing operations, may impact the implementation and operation of bike demand prediction models.

10. Business Model:

- Bike demand prediction models could be integrated into existing bike sharing platforms, and revenue could be shared based on the number of rentals or other metrics.
- Bike demand prediction models can be offered as a subscription-based service to bike sharing companies or other businesses in the transportation industry. Subscribers could pay a monthly or yearly fee for access to the prediction models and related services.
- Bike demand prediction models can provide valuable insights into user behavior and demand patterns. One potential monetization strategy is to sell these insights to bike sharing companies that can use the information to optimize their operations and improve user experience.

11. Concept Generation :

The concept for the bike demand prediction product/service involves using machine learning algorithms to predict the demand for bike sharing services in specific locations at different times of the day. The model will analyze various factors, such as weather conditions, day of the week, and local events, to provide accurate predictions of demand. The output of the prediction model can be used to optimize bike sharing operations, such as determining where to deploy bikes, when to schedule maintenance, and when to offer promotional discounts.

The product/service could be offered as a subscription-based service to bike sharing companies, transportation providers, or urban planners. Alternatively, the insights generated by the prediction model could be sold to these organizations for a fee. The product/service could also be integrated into existing bike sharing platforms, providing a value-added feature for users and generating revenue through commission-based revenue sharing.

12. Concept Development:

12.1. First we clean the data

12.2. Split the data in x, y variable

```
x=data.drop('count',axis=1)
y=data[['count']]
```

```
x.shape
```

(7685, 10)

```
x.head()
```

	year	hour	season	holiday	workingday	weather	temp	atemp	humidity	windspeed
0	2012	23	3	0	0	2	23.78	27.275	73	11.0014
1	2011	8	3	0	0	1	27.88	31.820	57	0.0000
2	2012	2	1	0	1	1	20.50	24.240	59	0.0000
3	2011	20	3	0	1	3	25.42	28.790	83	19.9995
4	2011	17	3	0	1	3	26.24	28.790	89	0.0000

```
y.shape
```

(7685, 1)

```
y.head()
```

	count
0	133
1	132
2	19
3	58
4	285

12.3. Train_Test_Split the Data in x_train, x_test, y_train, y_test

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=42)
```

```
x_train.shape
```

```
(6148, 10)
```

```
x_train.head()
```

	year	hour	season	holiday	workingday	weather	temp	atemp	humidity	windspeed
3204	2012	23	4	0	0	2	16.40	20.455	94	0.0000
1536	2011	23	1	0	1	1	4.92	5.305	33	22.0028
3668	2011	20	4	0	1	1	21.32	25.000	59	22.0028
6204	2012	12	2	0	1	1	28.70	32.575	65	6.0032
501	2011	6	4	0	0	1	15.58	19.695	66	11.0014

```
y_train.shape
```

```
(6148, 1)
```

```
x_test.shape
```

```
(1537, 10)
```

```
x_test.head()
```

	year	hour	season	holiday	workingday	weather	temp	atemp	humidity	windspeed
2388	2012	18	1	0	1	2	25.42	31.060	38	36.9974
6534	2012	7	2	0	1	3	18.04	21.970	94	7.0015
6371	2011	19	3	0	1	3	25.42	25.760	100	22.0028

We will use three different models and we will finalize the model which will give good accuracy

1. RandomForestRegressor:

RandomForestRegressor

```
from sklearn.ensemble import RandomForestRegressor
model=RandomForestRegressor(n_estimators=5)
```

```
model.fit(x_train,y_train)
```

```
C:\Users\k.vidya\Anaconda3\lib\site-packages\ipykernel_launcher.py:1: DataConversionWarning: A column-vector y was passed when a
1d array was expected. Please change the shape of y to (n_samples,), for example using ravel().
  """Entry point for launching an IPython kernel.
```

```
RandomForestRegressor(bootstrap=True, criterion='mse', max_depth=None,
           max_features='auto', max_leaf_nodes=None,
           min_impurity_decrease=0.0, min_impurity_split=None,
           min_samples_leaf=1, min_samples_split=2,
           min_weight_fraction_leaf=0.0, n_estimators=5, n_jobs=1,
           oob_score=False, random_state=None, verbose=0, warm_start=False)
```

```
y_pred=model.predict(x_test)
```

```
from sklearn.metrics import r2_score,mean_squared_error,mean_absolute_error
accuracy=r2_score(y_test,y_pred)
```

