Bike Sharing : Multiple Linear Regression

In [1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
from sklearn.linear_model import LinearRegression
from sklearn.feature_selection import RFE
from statsmodels.stats.outliers_influence import variance_inflation_factor
import statsmodels.api as sm
from sklearn.model_selection import train_test_split
from sklearn.metrics import r2_score
from sklearn.preprocessing import MinMaxScaler
import matplotlib.pyplot as plt
%matplotlib inline
import warnings
warnings.filterwarnings('ignore')
```

In [2]:
```
df=pd.read_csv('day.csv')
df.head()
```

Out[2]:

	instant	dteday	season	yr	mnth	holiday	weekday	workingday	weathersit	temp	atemp	hum	windspeed	casual	registered	cnt
0	1	01-01-2018	1	0	1	0	1	1	2	14.110847	18.18125	80.5833	10.749882	331	654	985
1	2	02-01-2018	1	0	1	0	2	1	2	14.902598	17.68695	69.6087	16.652113	131	670	801
2	3	03-01-2018	1	0	1	0	3	1	1	8.050924	9.47025	43.7273	16.636703	120	1229	1349
3	4	04-01-2018	1	0	1	0	4	1	1	8.200000	10.60610	59.0435	10.739832	108	1454	1562
4	5	05-01-2018	1	0	1	0	5	1	1	9.305237	11.46350	43.6957	12.522300	82	1518	1600

In [3]:
```
df.shape
```

Out[3]:
```
(730, 16)
```

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 730 entries, 0 to 729
Data columns (total 16 columns):
 #   Column      Non-Null Count  Dtype  
---  ------      --------------  -----  
 0   instant     730 non-null    int64  
 1   dteday      730 non-null    object 
 2   season      730 non-null    int64  
 3   yr          730 non-null    int64  
 4   mnth        730 non-null    int64  
 5   holiday     730 non-null    int64  
 6   weekday     730 non-null    int64  
 7   workingday  730 non-null    int64  
 8   weathersit  730 non-null    int64  
 9   temp        730 non-null    float64
 10  atemp       730 non-null    float64
 11  hum         730 non-null    float64
 12  windspeed   730 non-null    float64
 13  casual      730 non-null    int64  
 14  registered  730 non-null    int64  
 15  cnt         730 non-null    int64  
dtypes: float64(4), int64(11), object(1)
memory usage: 91.4+ KB
```

In [5]:
```
df.describe()
```

Out[5]:

	instant	season	yr	mnth	holiday	weekday	workingday	weathersit	temp	atemp	hum	windspeed
count	730.000000	730.000000	730.000000	730.000000	730.000000	730.000000	730.000000	730.000000	730.000000	730.000000	730.000000	730.000000
mean	365.500000	2.498630	0.500000	6.526027	0.028767	2.995890	0.690411	1.394521	20.319259	23.726322	62.765175	12.763620
std	210.877136	1.110184	0.500343	3.450215	0.167266	2.000339	0.462641	0.544807	7.506729	8.150308	14.237589	5.195841
min	1.000000	1.000000	0.000000	1.000000	0.000000	0.000000	0.000000	1.000000	2.424346	3.953480	0.000000	1.500244
25%	183.250000	2.000000	0.000000	4.000000	0.000000	1.000000	0.000000	1.000000	13.811885	16.889713	52.000000	9.041650

	instant	season	yr	mnth	holiday	weekday	workingday	weathersit	temp	atemp	hum	windspeed	
50%	365.500000	3.000000	0.500000	7.000000	0.000000	3.000000	1.000000	1.000000	20.465826	24.368225	62.625000	12.125325	
75%	547.750000	3.000000	1.000000	10.000000	0.000000	5.000000	1.000000	2.000000	26.880615	30.445775	72.989575	15.625589	1
max	730.000000	4.000000	1.000000	12.000000	1.000000	6.000000	1.000000	3.000000	35.328347	42.044800	97.250000	34.000021	3

there are 16 columns but in that some columns have categorical values in integer form Ex. season, months, weekdays, etc.

In [6]:

```
#Check for NULL/MISSING values
df.isnull().sum()
```

Out[6]:

```
instant       0
dteday        0
season        0
yr            0
mnth          0
holiday       0
weekday       0
workingday    0
weathersit    0
temp          0
atemp         0
hum           0
windspeed     0
casual        0
registered    0
cnt           0
dtype: int64
```

There are no null and missing values

Duplicate Check

In [7]:

```
df_duplicate = df.copy()
```

In [8]:

```
df_duplicate.drop_duplicates(subset=None, inplace=True)
```

In [9]:
```
df_duplicate.head()
```

Out[9]:

	instant	dteday	season	yr	mnth	holiday	weekday	workingday	weathersit	temp	atemp	hum	windspeed	casual	registered	cnt
0	1	01-01-2018	1	0	1	0	1	1	2	14.110847	18.18125	80.5833	10.749882	331	654	985
1	2	02-01-2018	1	0	1	0	2	1	2	14.902598	17.68695	69.6087	16.652113	131	670	801
2	3	03-01-2018	1	0	1	0	3	1	1	8.050924	9.47025	43.7273	16.636703	120	1229	1349
3	4	04-01-2018	1	0	1	0	4	1	1	8.200000	10.60610	59.0435	10.739832	108	1454	1562
4	5	05-01-2018	1	0	1	0	5	1	1	9.305237	11.46350	43.6957	12.522300	82	1518	1600

In [10]:
```
df.shape
```

Out[10]:
```
(730, 16)
```

In [11]:
```
df_duplicate.shape
```

Out[11]:
```
(730, 16)
```

There are no duplicate values

Removing unwanted columns

In [12]:
```
df.columns
```

Out[12]:
```
Index(['instant', 'dteday', 'season', 'yr', 'mnth', 'holiday', 'weekday',
       'workingday', 'weathersit', 'temp', 'atemp', 'hum', 'windspeed',
       'casual', 'registered', 'cnt'],
      dtype='object')
```

In [13]:
```
new_df=df[['season', 'yr', 'mnth', 'holiday', 'weekday',
```

```
        'workingday', 'weathersit', 'temp', 'atemp', 'hum', 'windspeed',
        'cnt']]
```

Creating Dummy Variables

```
we need to create dummy variables for 4 categorical variables
first we will change it's data type to categorical
```

In [14]:
```
new_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 730 entries, 0 to 729
Data columns (total 12 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   season      730 non-null    int64
 1   yr          730 non-null    int64
 2   mnth        730 non-null    int64
 3   holiday     730 non-null    int64
 4   weekday     730 non-null    int64
 5   workingday  730 non-null    int64
 6   weathersit  730 non-null    int64
 7   temp        730 non-null    float64
 8   atemp       730 non-null    float64
 9   hum         730 non-null    float64
 10  windspeed   730 non-null    float64
 11  cnt         730 non-null    int64
dtypes: float64(4), int64(8)
memory usage: 68.6 KB
```

In [15]:
```
new_df['season']=new_df['season'].astype('category')
new_df['weathersit']=new_df['weathersit'].astype('category')
new_df['mnth']=new_df['mnth'].astype('category')
new_df['weekday']=new_df['weekday'].astype('category')
```

In [16]:
```
new_df.season.replace({1:"spring", 2:"summer", 3:"fall", 4:"winter"},inplace = True)

new_df.weathersit.replace({1:'good',2:'moderate',3:'bad',4:'severe'},inplace = True)
```

```python
new_df.mnth = new_df.mnth.replace({1: 'jan',2: 'feb',3: 'mar',4: 'apr',5: 'may',6: 'jun',
                  7: 'jul',8: 'aug',9: 'sept',10: 'oct',11: 'nov',12: 'dec'})

new_df.weekday = new_df.weekday.replace({0: 'sun',1: 'mon',2: 'tue',3: 'wed',4: 'thu',5: 'fri',6: 'sat'})
new_df.head()
```

Out[16]:

	season	yr	mnth	holiday	weekday	workingday	weathersit	temp	atemp	hum	windspeed	cnt
0	spring	0	jan	0	mon	1	moderate	14.110847	18.18125	80.5833	10.749882	985
1	spring	0	jan	0	tue	1	moderate	14.902598	17.68695	69.6087	16.652113	801
2	spring	0	jan	0	wed	1	good	8.050924	9.47025	43.7273	16.636703	1349
3	spring	0	jan	0	thu	1	good	8.200000	10.60610	59.0435	10.739832	1562
4	spring	0	jan	0	fri	1	good	9.305237	11.46350	43.6957	12.522300	1600

In [17]:

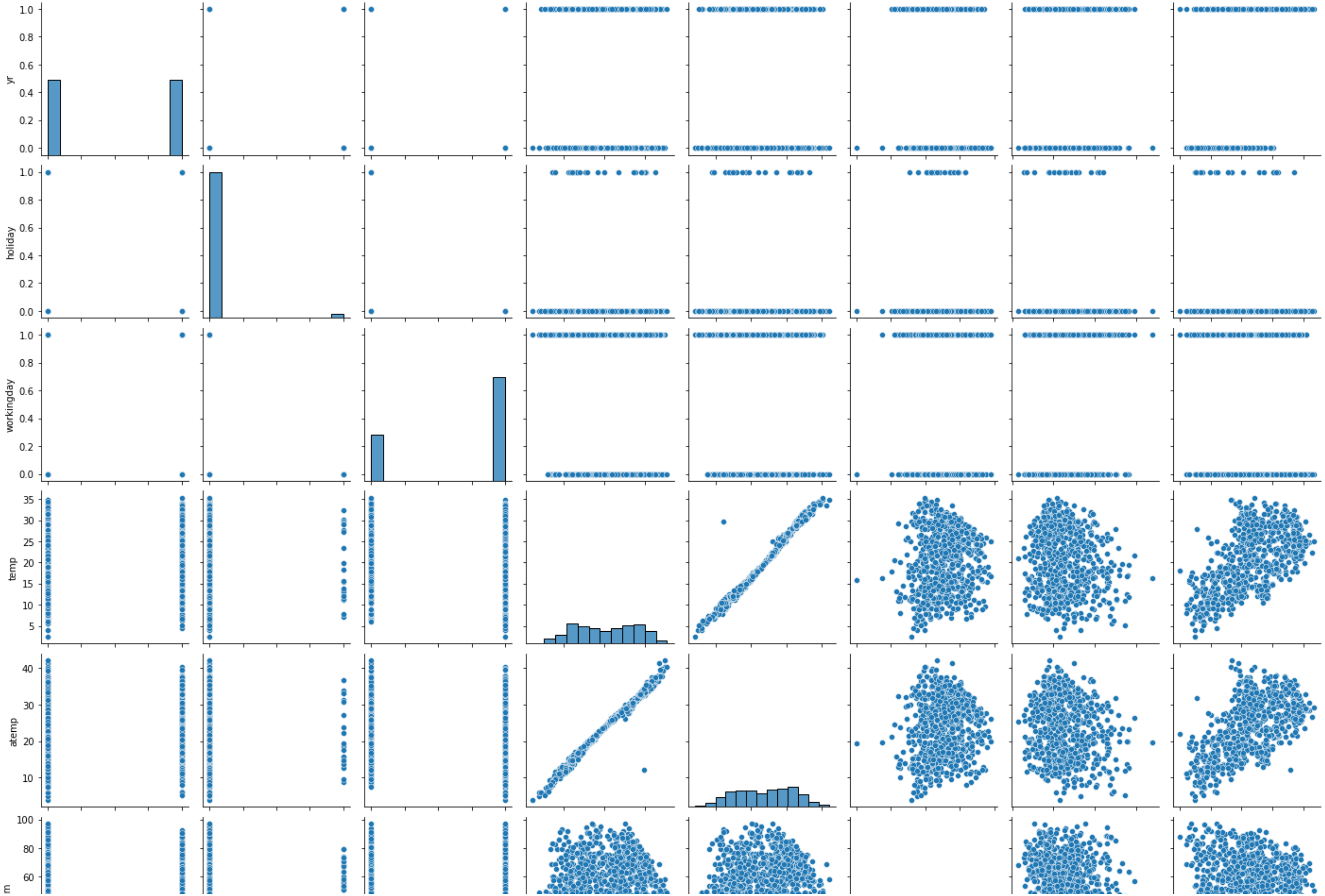
```python
new_df.info()
```

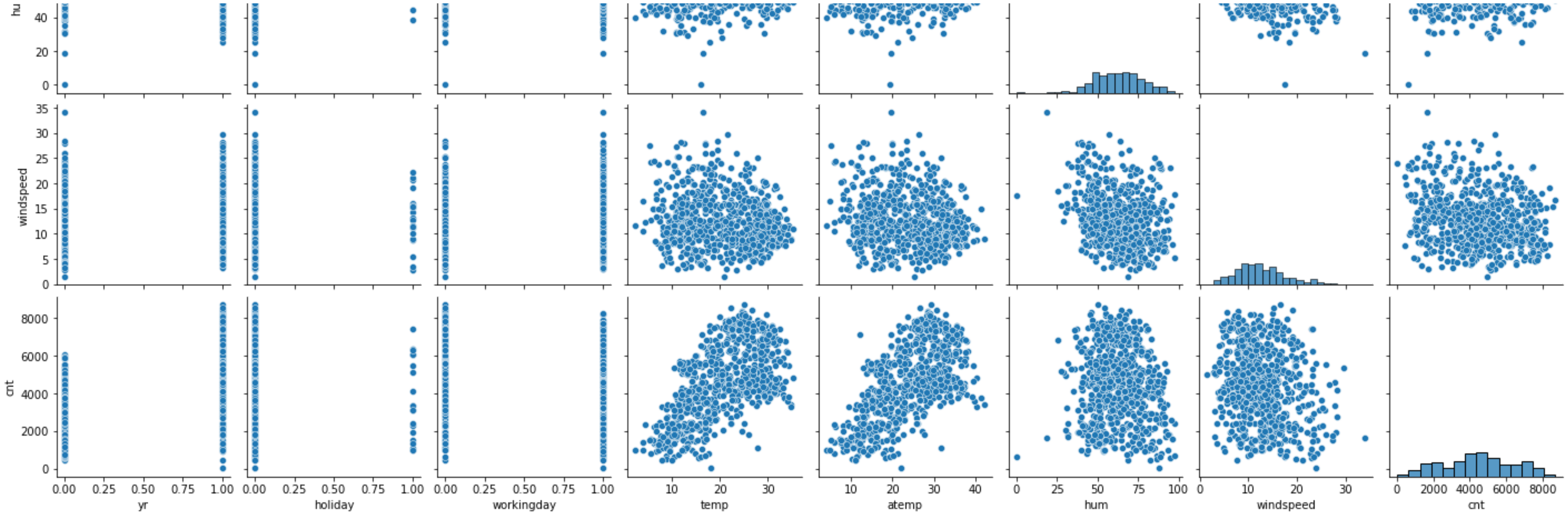
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 730 entries, 0 to 729
Data columns (total 12 columns):
 #   Column      Non-Null Count  Dtype  
---  ------      --------------  -----  
 0   season      730 non-null    object 
 1   yr          730 non-null    int64  
 2   mnth        730 non-null    object 
 3   holiday     730 non-null    int64  
 4   weekday     730 non-null    object 
 5   workingday  730 non-null    int64  
 6   weathersit  730 non-null    object 
 7   temp        730 non-null    float64
 8   atemp       730 non-null    float64
 9   hum         730 non-null    float64
 10  windspeed   730 non-null    float64
 11  cnt         730 non-null    int64  
dtypes: float64(4), int64(4), object(4)
memory usage: 68.6+ KB
```

EDA

In [18]:
```
plt.figure(figsize=(20,15))
sns.pairplot(new_df)
plt.show()
```

<Figure size 1440x1080 with 0 Axes>

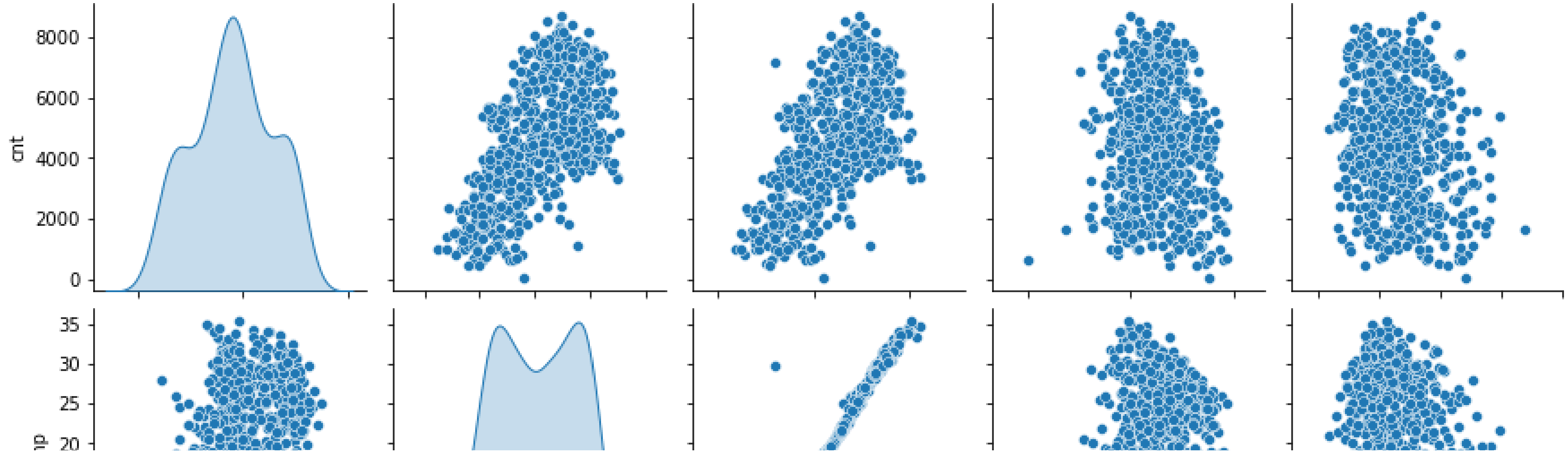


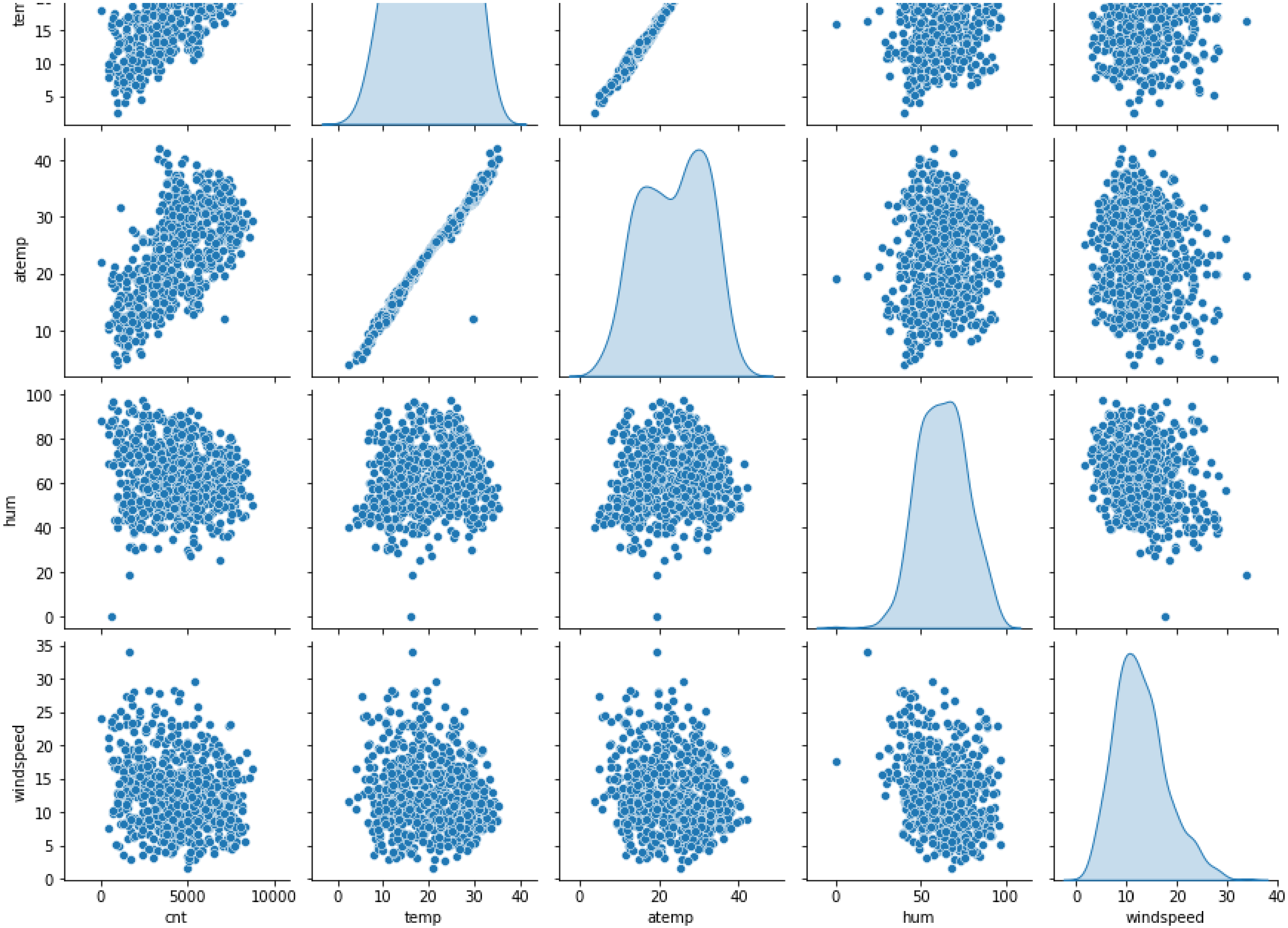


Visualising Numeric Variables

In [19]:
```
plt.figure(figsize = (30,30))
sns.pairplot(data=new_df,vars=['cnt', 'temp', 'atemp', 'hum','windspeed'],diag_kind='kde')
plt.show()
```

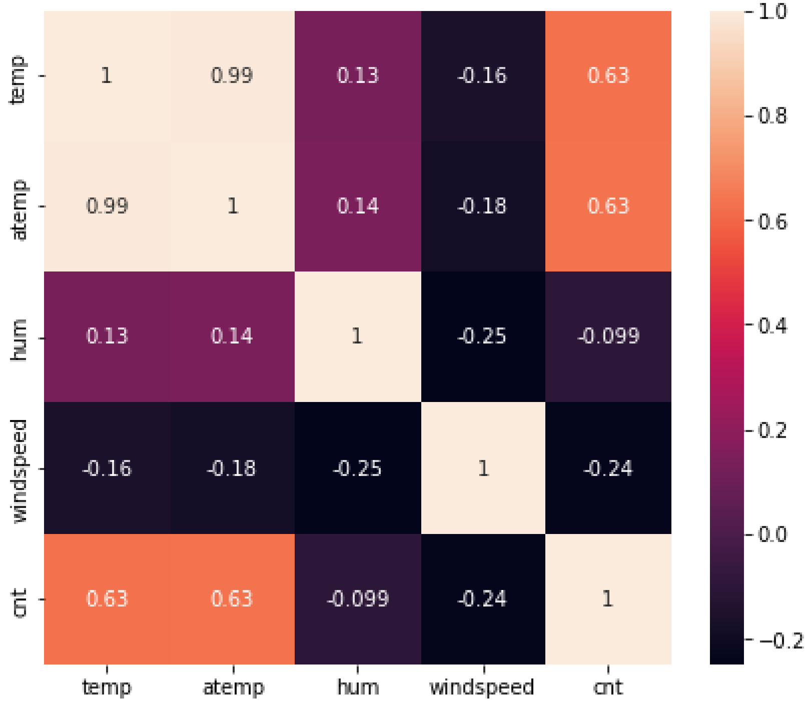