```
print(f"The accuracy of the model is {accuracy}")
```

```
The accuracy of the model is 0.9095008256717573
```

2.LinearRegression:

LinearRegression

```
]: from sklearn.preprocessing import StandardScaler
   st=StandardScaler()
   x_train=st.fit_transform(x_train)
   x_test=st.transform(x_test)
```

```
]: from sklearn.linear_model import LinearRegression
   model2=LinearRegression()
   model2.fit(x_train,y_train)
```

```
]: LinearRegression(copy_X=True, fit_intercept=True, n_jobs=1, normalize=False)
```

```
]: y_pred2=model2.predict(x_test)
```

```
]: acc2=r2_score(y_test,y_pred2)
   print(acc2)

   0.3780463136091806
```

```
]:
```

3. DecisionTreeRegressor:

DecisionTreeRegressor ¶

```
: from sklearn.tree import DecisionTreeRegressor
  model1=DecisionTreeRegressor()
```

```
: model1.fit(x_train,y_train)
```

```
: DecisionTreeRegressor(criterion='mse', max_depth=None, max_features=None,
             max_leaf_nodes=None, min_impurity_decrease=0.0,
             min_impurity_split=None, min_samples_leaf=1,
             min_samples_split=2, min_weight_fraction_leaf=0.0,
             presort=False, random_state=None, splitter='best')
```

```
: y_pred1=model1.predict(x_test)
```

```
: acc1=r2_score(y_test,y_pred1)
```

```
: print(acc1)