```
<Figure size 2160x2160 with 0 Axes>
```





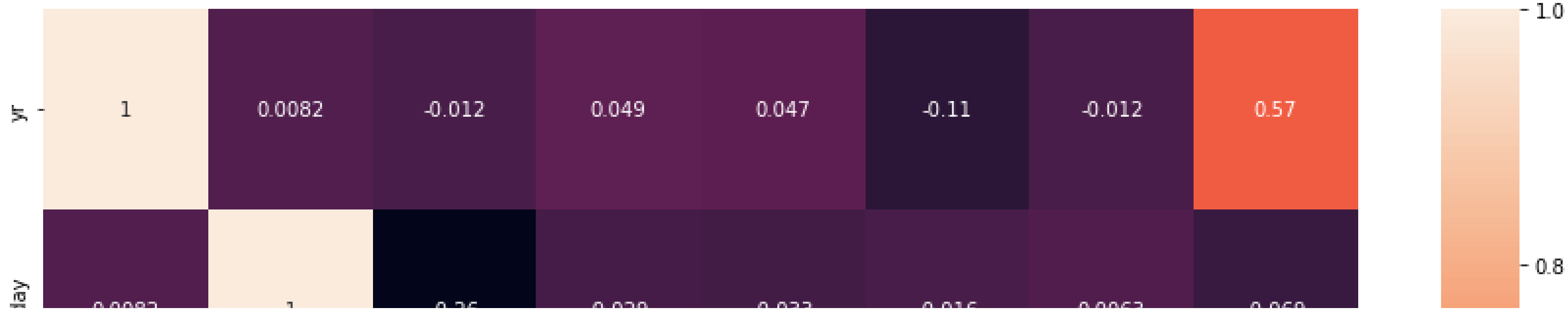
- temp and atemp has the highest correlation as compare to others with the target variable cnt.
- temp and atemp are highly co-related with each other.

In [20]:
```
plt.figure(figsize = (7,6))
sns.heatmap(new_df[['temp','atemp','hum','windspeed','cnt']].corr(), annot = True)
plt.show()
```



temp and atemp has correlation more than .99 means almost 1 (highly correlated).

In [21]:
```
plt.figure(figsize = (15,15))
sns.heatmap(new_df.corr(), annot = True)
plt.show()
```



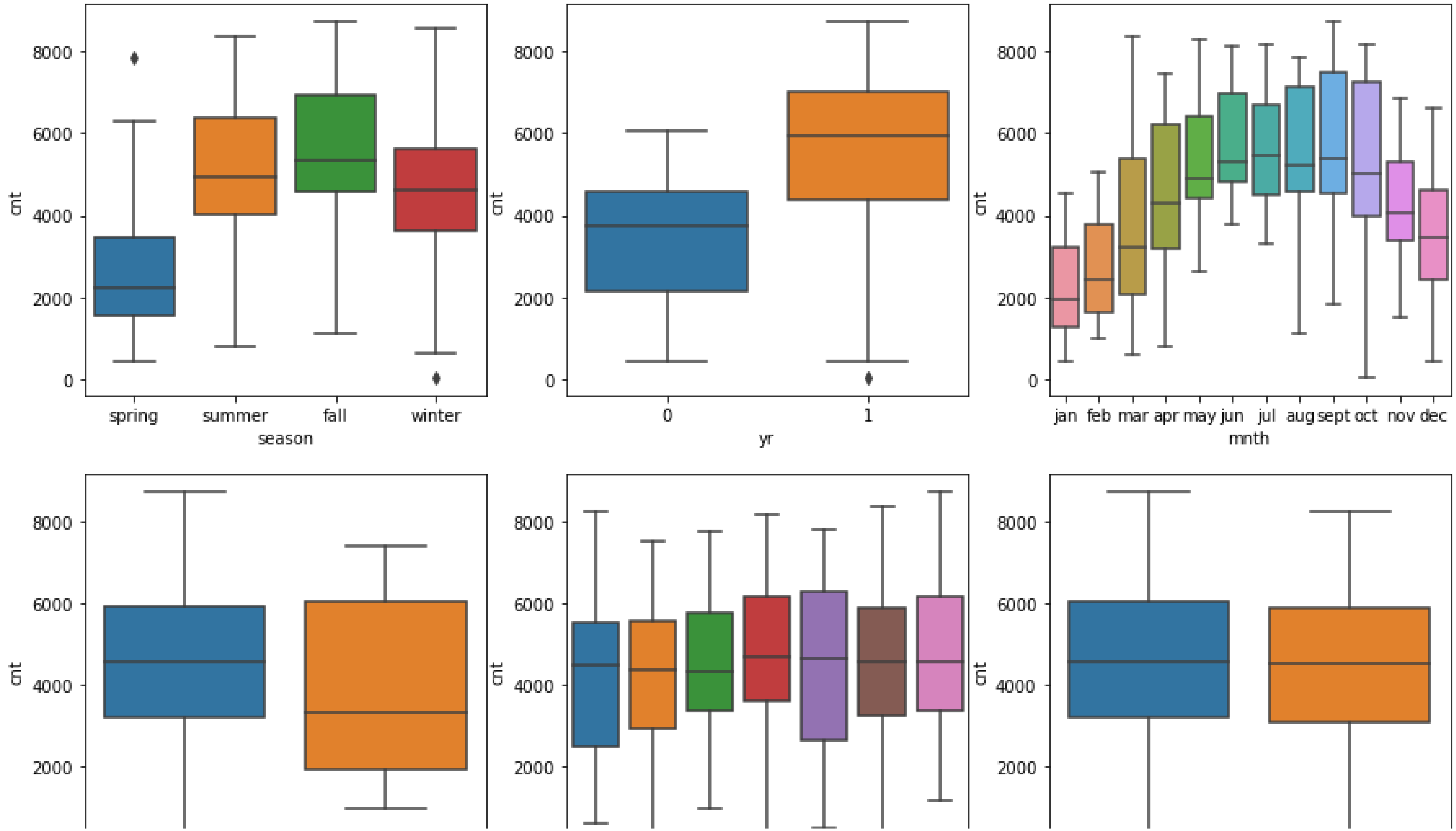

holic
workingday
temp
atemp
hum
windspeed
cnt
yr
holiday
workingday
temp
atemp
hum
windspeed
cnt
0.0082 1 -0.26 -0.029 -0.033 -0.016 0.0063 -0.069
-0.012 -0.26 1 0.002 0.011 0.054 -0.0025 -0.028
0.049 -0.029 0.002 1 0.99 0.13 -0.16 0.63
0.047 -0.033 0.011 0.99 1 0.14 -0.18 0.63
-0.11 -0.016 0.054 0.13 0.14 1 -0.25 -0.099
-0.012 0.0063 -0.0025 -0.16 -0.18 -0.25 1 -0.24
0.57 -0.069 -0.028 0.63 0.63 -0.099 -0.24 1
0.6
0.4
0.2
0.0
−0.2

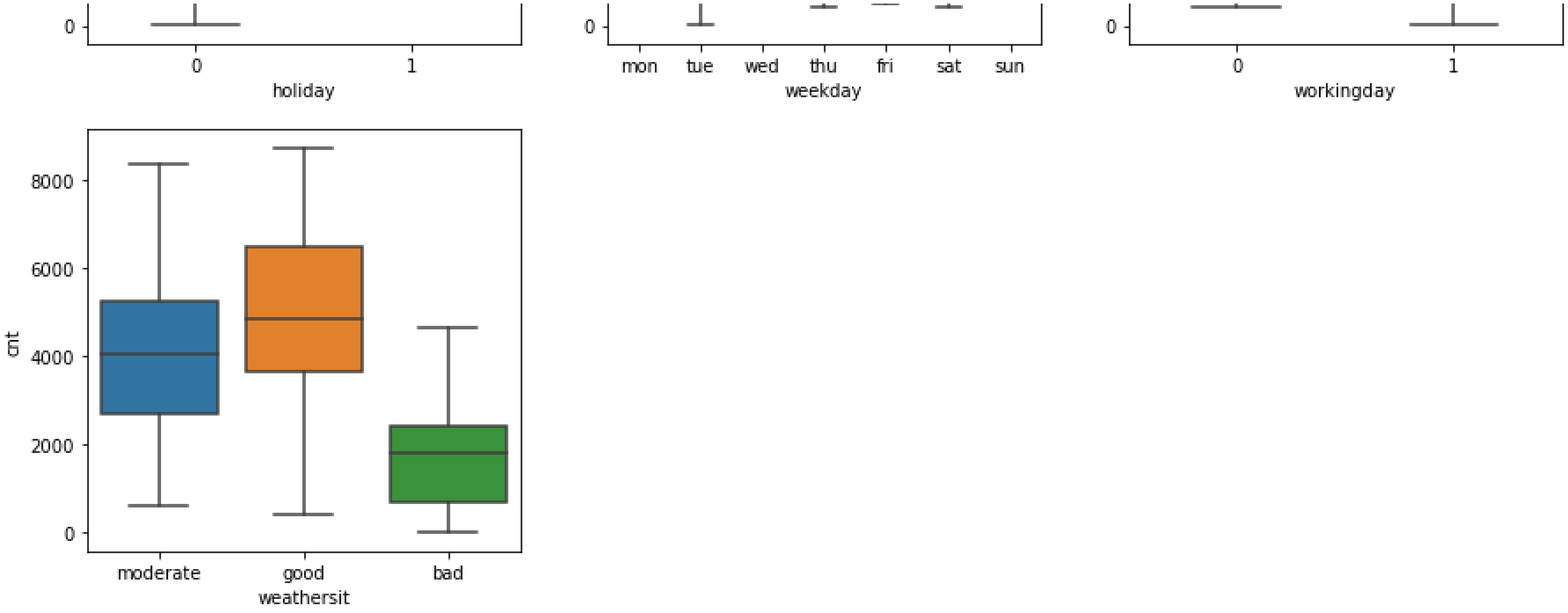
We also see Target variable has a linear relationship with some of the indeptendent variables. Good sign for building a linear regression Model.

Visualising Catagorical Variables

In [22]:

```
vars_cat = ['season','yr','mnth','holiday','weekday','workingday','weathersit']
plt.figure(figsize=(15, 15))
for i in enumerate(vars_cat):
    plt.subplot(3,3,i[0]+1)
    sns.boxplot(data=new_df, x=i[1], y='cnt')
plt.show()
```





1. Season: 3:fall has highest demand for rental bikes
2. I see that demand for next year has grown
3. Demand is continuously growing each month till June. September month has highest demand. After September, demand is decreasing
4. When there is a holiday, demand has decreased.
5. Weekday is not giving clear picture abount demand.
6. The clear weathershit has highest demand
7. During September, bike sharing is more. During the year end and beginning, it is less, could be due to extereme weather conditions.

Creating Dummy Variables

```
we need to create dummy variables for 4 categorical variables
first we will change it's data type to categorical
```

In [23]:
```python
new_df = pd.get_dummies(new_df, drop_first=True)
```

In [24]:

```
new_df.head()
```

Out[24]:

	yr	holiday	workingday	temp	atemp	hum	windspeed	cnt	season_spring	season_summer	...	mnth_oct	mnth_sept	weekday_mon	wee
0	0	0	1	14.110847	18.18125	80.5833	10.749882	985	1	0	...	0	0	1	
1	0	0	1	14.902598	17.68695	69.6087	16.652113	801	1	0	...	0	0	0	
2	0	0	1	8.050924	9.47025	43.7273	16.636703	1349	1	0	...	0	0	0	
3	0	0	1	8.200000	10.60610	59.0435	10.739832	1562	1	0	...	0	0	0	
4	0	0	1	9.305237	11.46350	43.6957	12.522300	1600	1	0	...	0	0	0	

5 rows × 30 columns

In [25]:

```
new_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 730 entries, 0 to 729
Data columns (total 30 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   yr                  730 non-null    int64  
 1   holiday             730 non-null    int64  
 2   workingday          730 non-null    int64  
 3   temp                730 non-null    float64
 4   atemp               730 non-null    float64
 5   hum                 730 non-null    float64
 6   windspeed           730 non-null    float64
 7   cnt                 730 non-null    int64  
 8   season_spring       730 non-null    uint8  
 9   season_summer       730 non-null    uint8  
 10  season_winter       730 non-null    uint8  
 11  mnth_aug            730 non-null    uint8  
 12  mnth_dec            730 non-null    uint8  
 13  mnth_feb            730 non-null    uint8  
 14  mnth_jan            730 non-null    uint8  
 15  mnth_jul            730 non-null    uint8  
 16  mnth_jun            730 non-null    uint8  
 17  mnth_mar            730 non-null    uint8  
```

```
 18  mnth_may            730 non-null    uint8
 19  mnth_nov            730 non-null    uint8
 20  mnth_oct            730 non-null    uint8
 21  mnth_sept           730 non-null    uint8
 22  weekday_mon         730 non-null    uint8
 23  weekday_sat         730 non-null    uint8
 24  weekday_sun         730 non-null    uint8
 25  weekday_thu         730 non-null    uint8
 26  weekday_tue         730 non-null    uint8
 27  weekday_wed         730 non-null    uint8
 28  weathersit_good     730 non-null    uint8
 29  weathersit_moderate 730 non-null    uint8
dtypes: float64(4), int64(4), uint8(22)
memory usage: 61.4 KB
```

SPLITTING THE DATA

    we will split data into 75:25 ratio for train and test data set

In [26]:

```
y=new_df.pop('cnt')
X=new_df
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.25)
```

In [27]:

```
print(X_train.shape)
print(X_test.shape)
```

```
(547, 29)
(183, 29)
```

RESCALING THE FEATURES

    scale continuous variables
    Fit and transform training set

In [28]:

```
num_vars = ['temp','atemp','hum','windspeed']
scaler = MinMaxScaler()
X_train[num_vars] = scaler.fit_transform(X_train[num_vars])
```

In [29]:
```
X_train.describe()
```

Out[29]:

	yr	holiday	workingday	temp	atemp	hum	windspeed	season_spring	season_summer	season_winter	...	mnth_oct
count	547.000000	547.000000	547.000000	547.000000	547.000000	547.000000	547.000000	547.000000	547.000000	547.000000	...	547.000000
mean	0.506399	0.023766	0.692870	0.549725	0.525543	0.652339	0.401932	0.226691	0.259598	0.255941	...	0.084095
std	0.500417	0.152459	0.461726	0.226757	0.212513	0.147580	0.183190	0.419074	0.438815	0.436789	...	0.277784
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...	0.000000
25%	0.000000	0.000000	0.000000	0.354221	0.345166	0.540488	0.269271	0.000000	0.000000	0.000000	...	0.000000
50%	1.000000	0.000000	1.000000	0.553679	0.541742	0.655527	0.379817	0.000000	0.000000	0.000000	...	0.000000
75%	1.000000	0.000000	1.000000	0.743002	0.697971	0.755887	0.502951	0.000000	1.000000	1.000000	...	0.000000
max	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	...	1.000000

8 rows × 29 columns

In [30]:
```
X_train.head()
```

Out[30]:

	yr	holiday	workingday	temp	atemp	hum	windspeed	season_spring	season_summer	season_winter	...	mnth_oct	mnth_sept	weekday
0	0	0	1	0.355170	0.373517	0.828620	0.329351	1	0	0	...	0	0	
113	0	0	1	0.651106	0.620474	0.833761	0.405045	0	1	0	...	0	0	
595	1	0	1	0.718600	0.688457	0.731791	0.152821	0	0	0	...	0	0	
662	1	0	1	0.611648	0.591497	0.823051	0.243297	0	0	1	...	1	0	
715	1	0	1	0.416433	0.423233	0.932733	0.180992	0	0	1	...	0	0	

5 rows × 29 columns

In [ ]:

In [31]:
```
lr = LinearRegression()
lr.fit(X_train, y_train)
rfe = RFE(lr, 15)
rfe = rfe.fit(X_train, y_train)
```

In [32]:
```
list(zip(X_train.columns,rfe.support_,rfe.ranking_))
```

Out[32]:
```
[('yr', True, 1),
 ('holiday', True, 1),
 ('workingday', True, 1),
 ('temp', False, 7),
 ('atemp', True, 1),
 ('hum', True, 1),
 ('windspeed', True, 1),
 ('season_spring', True, 1),
 ('season_summer', False, 13),
 ('season_winter', True, 1),
 ('mnth_aug', False, 12),
 ('mnth_dec', True, 1),
 ('mnth_feb', False, 3),
 ('mnth_jan', False, 2),
 ('mnth_jul', True, 1),
 ('mnth_jun', False, 6),
 ('mnth_mar', False, 14),
 ('mnth_may', False, 5),
 ('mnth_nov', True, 1),
 ('mnth_oct', False, 15),
 ('mnth_sept', False, 4),
 ('weekday_mon', False, 9),
 ('weekday_sat', True, 1),
 ('weekday_sun', True, 1),
 ('weekday_thu', False, 11),
 ('weekday_tue', False, 8),
 ('weekday_wed', False, 10),
 ('weathersit_good', True, 1),
 ('weathersit_moderate', True, 1)]
```

In [33]:
```
col = X_train.columns[rfe.support_]
col
```

Out[33]:
```
Index(['yr', 'holiday', 'workingday', 'atemp', 'hum', 'windspeed',
       'season_spring', 'season_winter', 'mnth_dec', 'mnth_jul', 'mnth_nov',
       'weekday_sat', 'weekday_sun', 'weathersit_good', 'weathersit_moderate'],
      dtype='object')
```

In [34]:
```
X_train.columns[~rfe.support_]
```

Out[34]:
```
Index(['temp', 'season_summer', 'mnth_aug', 'mnth_feb', 'mnth_jan', 'mnth_jun',
       'mnth_mar', 'mnth_may', 'mnth_oct', 'mnth_sept', 'weekday_mon',
       'weekday_thu', 'weekday_tue', 'weekday_wed'],
      dtype='object')
```

In [35]:
```
X_train_rfe = X_train[col]
```

In [36]:
```
#Function to build a model
def build_model(cols):
    X_train_sm = sm.add_constant(X_train[cols])
    lm = sm.OLS(y_train, X_train_sm).fit()
    print(lm.summary())
    return lm
```

In [37]:
```
#Function to calculate VIFs
def vif(cols):
    df1 = X_train[cols]
    vif = pd.DataFrame()
    vif['Features'] = df1.columns
    vif['VIF'] = [variance_inflation_factor(df1.values, i) for i in range(df1.shape[1])]
    vif['VIF'] = round(vif['VIF'],2)
    print(vif.sort_values(by='VIF',ascending=False))
```

Building Linear Model using 'STATS MODEL'

Model 1

In [38]:
```
#Selected columns for Model 1 - all columns selected by RFE
cols = ['yr', 'workingday', 'temp','atemp', 'hum', 'windspeed', 'season_spring',
        'season_winter', 'mnth_dec', 'mnth_jul', 'mnth_nov', 'weekday_sat','weekday_sun','weathersit_good', 'weathersit_moderate']
```

```
build_model(cols)
vif(cols)
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    cnt   R-squared:                       0.837
Model:                            OLS   Adj. R-squared:                  0.833
Method:                 Least Squares   F-statistic:                     182.4
Date:                Mon, 30 May 2022   Prob (F-statistic):          3.45e-198
Time:                        20:22:55   Log-Likelihood:                -4415.9
No. Observations:                 547   AIC:                             8864.
Df Residuals:                     531   BIC:                             8933.
Df Model:                          15
Covariance Type:            nonrobust
=======================================================================================
                          coef    std err          t      P>|t|      [0.025      0.975]
---------------------------------------------------------------------------------------
const                 429.4157    479.679      0.895      0.371    -512.885    1371.717
yr                   2053.9186     69.043     29.748      0.000    1918.287    2189.550
workingday            641.2905    243.982      2.628      0.009     162.003    1120.578
temp                 1216.5811   1903.666      0.639      0.523   -2523.059    4956.221
atemp                3024.4572   2025.549      1.493      0.136    -954.616    7003.531
hum                 -1425.8961    323.567     -4.407      0.000   -2061.525    -790.267
windspeed           -1257.7392    203.413     -6.183      0.000   -1657.332    -858.146
season_spring        -979.2636    129.120     -7.584      0.000   -1232.913    -725.614
season_winter         769.2418    116.837      6.584      0.000     539.722     998.762
mnth_dec             -481.7352    141.390     -3.407      0.001    -759.487    -203.983
mnth_jul             -823.7379    136.960     -6.014      0.000   -1092.787    -554.688
mnth_nov             -495.3220    154.149     -3.213      0.001    -798.139    -192.505
weekday_sat           822.8480    257.063      3.201      0.001     317.862    1327.834
weekday_sun           864.6451    256.685      3.369      0.001     360.403    1368.887
weathersit_good      1970.7615    228.674      8.618      0.000    1521.545    2419.979
weathersit_moderate  1494.7175    214.001      6.985      0.000    1074.326    1915.109
==============================================================================
Omnibus:                       92.372   Durbin-Watson:                   1.964
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              248.475
Skew:                          -0.837   Prob(JB):                     1.11e-54
Kurtosis:                       5.846   Cond. No.                         157.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

```
              Features      VIF
3               atemp  1158.80
2                temp  1121.10
4                 hum    27.44
1          workingday    24.85
13     weathersit_good    16.84
14 weathersit_moderate    10.08
12         weekday_sun     5.93
5           windspeed     5.75
11         weekday_sat     5.66
6        season_spring     3.04
7        season_winter     3.03
0                  yr     2.08
10            mnth_nov     1.79
8             mnth_dec     1.50
9             mnth_jul     1.45
```