  0.8642352988128148
```

By analyzing all the models, we can say that Random Forest is giving good results.

1. RandomForestRegressor:
Accuracy-------→ 90.95

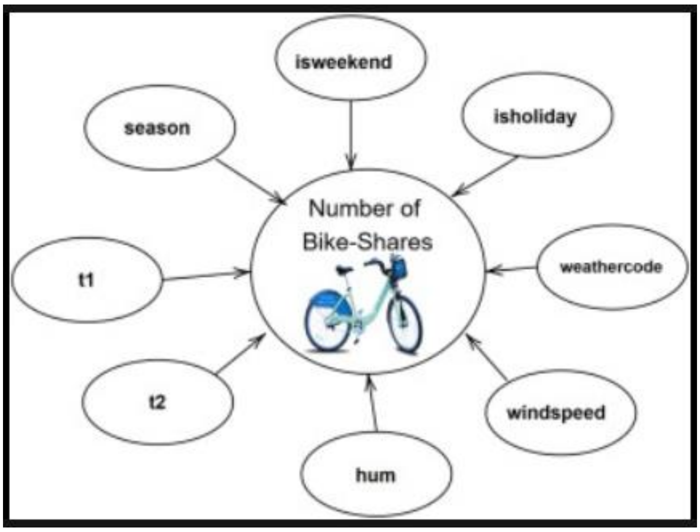
2. LinearRegression:
Accuracy------→37.80

3. DecisionTreeRegressor:
Accuracy-----→86.42

So We Finalize RandomForestRegressor (n_estimators=5) Model For Our Model Training And Model Deployment.

13. Final Product Prototype (abstract) with Schematic Diagram:

Machine learning model that uses historical data and real-time information to predict the demand for bike sharing services in specific locations at different times of the day. The model will take into account various factors such as weather conditions, day of the week, local events, and user behavior patterns to provide accurate predictions of demand.



The prediction model will be trained using a dataset of historical bike rental data, including rental times, locations, and user behavior patterns. The model will be validated using real-time data collected from sensors and other sources, such as weather forecasts and event calendars.

The output of the prediction model will be provided to bike sharing operators and other transportation providers through a dashboard or API, allowing them to optimize their operations in real-time. The dashboard will include visualizations of predicted demand and provide suggestions for optimal bike deployment, maintenance scheduling, and promotional discounts.

14. Product details:

- **How does it work?**

Bike sharing demand prediction works by using machine learning algorithms to analyze various factors that affect the demand for bike sharing services, such as weather conditions, day of the week, time of day, and local events. The prediction model analyzes historical data on bike rentals and user behavior patterns to identify trends and patterns that can be used to make predictions about future demand.

- Data Sources

The [Dataset](https://www.kaggle.com/competitions/bike-sharing-demand/data) can be found at kaggle. Here provided hourly rental data spanning two years. For this Project, the training set is comprised of the first 19 days of each month, while the test set is the 20th to the end of the month. We must predict the total count of bikes rented during each hour covered by the test set, using only information available prior to the rental period.

Dataset Origin:

https://www.kaggle.com/competitions/bike-sharing-demand/data

- Algorithms, frameworks, software etc. needed

For the above Business Case study, we came with RandomForestRegressor Algorithm and the software we are used to develop the model is Python.

15. Code Implementation/Validation on Small Scale:

- Some Basic Visualization on Real World or Augmented Data

Data:

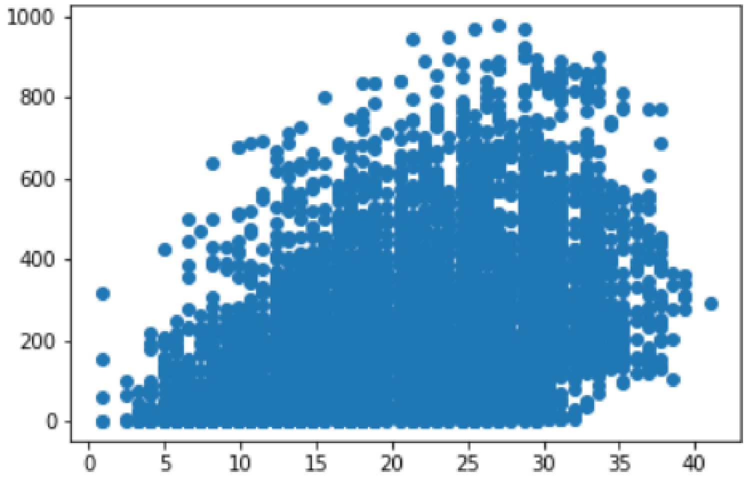
In [83]:
```
data.sample(20)
```

Out[83]:

	id	year	hour	season	holiday	workingday	weather	temp	atemp	humidity	windspeed	count
4041	5740	2012	19	3	0	1	2	25.42	29.545	73	7.0015	486
7286	10329	2012	6	1	0	1	1	8.20	9.090	75	19.0012	76
7239	10265	2011	0	4	0	0	1	18.86	22.725	44	8.9981	107
6450	9159	2011	14	2	0	0	3	13.12	15.150	76	22.0028	190
5404	7673	2011	10	1	0	1	2	8.20	10.605	51	8.9981	51
2704	3857	2011	11	1	0	1	1	9.02	11.365	51	11.0014	64
2254	3247	2011	3	2	0	1	1	22.14	25.760	49	15.0013	5
7124	10103	2011	12	1	0	1	1	8.20	9.850	44	16.9979	61
6345	8996	2011	10	2	0	0	2	14.76	17.425	57	11.0014	154
2905	4130	2011	15	3	0	0	1	31.16	34.090	45	22.0028	428
6612	9381	2012	2	4	0	1	1	11.48	15.150	65	6.0032	3
34	51	2011	8	1	0	0	1	5.74	7.575	86	8.9981	30
1763	2535	2011	6	4	0	1	1	9.84	10.605	65	19.0012	97
3358	4794	2011	21	2	0	1	1	16.40	20.455	62	11.0014	150
5710	8111	2012	4	2	0	1	1	26.24	29.545	78	8.9981	7
2273	3271	2012	7	2	0	0	1	20.50	24.240	68	7.0015	35
4849	6881	2012	12	2	0	1	1	21.32	25.000	25	0.0000	331
4773	6779	2012	5	3	0	1	2	26.24	28.790	89	22.0028	48
1611	2314	2011	14	4	0	0	1	24.60	31.060	40	19.9995	462
3639	5190	2012	0	3	0	1	1	27.88	12.120	57	11.0014	88

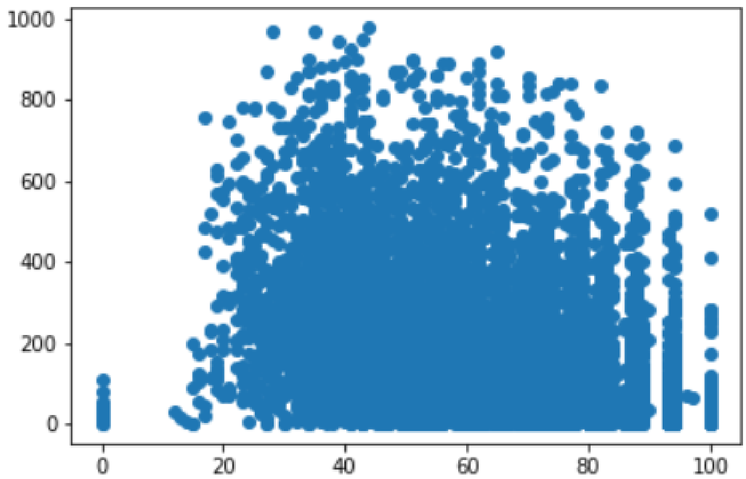
Temperature Vs Counts:

```
plt.scatter(x=data['temp'],y=data['count'])
plt.show()
```



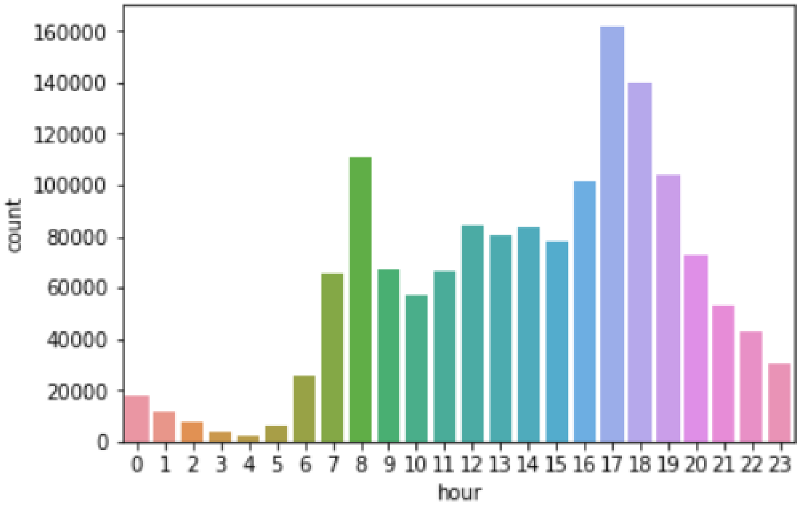
Humidity Vs Counts:

```
plt.scatter(x=data['humidity'],y=data['count'])
plt.show()
```



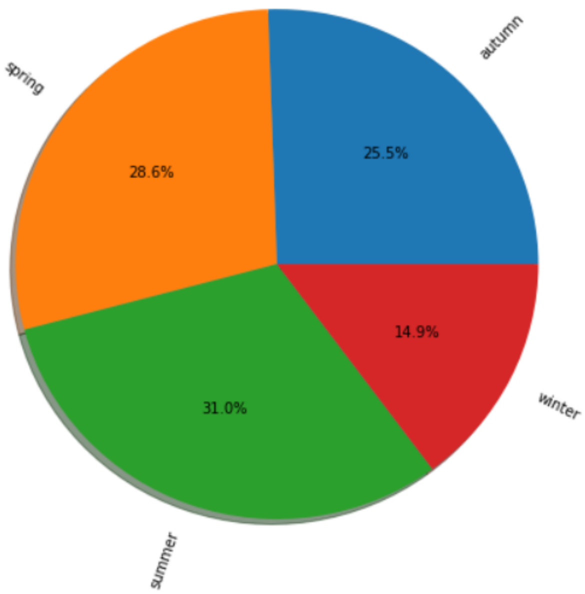
- Simple EDA

```
sns.barplot(x=dt_month['hour'],y=dt_month['count'])
plt.show()
```



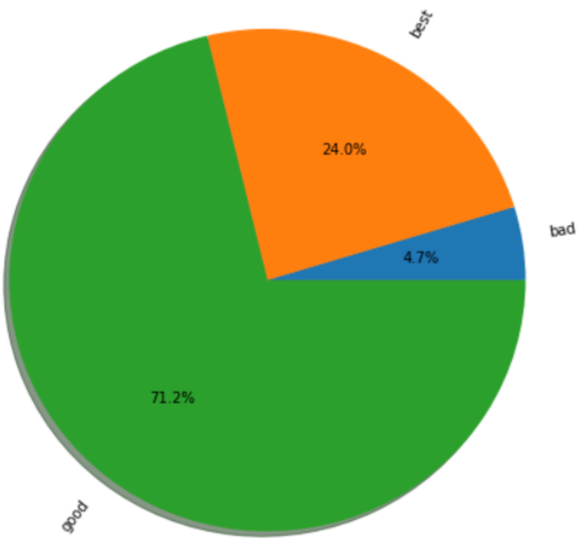
The above picture shows in the 17 hour most of the bikes were rented.