Model 2

In [39]:

```
#Dropping the variable atemp as it has negative coefficient
cols = ['yr', 'workingday', 'temp', 'hum', 'windspeed', 'season_spring',
        'season_winter', 'mnth_dec', 'mnth_jul', 'mnth_nov', 'weekday_sat','weekday_sun','weathersit_good', 'weathersit_moderate']

build_model(cols)
vif(cols)
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    cnt   R-squared:                       0.837
Model:                            OLS   Adj. R-squared:                  0.832
Method:                 Least Squares   F-statistic:                     194.8
Date:                Mon, 30 May 2022   Prob (F-statistic):          7.34e-199
Time:                        20:22:55   Log-Likelihood:                -4417.1
No. Observations:                 547   AIC:                             8864.
Df Residuals:                     532   BIC:                             8929.
Df Model:                          14
Covariance Type:            nonrobust
=======================================================================================
                          coef    std err          t      P>|t|      [0.025      0.975]
---------------------------------------------------------------------------------------
const                 388.8672    479.463      0.811      0.418    -553.005    1330.739
yr                   2061.5376     68.934     29.906      0.000    1926.121    2196.954
```

```
workingday           659.3286    243.964      2.703      0.007     180.078    1138.579
temp                4030.7677    268.170     15.031      0.000    3503.965    4557.570
hum                -1366.5533    321.488     -4.251      0.000   -1998.095    -735.011
windspeed          -1308.2761    200.809     -6.515      0.000   -1702.752    -913.800
season_spring       -984.8215    129.216     -7.622      0.000   -1238.657    -730.986
season_winter        782.7060    116.623      6.711      0.000     553.607    1011.805
mnth_dec            -469.2488    141.305     -3.321      0.001    -746.834    -191.664
mnth_jul            -827.0616    137.100     -6.033      0.000   -1096.385    -557.738
mnth_nov            -492.4505    154.315     -3.191      0.002    -795.593    -189.308
weekday_sat          839.4844    257.118      3.265      0.001     334.392    1344.576
weekday_sun          879.7216    256.782      3.426      0.001     375.290    1384.153
weathersit_good     2017.0251    226.827      8.892      0.000    1571.439    2462.612
weathersit_moderate 1528.4341    213.052      7.174      0.000    1109.908    1946.960
==============================================================================
Omnibus:                       89.035   Durbin-Watson:                   1.967
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              239.204
Skew:                          -0.808   Prob(JB):                     1.14e-52
Kurtosis:                       5.808   Cond. No.                         33.6
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
               Features    VIF
3                   hum  27.10
1            workingday  24.84
2                  temp  21.19
12      weathersit_good  16.55
13  weathersit_moderate  10.00
11          weekday_sun   5.93
10          weekday_sat   5.66
4             windspeed   5.49
5         season_spring   3.04
6         season_winter   3.01
0                    yr   2.07
9              mnth_nov   1.79
7              mnth_dec   1.50
8              mnth_jul   1.45
```

Model 3

In [40]:

```
#Dropping the variable hum as it has negative coefficient
cols = ['yr', 'workingday', 'temp', 'windspeed', 'season_spring',
        'season_winter', 'mnth_dec', 'mnth_jul', 'mnth_nov', 'weekday_sat','weekday_sun','weathersit_good', 'weathersit_moderate']
```

```
build_model(cols)
vif(cols)
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    cnt   R-squared:                       0.831
Model:                            OLS   Adj. R-squared:                  0.827
Method:                 Least Squares   F-statistic:                     201.9
Date:                Mon, 30 May 2022   Prob (F-statistic):          3.51e-196
Time:                        20:22:55   Log-Likelihood:                -4426.2
No. Observations:                 547   AIC:                             8880.
Df Residuals:                     533   BIC:                             8941.
Df Model:                          13
Covariance Type:            nonrobust
=======================================================================================
                          coef    std err          t      P>|t|      [0.025      0.975]
---------------------------------------------------------------------------------------
const                -788.4340    397.592     -1.983      0.048   -1569.474      -7.394
yr                   2108.0625     69.141     30.489      0.000    1972.241    2243.884
workingday            693.8301    247.702      2.801      0.005     207.238    1180.422
temp                 3734.7335    263.083     14.196      0.000    3217.926    4251.541
windspeed           -1074.5847    196.205     -5.477      0.000   -1460.014    -689.155
season_spring       -1019.9918    130.999     -7.786      0.000   -1277.330    -762.654
season_winter         720.6695    117.545      6.131      0.000     489.762     951.577
mnth_dec             -529.9093    142.816     -3.710      0.000    -810.461    -249.358
mnth_jul             -748.5225    138.007     -5.424      0.000   -1019.627    -477.418
mnth_nov             -498.5972    156.760     -3.181      0.002    -806.540    -190.654
weekday_sat           923.5288    260.429      3.546      0.000     411.935    1435.123
weekday_sun           943.6203    260.414      3.624      0.000     432.056    1455.184
weathersit_good      2419.1398    209.434     11.551      0.000    2007.723    2830.557
weathersit_moderate  1699.1657    212.555      7.994      0.000    1281.617    2116.714
==============================================================================
Omnibus:                       93.604   Durbin-Watson:                   1.952
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              253.363
Skew:                          -0.846   Prob(JB):                     9.61e-56
Kurtosis:                       5.873   Cond. No.                         27.8
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
               Features    VIF
1            workingday  20.23
```

```
11      weathersit_good  16.46
2                  temp  16.03
12  weathersit_moderate   9.68
3             windspeed   5.49
10          weekday_sun   5.08
9           weekday_sat   4.93
5         season_winter   2.83
4         season_spring   2.76
0                    yr   2.06
8              mnth_nov   1.78
6              mnth_dec   1.44
7              mnth_jul   1.41
```

Model 4

In [41]:

```
#Dropping the variable workingday as it has high VIF
cols = ['yr', 'temp', 'windspeed', 'season_spring',
        'season_winter', 'mnth_dec', 'mnth_jul', 'mnth_nov', 'weekday_sat','weekday_sun','weathersit_good', 'weathersit_moderate']

build_model(cols)
vif(cols)
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    cnt   R-squared:                       0.829
Model:                            OLS   Adj. R-squared:                  0.825
Method:                 Least Squares   F-statistic:                     215.3
Date:                Mon, 30 May 2022   Prob (F-statistic):          1.13e-195
Time:                        20:22:55   Log-Likelihood:                -4430.2
No. Observations:                 547   AIC:                             8886.
Df Residuals:                     534   BIC:                             8942.
Df Model:                          12
Covariance Type:            nonrobust
=======================================================================================
                          coef    std err          t      P>|t|      [0.025      0.975]
---------------------------------------------------------------------------------------
const                 -48.0196    298.895     -0.161      0.872    -635.174     539.135
yr                   2095.9453     69.446     30.181      0.000    1959.524    2232.366
temp                 3699.8013    264.467     13.990      0.000    3180.279    4219.324
windspeed           -1110.8402    197.028     -5.638      0.000   -1497.886    -723.794
season_spring       -1044.1901    131.549     -7.938      0.000   -1302.607    -785.773
season_winter         726.2638    118.279      6.140      0.000     493.915     958.613
```

```
mnth_dec             -559.6773    143.330     -3.905      0.000     -841.237    -278.117
mnth_jul             -731.8317    138.759     -5.274      0.000    -1004.413    -459.250
mnth_nov             -518.6200    157.597     -3.291      0.001     -828.207    -209.033
weekday_sat           250.1685    100.810      2.482      0.013       52.136     448.202
weekday_sun           268.8416     99.538      2.701      0.007       73.308     464.375
weathersit_good      2399.6211    210.655     11.391      0.000     1985.806    2813.436
weathersit_moderate  1678.2917    213.782      7.850      0.000     1258.335    2098.248
==============================================================================
Omnibus:                       89.938   Durbin-Watson:                   1.954
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              234.530
Skew:                          -0.826   Prob(JB):                     1.18e-51
Kurtosis:                       5.750   Cond. No.                         19.0
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
               Features    VIF
1                  temp  13.26
10      weathersit_good  12.12
11  weathersit_moderate   7.15
2             windspeed   5.09
4         season_winter   2.65
3         season_spring   2.51
0                    yr   2.05
7              mnth_nov   1.77
5              mnth_dec   1.43
6              mnth_jul   1.40
9           weekday_sun   1.22
8           weekday_sat   1.20
```

Model 5

In [42]:

```
# Dropping the variable windspeed as it has negative coefficient
cols = ['yr', 'temp', 'season_spring','weathersit_good',
       'season_summer', 'season_winter', 'mnth_jul', 'mnth_sept', 'weathersit_moderate']

build_model(cols)
vif(cols)
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    cnt   R-squared:                       0.814
```

```
Model:                            OLS   Adj. R-squared:                  0.811
Method:                 Least Squares   F-statistic:                     261.7
Date:                Mon, 30 May 2022   Prob (F-statistic):          6.03e-190
Time:                        20:22:55   Log-Likelihood:                -4452.3
No. Observations:                 547   AIC:                             8925.
Df Residuals:                     537   BIC:                             8968.
Df Model:                           9
Covariance Type:            nonrobust
=======================================================================================
                          coef    std err          t      P>|t|      [0.025      0.975]
---------------------------------------------------------------------------------------
const               -1191.2018    308.822     -3.857      0.000   -1797.848    -584.555
yr                   2113.5071     72.144     29.296      0.000    1971.789    2255.226
temp                 4136.2291    283.983     14.565      0.000    3578.375    4694.083
season_spring        -752.2283    184.138     -4.085      0.000   -1113.947    -390.509
weathersit_good      2617.1339    216.151     12.108      0.000    2192.529    3041.739
season_summer         255.5120    132.433      1.929      0.054      -4.638     515.661
season_winter         761.0041    148.760      5.116      0.000     468.781    1053.227
mnth_jul             -507.7791    161.678     -3.141      0.002    -825.377    -190.181
mnth_sept             544.3750    143.006      3.807      0.000     263.455     825.295
weathersit_moderate  1917.9020    219.032      8.756      0.000    1487.637    2348.167
==============================================================================
Omnibus:                       76.273   Durbin-Watson:                   2.004
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              182.354
Skew:                          -0.734   Prob(JB):                     2.53e-40
Kurtosis:                       5.418   Cond. No.                         18.9
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
              Features    VIF
3      weathersit_good  13.67
1                 temp  12.87
8  weathersit_moderate   8.23
2        season_spring   3.33
4        season_summer   2.75
5        season_winter   2.66
0                   yr   2.06
6             mnth_jul   1.78
7            mnth_sept   1.42
```