```
plt.figure(figsize = (8,8))
plt.pie(dt_season['count'],labels=dt_season['season'],rotatelabels=True, autopct='%1.1f%%',shadow=True)
plt.show()
```



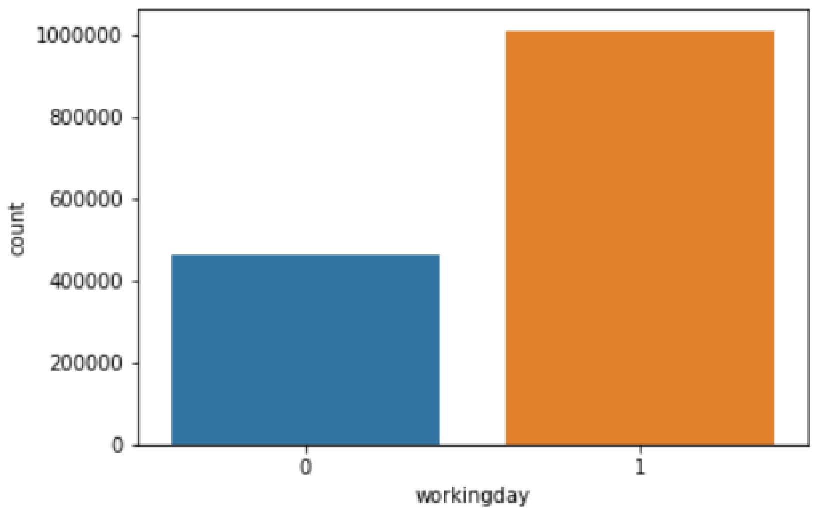
The above picture shows that in the summer season the most of the bikes were rented.

```
plt.figure(figsize = (8,8))
plt.pie(dt_weather['count'],labels=dt_weather['weather'],rotatelabels=True, autopct='%1.1f%%',shadow=True)
plt.show()
```



The above picture shows that the most of the bikes were rented during the good weather conditions only.

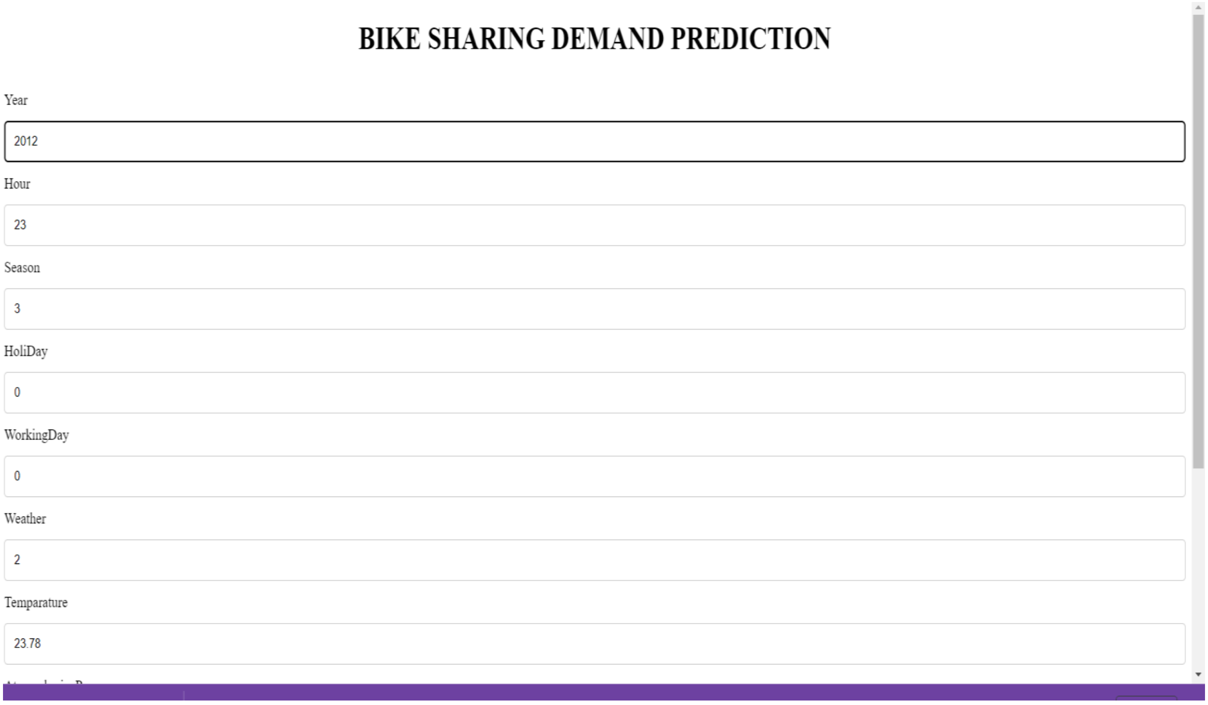
```
sns.barplot(x=dt_day['workingday'],y=dt_day['count'])
plt.show()
```

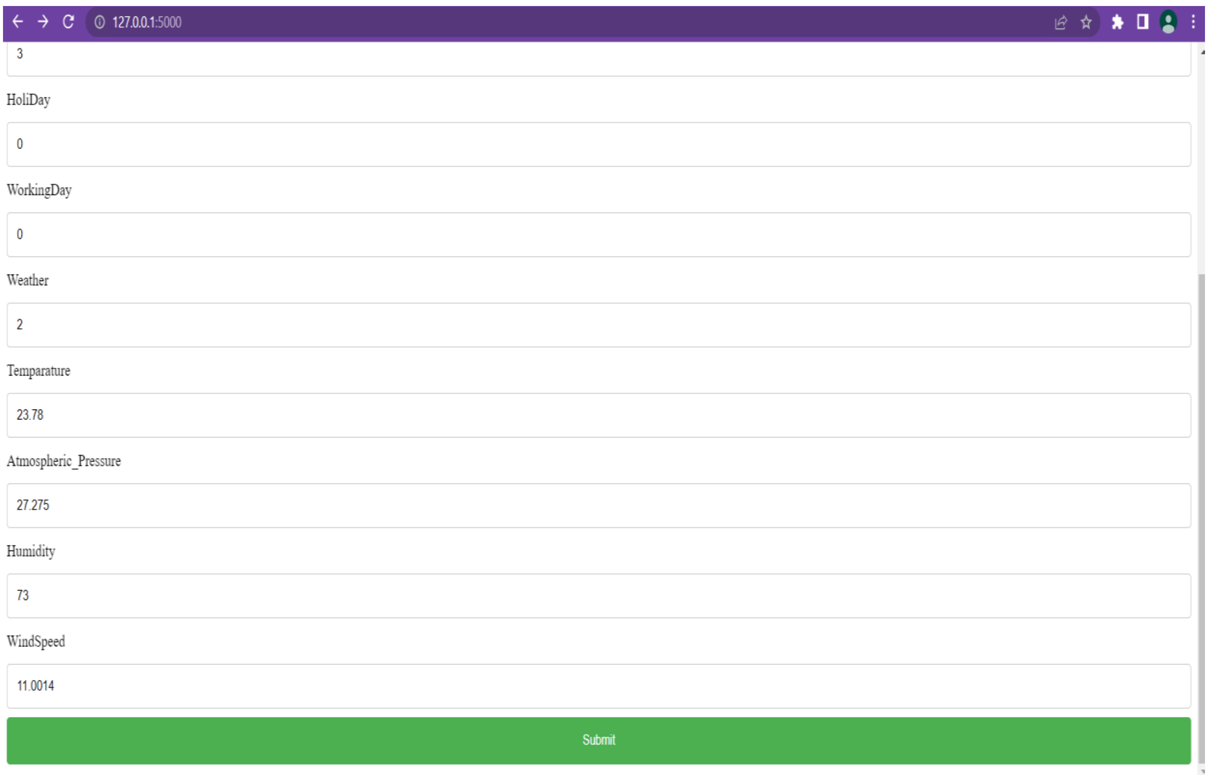


The above picture states that the most of the bikes were rented during the working days not on holidays.

- ML Modelling:

The modeling shows how the machine learning model works on backend when user provides the input on frontend. It describes the prediction of output for new input data.





Output:

The Number Of Bikes To Provide Is 65

- Github link to the code implementation:

This is a github link :-

https://github.com/kannalokesh13/bike_share_demand_pred_prototype

16. Conclusion:

Bike sharing demand prediction is a valuable application of machine learning in the transportation industry. By using historical data and real-time information, bike sharing operators and other transportation providers can accurately predict demand for bike sharing services in specific locations at different times of the day.This prediction model can help optimize bike sharing operations by providing insights into where to deploy bikes, when to schedule maintenance, and when to offer promotional discounts. In addition, bike sharing demand prediction can also improve the user experience by ensuring bikes are available when and where they are needed.