Model 6

In [43]:
```python
# Dropping the variable season_spring as it has negative coefficient
cols = ['yr', 'temp','weathersit_good',
        'season_summer', 'season_winter', 'mnth_jul', 'mnth_sept', 'weathersit_moderate']

build_model(cols)
vif(cols)
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    cnt   R-squared:                       0.809
Model:                            OLS   Adj. R-squared:                  0.806
Method:                 Least Squares   F-statistic:                     284.1
Date:                Mon, 30 May 2022   Prob (F-statistic):          1.25e-187
Time:                        20:22:55   Log-Likelihood:                -4460.7
No. Observations:                 547   AIC:                             8939.
Df Residuals:                     538   BIC:                             8978.
Df Model:                           8
Covariance Type:            nonrobust
=======================================================================================
                          coef    std err          t      P>|t|      [0.025      0.975]
---------------------------------------------------------------------------------------
const               -2032.8971    233.371     -8.711      0.000   -2491.327   -1574.467
yr                   2089.4050     72.943     28.644      0.000    1946.117    2232.693
temp                 4971.0131    200.057     24.848      0.000    4578.025    5364.001
weathersit_good      2601.2182    219.244     11.864      0.000    2170.539    3031.897
season_summer         622.3268     98.752      6.302      0.000     428.340     816.314
season_winter        1240.2422     92.797     13.365      0.000    1057.954    1422.531
mnth_jul             -363.2369    160.042     -2.270      0.024    -677.621     -48.853
mnth_sept             701.5252    139.729      5.021      0.000     427.044     976.006
weathersit_moderate  1887.8028    222.077      8.501      0.000    1451.559    2324.047
==============================================================================
Omnibus:                       67.771   Durbin-Watson:                   2.009
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              134.688
Skew:                          -0.719   Prob(JB):                     5.66e-30
Kurtosis:                       4.961   Cond. No.                         15.6
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
              Features   VIF
1                 temp  9.58
2      weathersit_good  5.92
7  weathersit_moderate  3.55
```

```
0                 yr  2.03
3      season_summer  1.92
5          mnth_jul   1.70
4      season_winter  1.58
6         mnth_sept   1.35
```

Model 7

In [44]:

```python
# Dropping the variable mnth_jul as it has negative coefficient
cols = ['yr', 'temp','weathersit_good',
       'season_summer', 'season_winter', 'mnth_sept', 'weathersit_moderate']

build_model(cols)
vif(cols)
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    cnt   R-squared:                       0.807
Model:                            OLS   Adj. R-squared:                  0.804
Method:                 Least Squares   F-statistic:                     321.4
Date:                Mon, 30 May 2022   Prob (F-statistic):          8.68e-188
Time:                        20:22:55   Log-Likelihood:                -4463.3
No. Observations:                 547   AIC:                             8943.
Df Residuals:                     539   BIC:                             8977.
Df Model:                           7
Covariance Type:            nonrobust
=======================================================================================
                          coef    std err          t      P>|t|      [0.025      0.975]
---------------------------------------------------------------------------------------
const               -1995.1587    233.673     -8.538      0.000   -2454.179   -1536.138
yr                   2094.0993     73.194     28.610      0.000    1950.319    2237.879
temp                 4742.2281    173.469     27.338      0.000    4401.469    5082.987
weathersit_good      2608.6724    220.062     11.854      0.000    2176.388    3040.957
season_summer         709.5448     91.318      7.770      0.000     530.162     888.928
season_winter        1282.7491     91.237     14.060      0.000    1103.526    1461.972
mnth_sept             796.7939    133.787      5.956      0.000     533.986    1059.602
weathersit_moderate  1907.4821    222.760      8.563      0.000    1469.897    2345.067
==============================================================================
Omnibus:                       68.833   Durbin-Watson:                   2.009
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              136.488
Skew:                          -0.729   Prob(JB):                     2.30e-30
Kurtosis:                       4.965   Cond. No.                         15.5
```

```
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
              Features   VIF
1                 temp  7.09
2      weathersit_good  5.79
6  weathersit_moderate  3.42
0                   yr  2.03
3        season_summer  1.62
4        season_winter  1.50
5            mnth_sept  1.23
```

Model 8

In [45]:

```python
# Dropping the variable weathersit_moderate
cols = ['yr', 'temp','weathersit_good',
       'season_summer', 'season_winter', 'mnth_sept']

build_model(cols)
vif(cols)
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    cnt   R-squared:                       0.780
Model:                            OLS   Adj. R-squared:                  0.778
Method:                 Least Squares   F-statistic:                     320.0
Date:                Mon, 30 May 2022   Prob (F-statistic):          3.63e-174
Time:                        20:22:55   Log-Likelihood:                -4498.1
No. Observations:                 547   AIC:                             9010.
Df Residuals:                     540   BIC:                             9040.
Df Model:                           6
Covariance Type:            nonrobust
===================================================================================
                      coef    std err          t      P>|t|      [0.025      0.975]
-----------------------------------------------------------------------------------
const            -272.9090    126.670     -2.154      0.032    -521.736     -24.082
yr               2130.2938     77.811     27.378      0.000    1977.444    2283.144
temp             4786.4160    184.639     25.923      0.000    4423.717    5149.115
weathersit_good   841.2969     81.286     10.350      0.000     681.622    1000.972
season_summer     736.8244     97.182      7.582      0.000     545.923     927.726
season_winter    1256.4925     97.100     12.940      0.000    1065.753    1447.232
```

```
mnth_sept        795.5029    142.465      5.584      0.000     515.650    1075.356
==============================================================================
Omnibus:                      112.651   Durbin-Watson:                   2.028
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              336.215
Skew:                          -0.977   Prob(JB):                     9.81e-74
Kurtosis:                       6.306   Cond. No.                         8.19
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
          Features   VIF
1             temp  3.90
2  weathersit_good  2.45
0               yr  1.89
3    season_summer  1.56
4    season_winter  1.30
5        mnth_sept  1.23
```

Here VIF seems to be almost accepted. p-value for all the features is almost 0.0 and R2 is 0.815 Let us select **Model 7** as our final as it has all important statistics high (R-square, Adjusted R-squared and F-statistic), along with no insignificant variables and no multi coliinear (high VIF) variables. Difference between R-squared and Adjusted R-squared values for this model is veryless, which also means that there are no additional parameters that can be removed from this model.

In [46]:

```
def build_model_sk(X,y):
    lr1 = LinearRegression()
    lr1.fit(X,y)
    return lr1
```

In [47]:

```
cols = ['yr', 'temp','weathersit_good',
       'season_summer', 'season_winter', 'mnth_sept', 'weathersit_moderate']
lr = build_model_sk(X_train[cols],y_train)
print(lr.intercept_,lr.coef_)
```

```
-1995.1586680261207 [2094.09931633 4742.22814376 2608.67237747  709.54475518 1282.74912191
  796.79392462 1907.48208925]
```

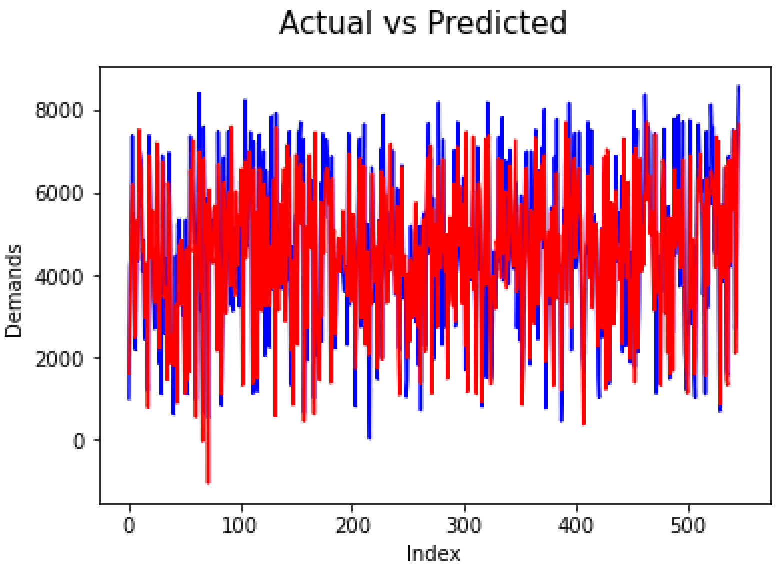
Model Evaluation

In [48]:

```
y_train_pred = lr.predict(X_train[cols])
```

In [49]:

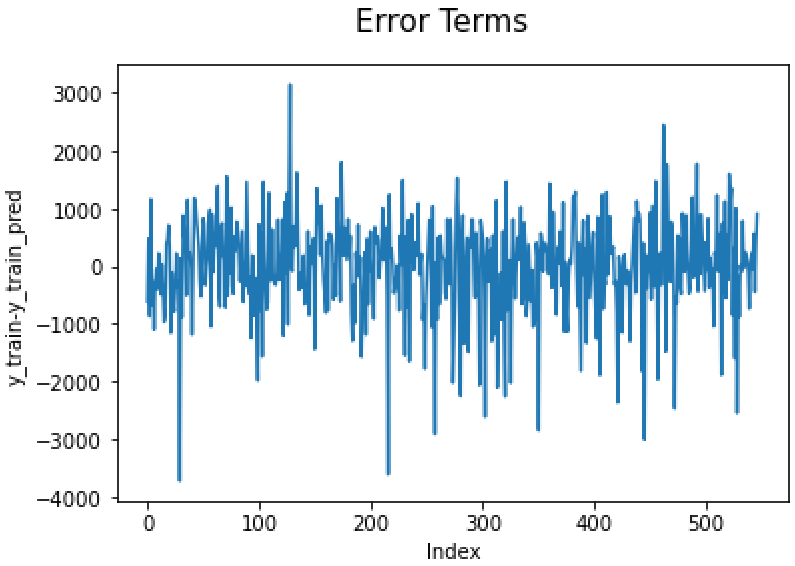
```
# Actual vs Predicted
c = [i for i in range(0,len(X_train),1)]
plt.plot(c,y_train, color="blue")
plt.plot(c,y_train_pred, color="red")
plt.suptitle('Actual vs Predicted', fontsize = 15)
plt.xlabel('Index')
plt.ylabel('Demands')
plt.show()
```



Actual and Predicted result following almost the same pattern so this model seems ok

In [50]:

```
# Error Terms
c = [i for i in range(0,len(X_train),1)]
plt.plot(c,y_train-y_train_pred)
plt.suptitle('Error Terms', fontsize = 15)
plt.xlabel('Index')
plt.ylabel('y_train-y_train_pred')
plt.show()
```



In [51]:
```
#Scale variables in X_test
num_vars = ['temp','atemp','hum','windspeed']

#Test data to be transformed only, no fitting
X_test[num_vars] = scaler.transform(X_test[num_vars])
```

In [52]:
```
#Columns from our final model
cols = ['yr', 'temp','weathersit_good',
       'season_summer', 'season_winter', 'mnth_sept', 'weathersit_moderate']

#Predict the values for test data
y_test_pred = lr.predict(X_test[cols])
```

Here,If we see the error terms are independent of each other

In [53]:
```
r2_score(y_train,y_train_pred)
```

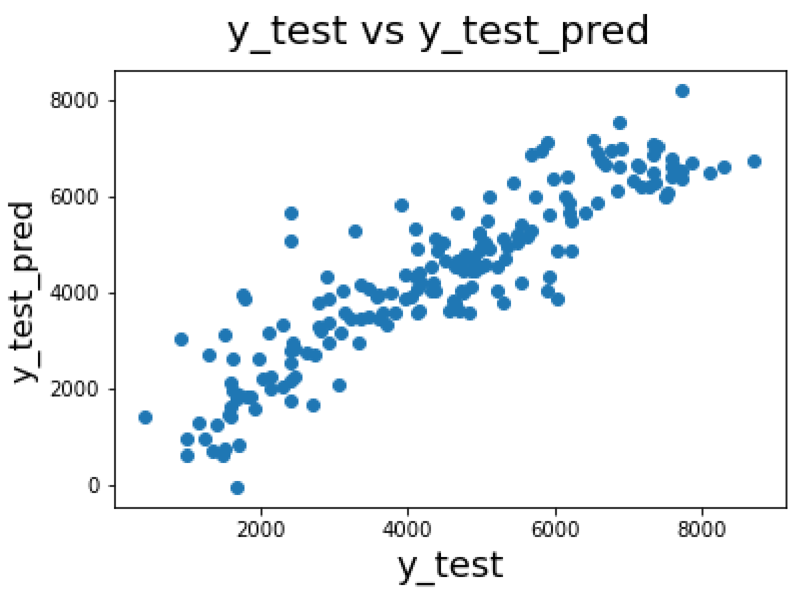
Out[53]:
```
0.8067505299801407
```

R2 Same as we obtained for our final model

In [54]:
```
# Plotting y_test and y_test_pred to understand the spread

fig = plt.figure()
plt.scatter(y_test, y_test_pred)
fig.suptitle('y_test vs y_test_pred', fontsize = 20)
plt.xlabel('y_test', fontsize = 18)
plt.ylabel('y_test_pred', fontsize = 16)
```

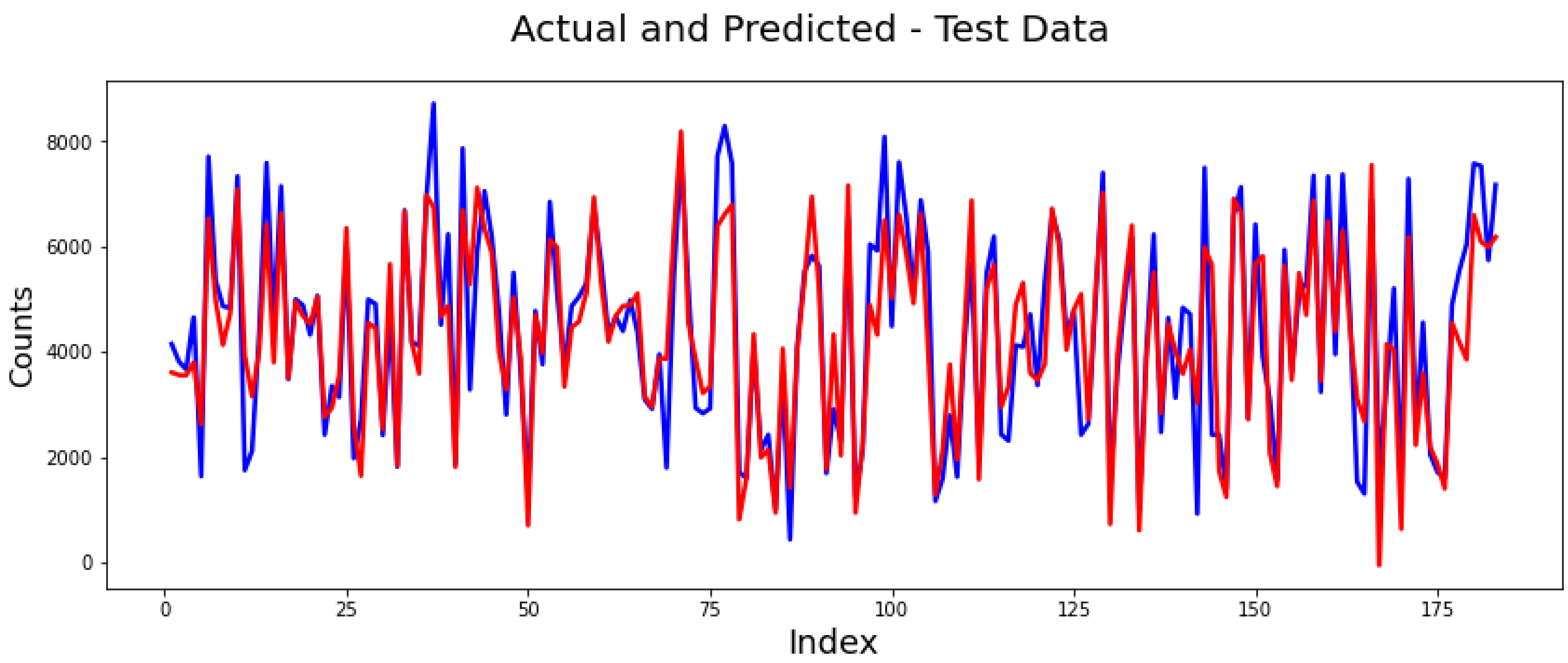
Out[54]:
```
Text(0, 0.5, 'y_test_pred')
```



We can observe that variance of the residuals (error terms) is constant across predictions. i.e error term does not vary much as the value of the predictor variable changes.

In [55]:
```
# Plot Test vs Predicted test values
c = [i for i in range(1,len(y_test)+1,1)]
fig = plt.figure(figsize=(14,5))
plt.plot(c,y_test, color="blue", linewidth=2.5, linestyle="-")
plt.plot(c,y_test_pred, color="red",  linewidth=2.5, linestyle="-")
fig.suptitle('Actual and Predicted - Test Data', fontsize=20)
plt.xlabel('Index', fontsize=18)
plt.ylabel('Counts', fontsize=16)
```

Out[55]:
```
Text(0, 0.5, 'Counts')
```



R^2 Value for TEST

In [56]:
```
r2_score(y_test, y_test_pred)
```
Out[56]: 0.8060731984796423

- **Train R^2 :0.807**
- **Test R^2 :0.806**
- This seems to be a really good model that can very well 'Generalize' various datasets.

conclusion

As per our final Model, the top 3 predictor variables that influences the bike booking are:

- **Temperature (temp)**: - High coefficient value indicated that a unit increase in temp variable increases the bike hire numbers.

- **weathersit_good**: - 2nd Highest coefficient value indicated that, a unit increase in Weathersit_good variable increases the bike hire numbers.

- **Year (yr)**: - 3rd Highest coefficient value indicated that a unit increase in yr variable increases the bike hire numbers.

So, it's suggested to consider these variables most importance while planning, to achive maximum Booking The next best features that can also be considered are

- **weathersit_moderate**: - 4th Highest coefficient value indicated that, a unit increase in weathersit_moderate variable increases the bike hire numbers.

- **season_ winter**: - 5th Highest coefficient value indicated that, a unit increase in season_ winter variable increases the bike hire numbers.

NOTE:

The details of weathersit_good & weathersit_moderate:

- weathersit_good: Clear, Few clouds, Partly cloudy, Partly cloudy
- weathersit_moderate: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist

In [ ]: