Problem Statement

Yulu, a leading shared electric cycle rental service, has recently experienced a noticeable decline in its revenue. To address this issue and enhance operational efficiency, the company seeks to analyze and understand the key factors influencing the demand for its electric cycles in the Indian market.

Objective

The objective of this analysis is to identify and understand how various factors such as seasonality, weather conditions, holidays, working days, and other environmental factors affect the usage and rental patterns of shared electric cycles. This insight will help stakeholders optimize operations, marketing strategies, and resource allocation to better meet consumer needs and increase adoption of shared electric cycles. The specific objectives are as follows:

- **Analyze the impact of environmental factors** (e.g., temperature, humidity, wind speed) on the number of shared electric cycles rented.
- **Evaluate the influence of working days** on the demand for shared electric cycles.
- **Assess whether seasonal variations** affect the number of cycles rented.
- **Assess the effect of weather conditions** on rental patterns.

Importing Libraries

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import scipy.stats as stats
```

Loading the data

```
df=pd.read_csv("bike_sharing.csv")
df.head()
```

	datetime	season	holiday	workingday	weather	temp	atemp	humidity	windspeed	casual	registered	coun
0	2011-01-01 00:00:00	1	0	0	1	9.84	14.395	81	0.0	3	13	1
1	2011-01-01 01:00:00	1	0	0	1	9.02	13.635	80	0.0	8	32	4
2	2011-01-01 02:00:00	1	0	0	1	9.02	13.635	80	0.0	5	27	3
3	2011-01-01 03:00:00	1	0	0	1	9.84	14.395	75	0.0	3	10	1
4	2011-01-01 04:00:00	1	0	0	1	9.84	14.395	75	0.0	0	1	

Next steps: Generate code with df New interactive sheet

Basic Analysis of Data

```
df.shape
```

```
(10886, 12)
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10886 entries, 0 to 10885
Data columns (total 12 columns):
 #   Column      Non-Null Count  Dtype  
---  ------      --------------  -----  
 0   datetime    10886 non-null  object 
 1   season      10886 non-null  int64  
 2   holiday     10886 non-null  int64  
 3   workingday  10886 non-null  int64  
 4   weather     10886 non-null  int64  
 5   temp        10886 non-null  float64
 6   atemp       10886 non-null  float64
 7   humidity    10886 non-null  int64  
 8   windspeed   10886 non-null  float64
 9   casual      10886 non-null  int64  
 10  registered  10886 non-null  int64  
 11  count       10886 non-null  int64  
dtypes: float64(3), int64(8), object(1)
memory usage: 1020.7+ KB
```

```
#Checking null values
print(df.isnull().sum())
```

```
datetime      0
season        0
holiday       0
workingday    0
weather       0
temp          0
atemp         0
humidity      0
windspeed     0
casual        0
registered    0
count         0
dtype: int64
```

```
#checking duplicates
df[df.duplicated()]
```

datetime	season	holiday	workingday	weather	temp	atemp	humidity	windspeed	casual	registered	count

Data Type Conversion

```
yulu_df=df.copy()
yulu_df['datetime']=pd.to_datetime(yulu_df['datetime'])
cat_col=['season','holiday','workingday','weather']
for col in cat_col:
  yulu_df[col]=yulu_df[col].astype('category')
```

Replacing Values

```
yulu_df['season'] = yulu_df['season'].cat.rename_categories({1: "spring",2: "summer",3: "fall",4: "winter"}
yulu_df['holiday']=yulu_df['holiday'].cat.rename_categories({1:"Holiday",0:"Non_Holiday"})
```

```
yulu_df['workingday']=yulu_df['workingday'].cat.rename_categories({1:"Workingday",0:"Nonworkingday"})
yulu_df['weather'] = yulu_df['weather'].cat.rename_categories({1: "Clear",2: "Mist",3: "Light Snow",4:"Heavy
```

```
yulu_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10886 entries, 0 to 10885
Data columns (total 12 columns):
 #   Column      Non-Null Count  Dtype         
---  ------      --------------  -----         
 0   datetime    10886 non-null  datetime64[ns]
 1   season      10886 non-null  category      
 2   holiday     10886 non-null  category      
 3   workingday  10886 non-null  category      
 4   weather     10886 non-null  category      
 5   temp        10886 non-null  float64       
 6   atemp       10886 non-null  float64       
 7   humidity    10886 non-null  int64         
 8   windspeed   10886 non-null  float64       
 9   casual      10886 non-null  int64         
 10  registered  10886 non-null  int64         
 11  count       10886 non-null  int64         
dtypes: category(4), datetime64[ns](1), float64(3), int64(4)
memory usage: 723.7 KB
```

Statistical summary

```
yulu_df.describe(include='number').T
```

	count	mean	std	min	25%	50%	75%	max
temp	10886.0	20.230860	7.791590	0.82	13.9400	20.500	26.2400	41.0000
atemp	10886.0	23.655084	8.474601	0.76	16.6650	24.240	31.0600	45.4550
humidity	10886.0	61.886460	19.245033	0.00	47.0000	62.000	77.0000	100.0000
windspeed	10886.0	12.799395	8.164537	0.00	7.0015	12.998	16.9979	56.9969
casual	10886.0	36.021955	49.960477	0.00	4.0000	17.000	49.0000	367.0000
registered	10886.0	155.552177	151.039033	0.00	36.0000	118.000	222.0000	886.0000
count	10886.0	191.574132	181.144454	1.00	42.0000	145.000	284.0000	977.0000

Observation

- The average temperature is 20.5°C, with a maximum of 41°C.
- The average atemp (feels-like temperature) is 24.24°C, with a maximum of 45.45°C.
- The average humidity is 62%, with a maximum of 100%.
- The ranges for casual, registered, and total bike rentals are quite high, indicating the presence of possible outliers.

```
yulu_df.describe(include='category')
```

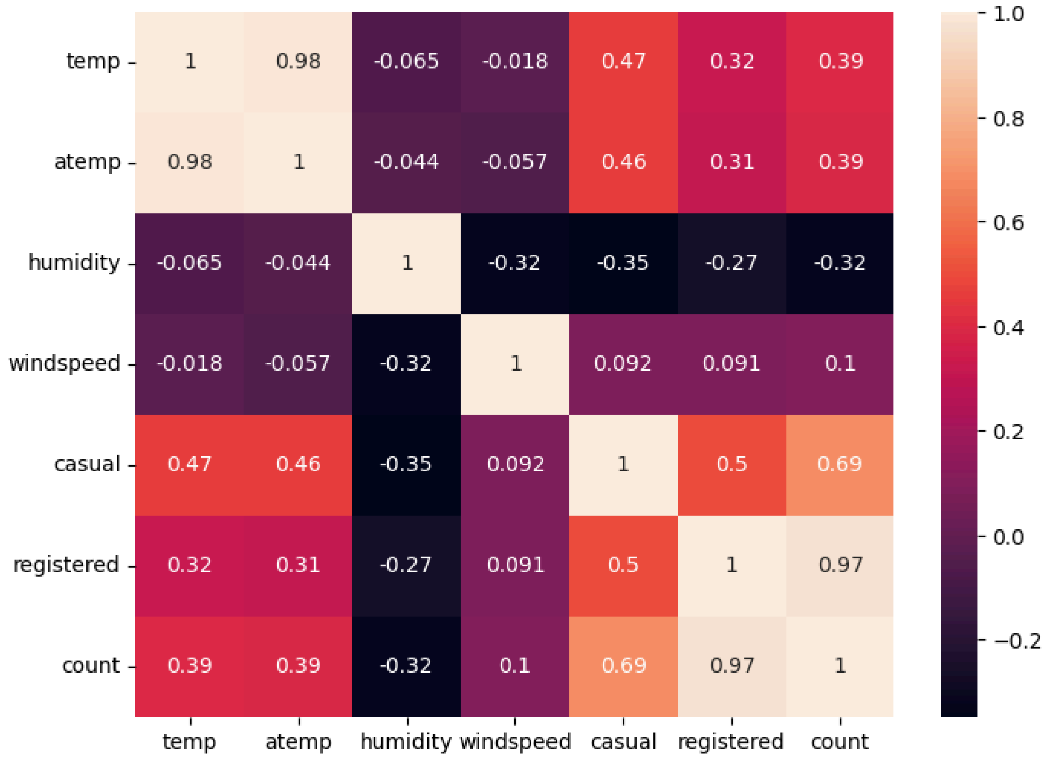
	season	holiday	workingday	weather
count	10886	10886	10886	10886
unique	4	2	2	4
top	winter	Non_Holiday	Workingday	Clear
freq	2734	10575	7412	7192

Observation

- More electric cycles are rented during Non_holidays.
- The clear weather occures more frequently than other weather conditions.
- During working day the more number of cycles are rented.

Correlation

```
cor_mat=yulu_df.select_dtypes(include='number').corr()
plt.figure(figsize=(8,6))
sns.heatmap(cor_mat,annot=True)
plt.show()
```



Observation

- **Humidity and windspeed** show a **negative correlation** with the total count of electric cycles rented. This suggests that **lower humidity and lower windspeed** are associated with **higher demand**.
- **Temperature, feels-like temperature (atemp), casual users, and registe**red users are all **positively correlated** with the rental count. So, when the weather is warmer, and there are more casual or registered users, are tend to increase the demand for cycle rentels.

Unique values

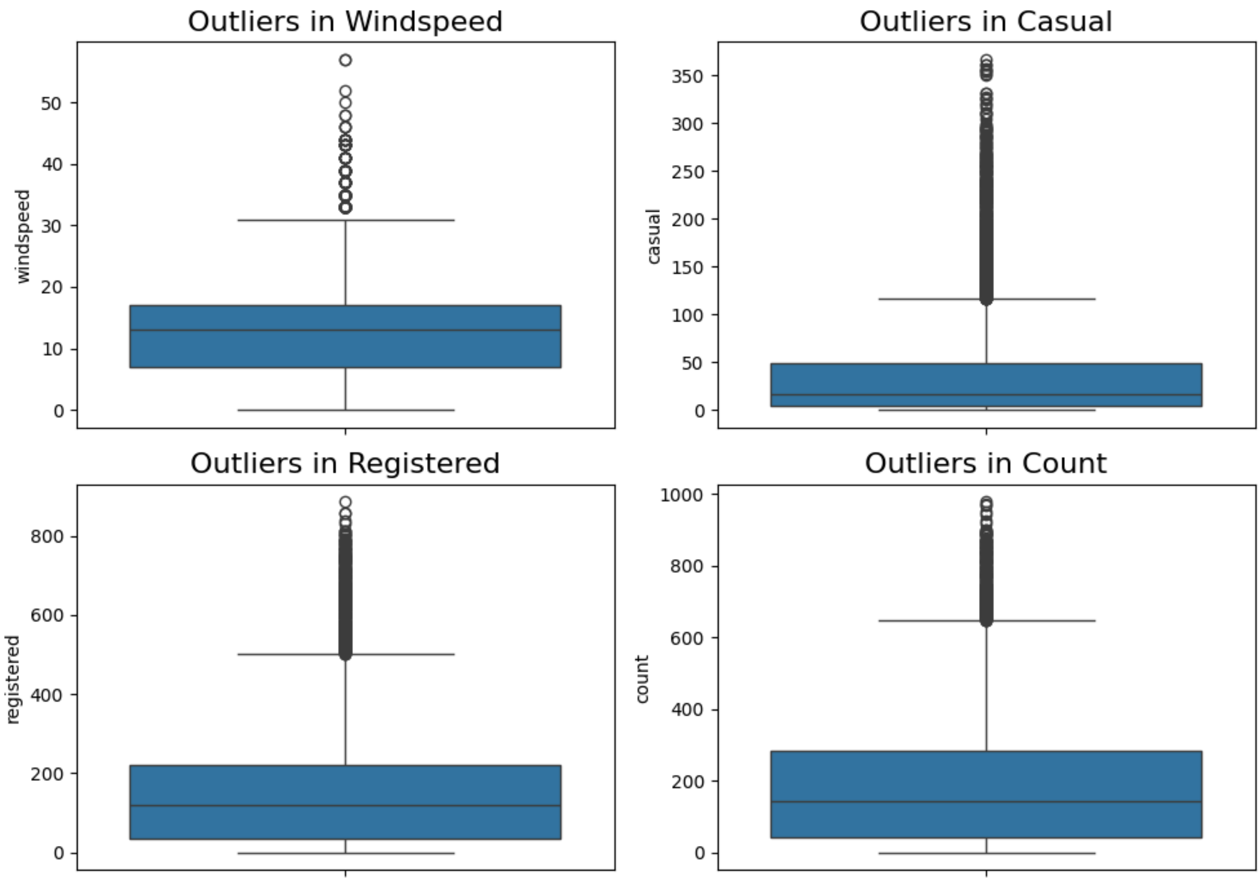
```
df.columns
for i in df.columns:
  print(i, ":", df[i].nunique())
```

```
datetime : 10886
season : 4
holiday : 2
workingday : 2
weather : 4
temp : 49
atemp : 60
humidity : 89
```

```
windspeed : 28
casual : 309
registered : 731
count : 822
```

Outlier Treatment

```
columns = ['windspeed', 'casual', 'registered', 'count']
plt.figure(figsize=(10,7))
for i, col in enumerate(columns,1):
    plt.subplot(2, 2, i)
    sns.boxplot(y=yulu_df[col])
    plt.title(f"Outliers in {col.capitalize()}",fontsize=16)
plt.tight_layout()
plt.show()
```




```
df_out=yulu_df[columns]
Q1=df_out.quantile(0.25)
Q3=df_out.quantile(0.75)
IQR=Q3-Q1
lower_whisker=Q1-1.5*IQR
upper_whisker=Q1+1.5*IQR
print(f"Lower whisker:\n{lower_whisker.round(2)}\nUpper whisker:\n{upper_whisker.round(2)}")
```

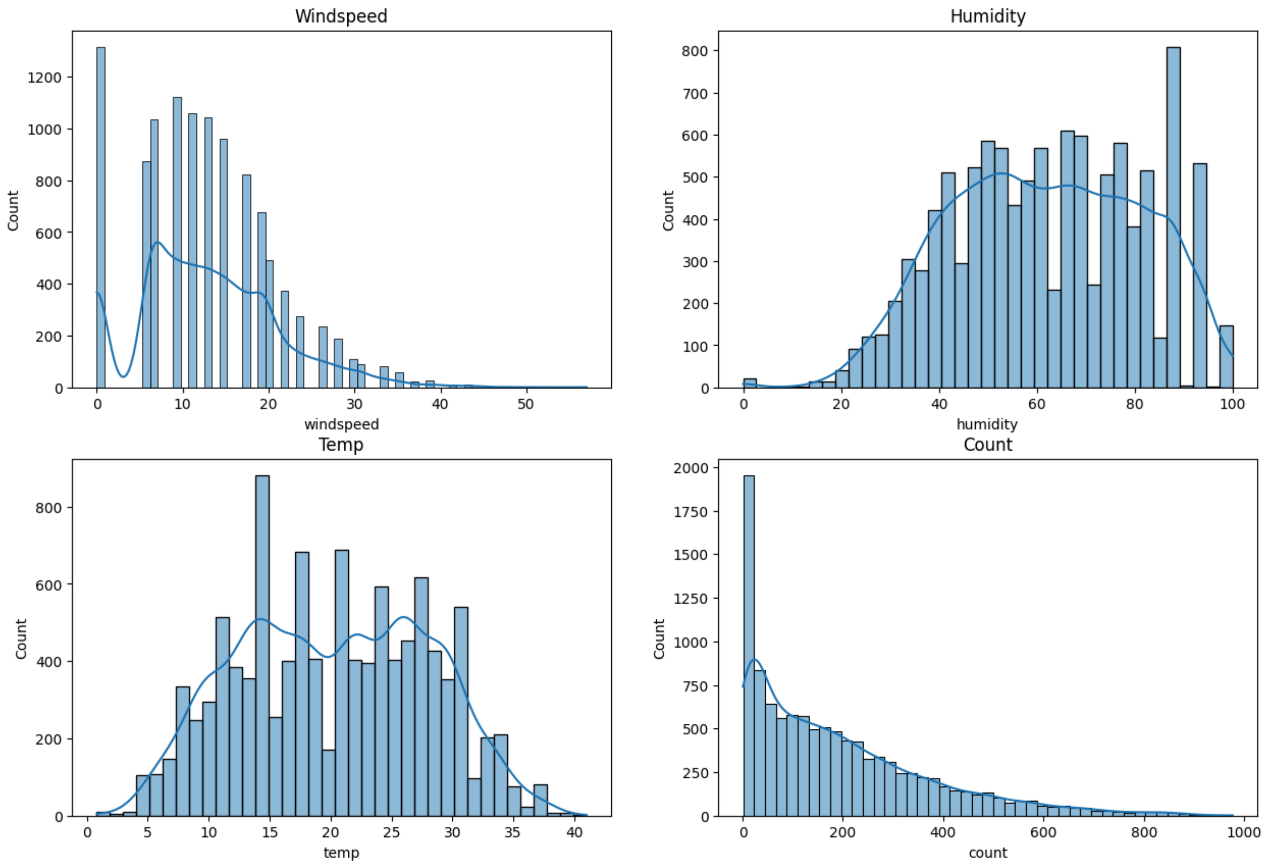
```
Lower whisker:
windspeed      -7.99
casual        -63.50
registered   -243.00
count        -321.00
dtype: float64
Upper whisker:
windspeed      22.0
casual         71.5
registered    315.0
count         405.0
dtype: float64
```

Observation

- There are many outliers present in the columns total bike rented count and registered total.
- When we visualize the data, outliers in the causual column appear more extremes compared to other column, but when we look at the actual data it shows only 71. The IQR also less compared to other variables.
- Windspend and casual have less outliers.
- Since these values represent the total number of bikes rented by customers, removing them would not be the right choice.

Univariate Analysis

```
uni_col=['windspeed','humidity','temp','count']
plt.figure(figsize=(15,10))
for i,col in enumerate(uni_col,1):
  plt.subplot(3, 2, i)
  sns.histplot(yulu_df[col],kde=True)
  plt.title(col.capitalize())
```

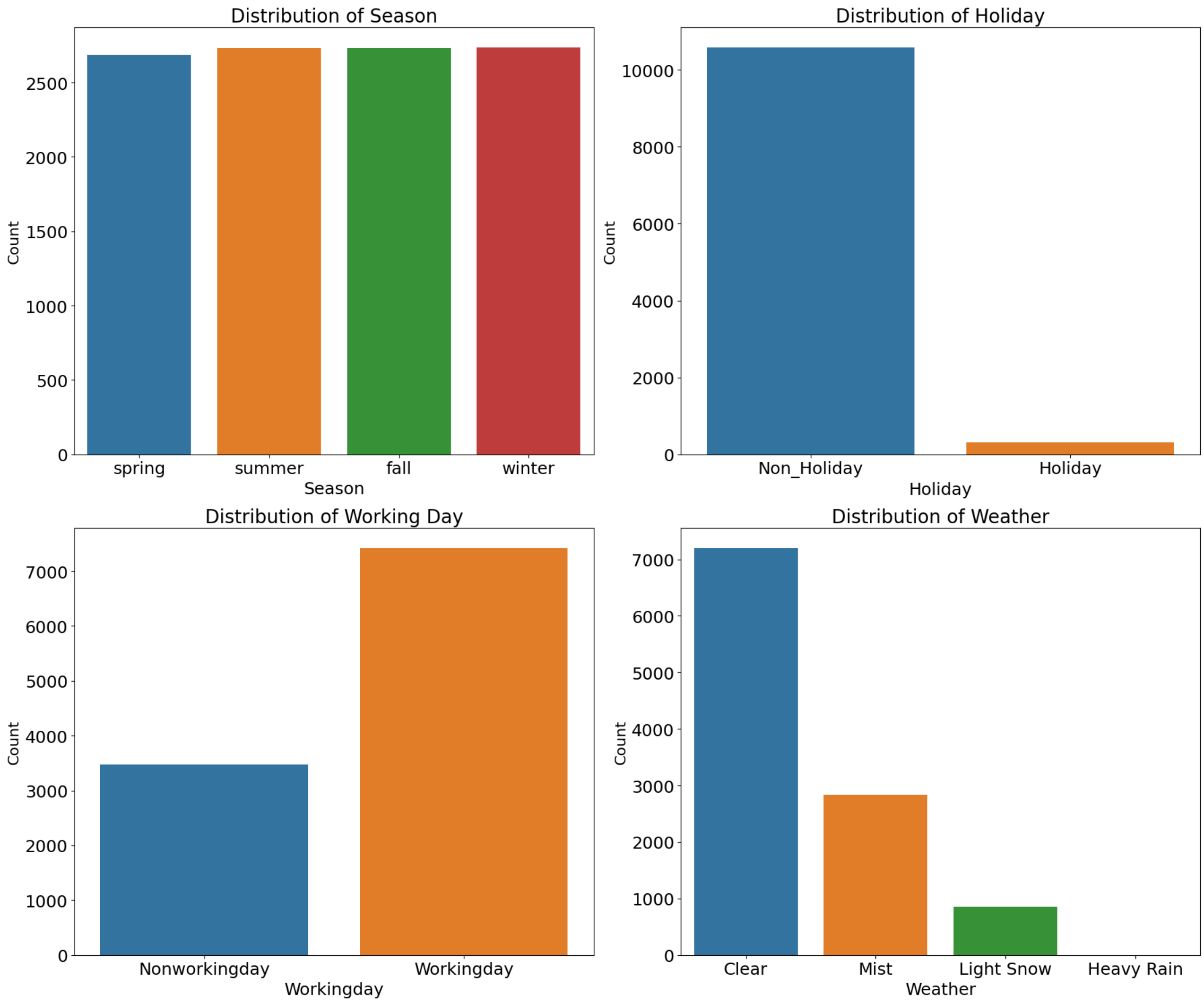


Insights

- Tempeture is roughly normally distributed.
- Humidity is left skewed, indicating most values are concentrated at higher levels.
- Although Windspeed is continuoes variable, its distribution looks similar to a geometric distribution, it is peak when it is zero(no wind) and starts decreasing.

Categorical Variables

```
fig, axes = plt.subplots(2, 2, figsize=(18, 15))
axes = axes.flatten()
columns = ['season', 'holiday', 'workingday', 'weather']
titles = ['Distribution of Season', 'Distribution of Holiday', 'Distribution of Working Day', 'Distribution
for i, col in enumerate(columns):
    sns.countplot(x=col, hue=col, data=yulu_df, ax=axes[i], palette='tab10', legend=False)
    axes[i].set_title(titles[i], fontsize=20)
    axes[i].set_ylabel('Count', fontsize=16)
    axes[i].set_xlabel(col.capitalize(), fontsize=18)
    axes[i].tick_params(axis='both', labelsize=18)
plt.tight_layout()
plt.show()
```

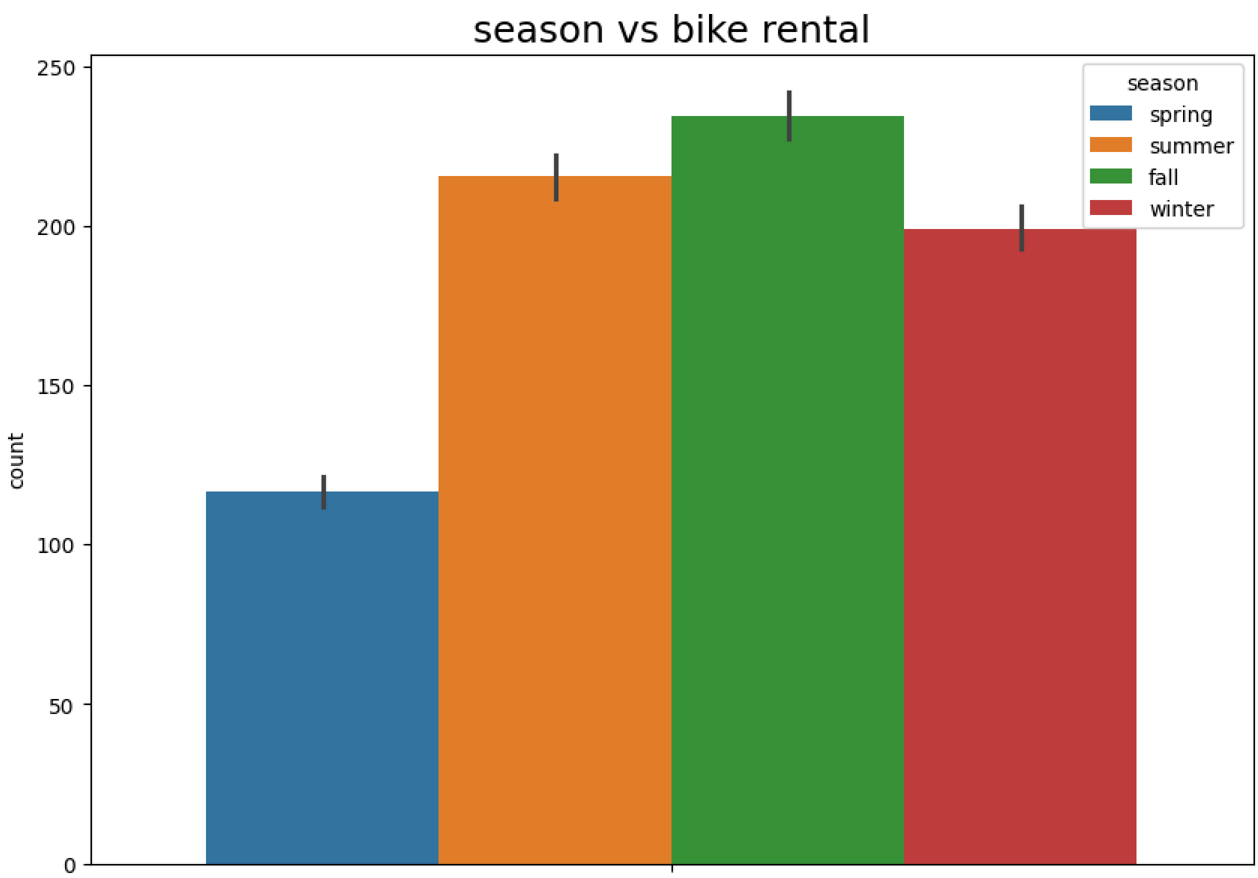


Insights

- The counts for all Seasons are identical.
- Working days have much higher counts compared to non-working days.
- Non Holiday count is very high while holiday is very less. Indicates demands on non holiday.
- Clear weather has high count compared to other weather conditions. while heavy rain is absent.

Bivariate Analysis

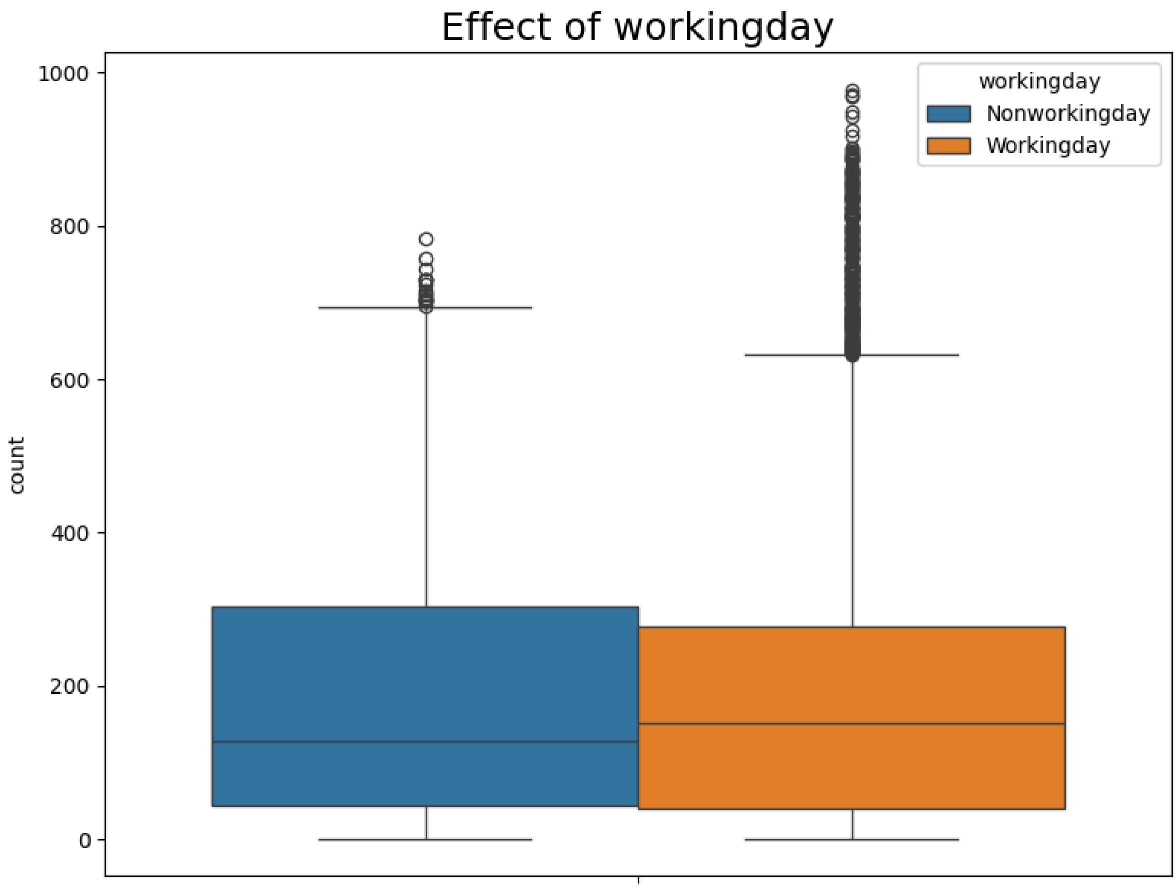
```
plt.figure(figsize=(10,7))
sns.barplot(y="count",hue='season',data=yulu_df)
plt.title("season vs bike rental",fontsize=18)
plt.show()
```



Insights

- Compared to other seasons, the average number of bikes rented in spring is much lower.
- The median number of bikes rented during the fall season is higher.
- A higher number of bikes are rented during spring and summer.

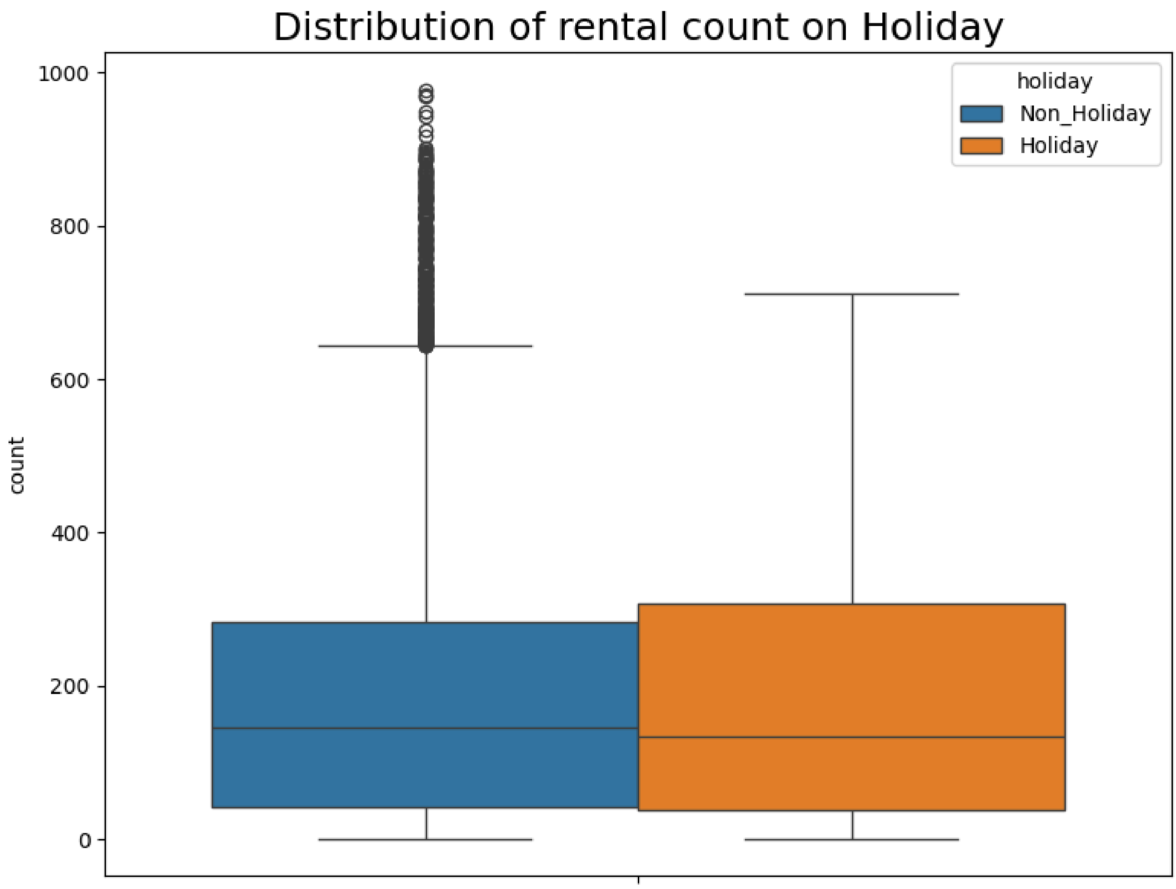
```
plt.figure(figsize=(9,7))
sns.boxplot(y="count",hue='workingday',data=yulu_df)
plt.title("Effect of workingday",fontsize=18)
plt.show()
```



Insights

- The median number of bikes rented on working day is higher compared to non working day.
- There are also some extreme high counts on working days compared to non-working days.
- Hence yulu bikes are more tend to be rented during working days.

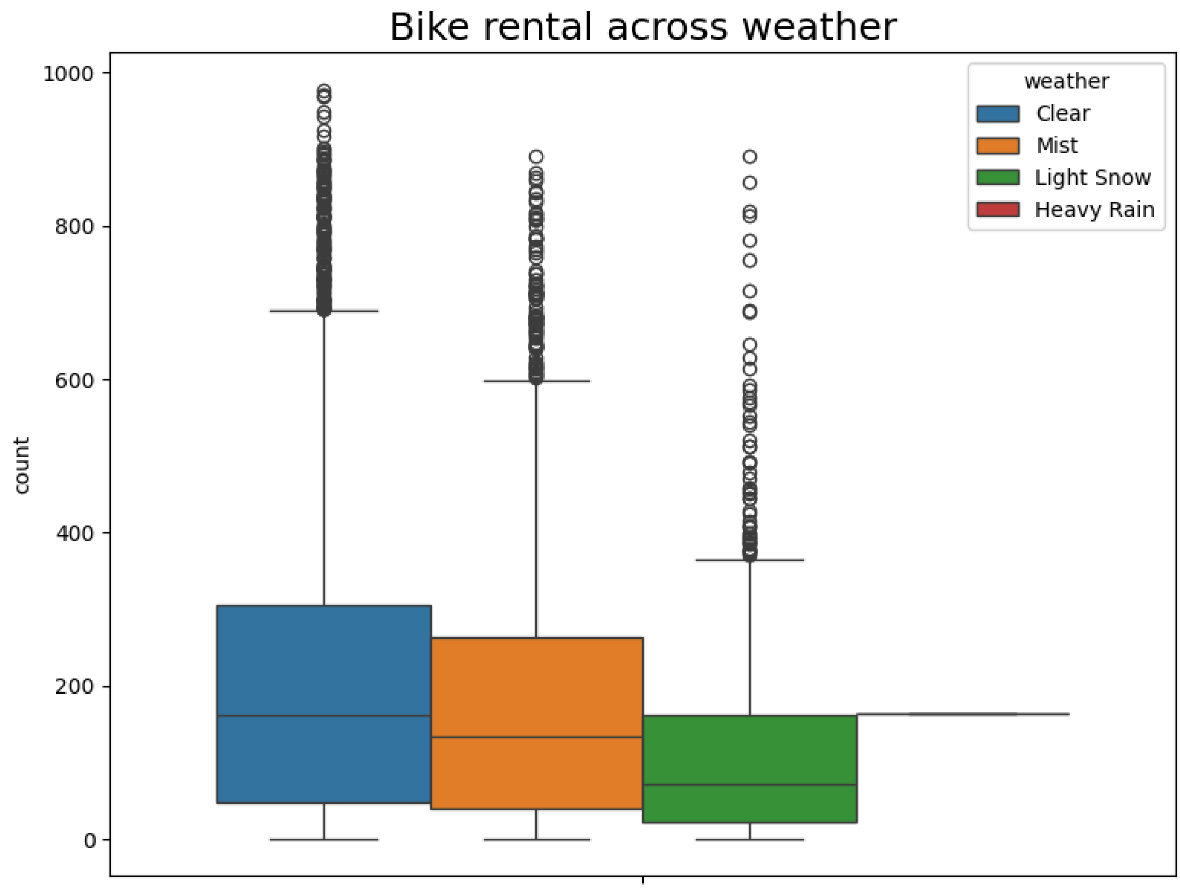
```
plt.figure(figsize=(9,7))
sns.boxplot(hue='holiday',y='count',data=yulu_df)
plt.title("Distribution of rental count on Holiday",fontsize=18)
plt.show()
```



Insights

- The median rental count is quite similar for both holidays and non-holidays.
- During non holidays, the rental count is inconsistent, showing unusually high usage.
- The box plot indicates higher bike rental counts on non holidays. Therefore the demand for electric bike rental is high during non holidays.

```
plt.figure(figsize=(9,7))
sns.boxplot(hue='weather',y='count',data=yulu_df)
plt.title("Bike rental across weather",fontsize=18)
plt.show()
```



Insights

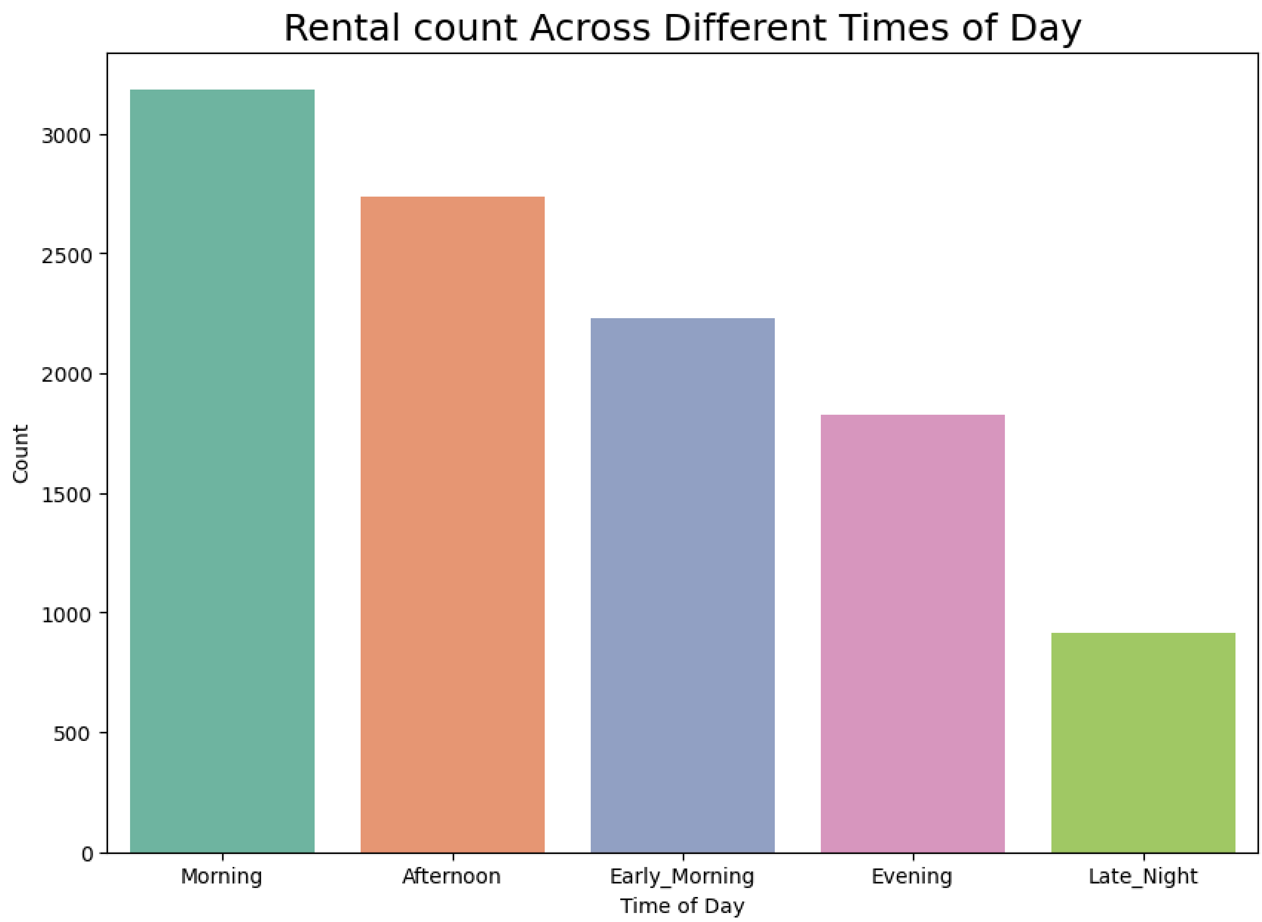
- When the weather is clear, the median bike rental count is very high compared to other weather conditions.
- When there is a light snow, the number of bike rentals is very low.
- During rain, only single record is present in the dataset.
- The demand for renting bikes is higher when the weather is good and predictable.
- heavy rain leads to complete decline in rentals.

How Hourly Trends Influence Bike Rentals

```
hour = yulu_df['datetime'].dt.hour
def get_time_of_day(hour):
    if 5 <= hour <= 11:
        return 'Morning'
    elif 12 <= hour <= 17:
        return 'Afternoon'
    elif 18 <= hour <= 21:
        return 'Evening'
    elif 22 <= hour <= 23:
        return 'Late_Night'
    elif 0 <= hour <= 4:
        return 'Early_Morning'
    else:
        return 'Unknown'

yulu_df['time_of_day'] = hour.apply(get_time_of_day)
time_df = yulu_df['time_of_day'].value_counts().reset_index()
time_df.columns = ['time_of_day', 'count']
plt.figure(figsize=(10,7))
sns.barplot(x='time_of_day',y='count',hue='time_of_day',data=time_df,palette='Set2',legend=False)
```

```
plt.xlabel("Time of Day")
plt.ylabel("Count")
plt.title("Rental count Across Different Times of Day",fontsize=18)
plt.show()
```

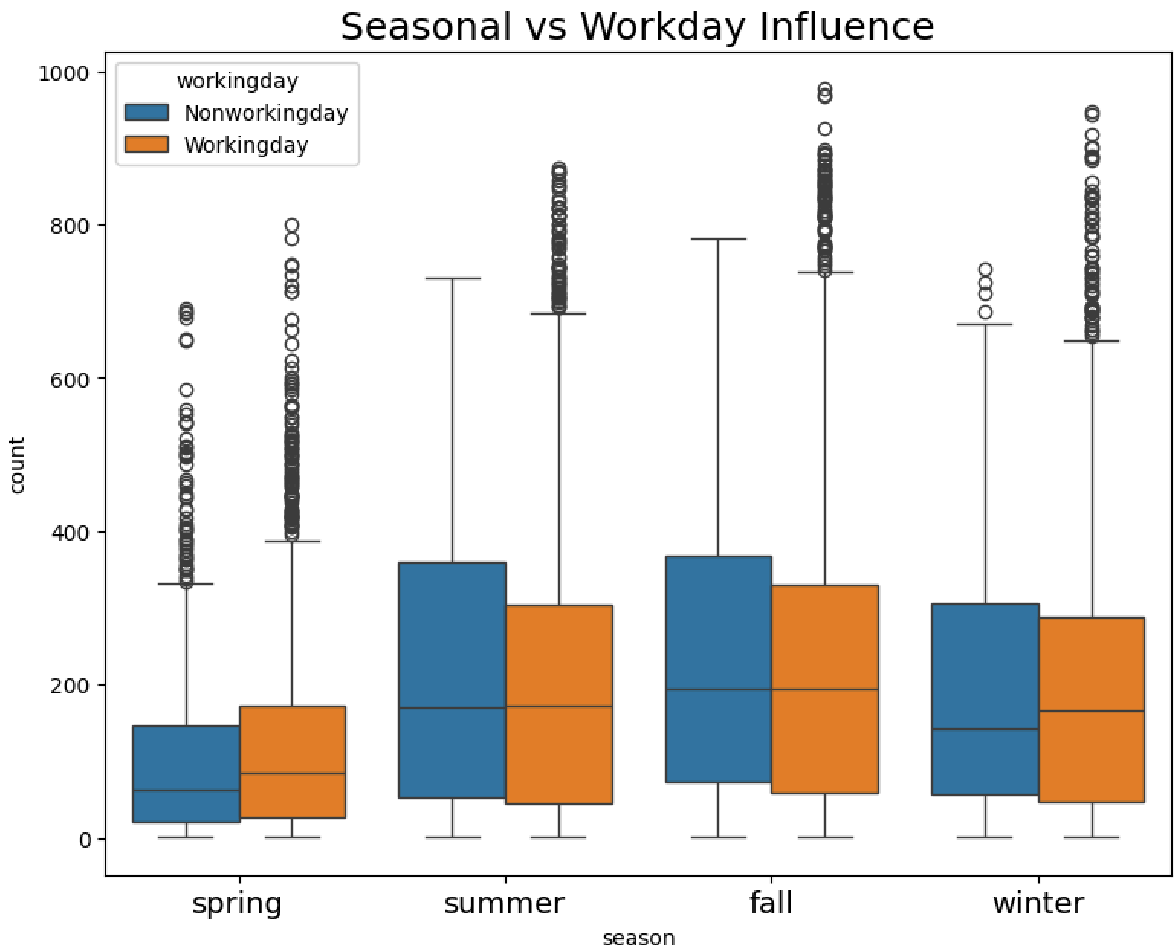


Insights

- More cycles are rented during the morning time, followed by the afternoon.
- This shows that demand for electric cycles peaks in the morning hours and remains high in the afternoon.

Seasonal and Working Day Effects on Rental Counts

```
plt.figure(figsize=(9,7))
sns.boxplot(x='season',y='count',hue='workingday',data=yulu_df)
plt.title('Seasonal vs Workday Influence',fontsize=18)
plt.xticks(fontsize=14)
plt.show()
```



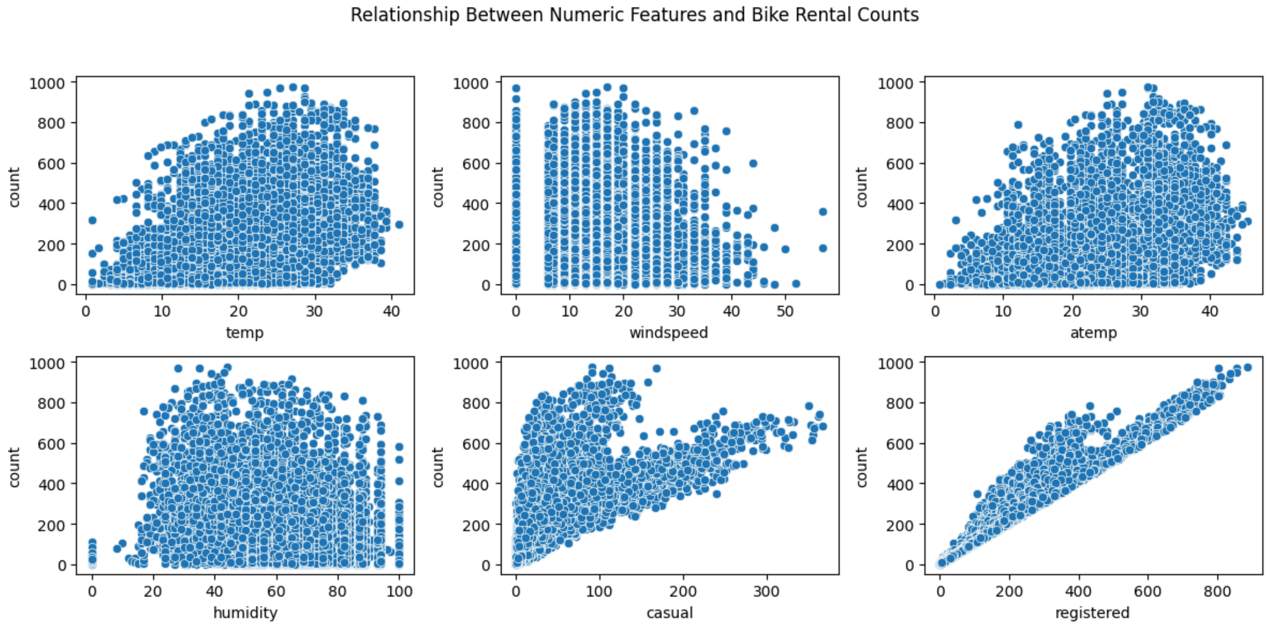
Insights

- During the spring, the median number of bikes rented on working day is higher compared to non working day.
- The median number of bikes rented in summer and fall is the same for both working days and non-working days.
- Hence the median number of bikes rented is evenly distributed across all the seasons regardless whether its a working day or not.

Exploring Correlations Between Features and Bike Rental Counts

```
num_cal=['temp', 'windspeed', 'atemp', 'humidity','casual','registered']
fig,axes=plt.subplots(2,3, figsize=(12,6))
i=0
for row in range(2):
  for col in range(3):
      sns.scatterplot(x=num_cal[i],y='count',data=yulu_df,ax=axes[row, col])
      i += 1

fig.suptitle('Relationship Between Numeric Features and Bike Rental Counts')
plt.tight_layout(rect=[0, 0, 1, 0.95])
plt.show()
```



Insights

- The total number of bikes rented is generally high when the temperature is between 20 to 30°C.
- There is a negative correlation between windspeed and bike rentals. Specifically, when the windspeed is less than 10, the number of bikes rented tends to be very high.
- When humidity is below 20%, the bike rental count is noticeably low.
- Therefore, windspeed is a notable factor affecting demand, with lower windspeed encouraging more bike rentals.
- A strong linear correlation, indicating that registered users are consistent contributors to bike rental counts.

Hypothesis Testing

Is there any significant effect of Working Day on number of electric cycles rented?

```
yulu_df.groupby('workingday',observed=True)['count'].describe()
```

workingday	count	mean	std	min	25%	50%	75%	max
0	3474.0	188.506621	173.724015	1.0	44.0	128.0	304.0	783.0
1	7412.0	193.011873	184.513659	1.0	41.0	151.0	277.0	977.0

STEP-1: Set up Null Hypothesis

- **Null Hypothesis (Ho):** Working day has no effect on the number of electric cycles rented.
- **Alternative Hypothesis (Ha):** Working day has an effect on the number of electric cycles rented.

STEP-2: Check assumptions for the test

- QQ plots to check for normality of the data distributions.
- Levene's test to check for homogeneity of variances between groups.

STEP-3: Set the Significance Level (α).

STEP-4: Calculate the p_value.

STEP-5: Interpret the Results.

Normality test using QQ plot

```
samp_1=yulu_df[yulu_df['workingday']=="Workingday"]["count"]
samp_2=yulu_df[yulu_df['workingday']=="Nonworkingday"]["count"]
```

```
samp_1
```

	count
47	5
48	2
49	1
50	3
51	30
...	...
10881	336
10882	241
10883	168
10884	129
10885	88

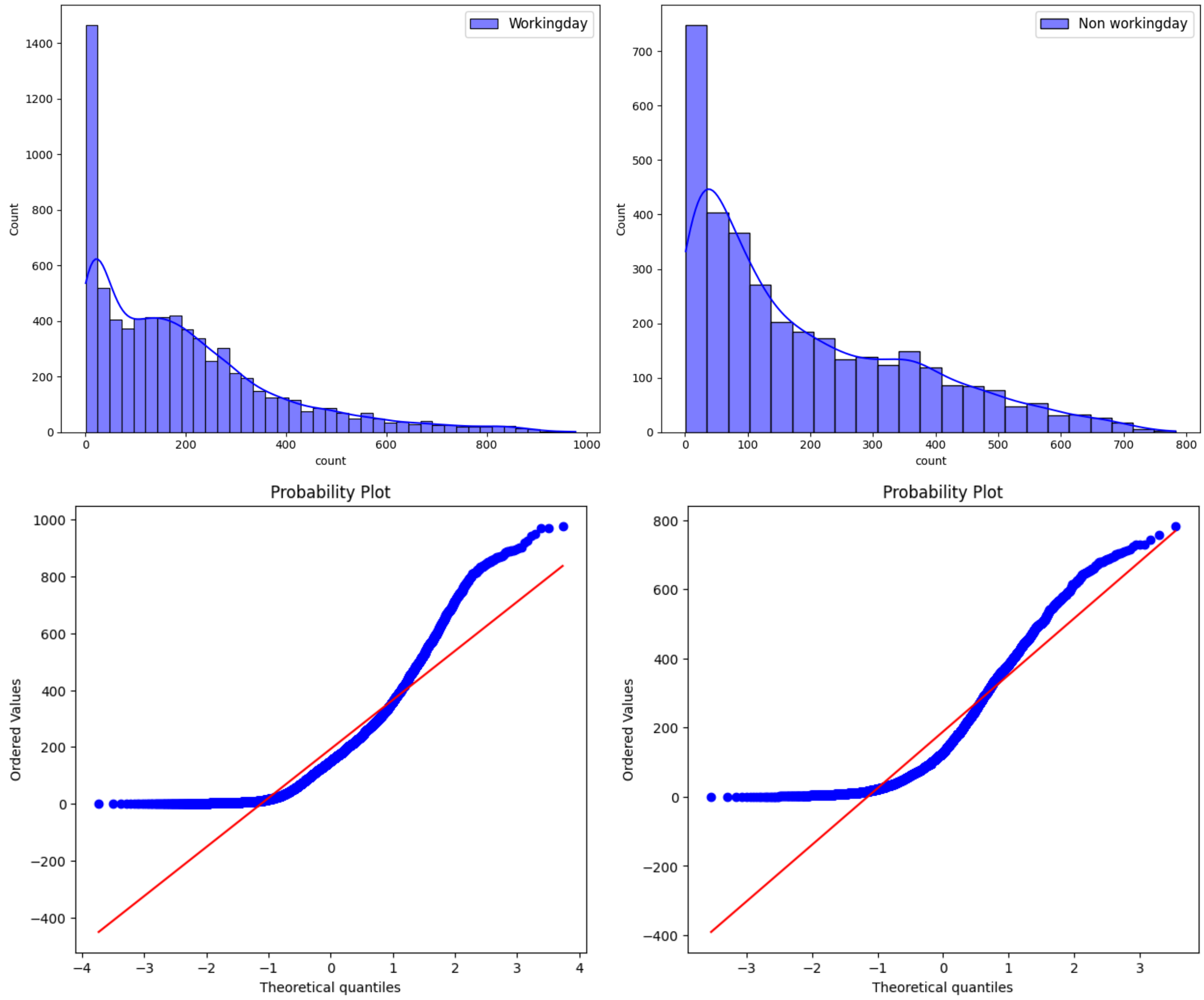
7412 rows × 1 columns

dtype: int64

```
plot_vars = [samp_1, samp_2]
labels = ['Workingday', 'Non workingday']

fig, axes = plt.subplots(1, 2, figsize=(15, 6))

for i, val in enumerate(plot_vars):
    sns.histplot(val, kde=True, ax=axes[i], label=labels[i],color='blue')
    axes[i].legend(fontsize=12)
plt.tight_layout()
plt.show()
fig, axes = plt.subplots(1, 2, figsize=(15, 6))
for i, val in enumerate(plot_vars):
    stats.probplot(val,dist='norm',plot=axes[i])
plt.show()
```



Observation

The observed quantiles deviate from the straight line in the Q-Q plot, and the histogram shows a longer tail on the right side stretched more on the right side. It indicates that the data is right skewed and not normally distributed.

Homogeneity of Variances using Levene's test

```
leve_stat,p_value=stats.levene(samp_1,samp_2)
alpha=0.05
print("alpha:",alpha,"\nleve_stat :", leve_stat,"\np_value :", p_value)
if p_value > alpha:
  print("Fail to Reject Null Hypothesis")
  print("The variance are equal.")
else:
  print("Reject Null Hypothesis:")
  print("The variance are not equal.")
```

```
alpha: 0.05
leve_stat : 0.004972848886504472
p_value : 0.9437823280916695
Fail to Reject Null Hypothesis
The variance are equal.
```

Since the Variance are equal we can use independence T test.

Independence T-Test

```
alpha=0.05
t_statistic,p_value = stats.ttest_ind(samp_1,samp_2)
print("alpha:",alpha,"\nt_statistic :", t_statistic,"\np_value :", p_value)
alpha=0.05
if p_value > alpha:
  print("Fail to Reject Null Hypothesis")
  print("Working day has no effect on the number of electric cycles rented.")
else:
  print("Reject Null Hypothesis: Working day has an effect on the number of electric cycles rented")
```

```
alpha: 0.05
t_statistic : 1.2096277376026694
p_value : 0.22644804226361348
Fail to Reject Null Hypothesis
Working day has no effect on the number of electric cycles rented.
```

Conclusion

Since p_value is greater than alpha we fail to reject the Null hypothesis.Therefore, Workingday has no effect on the number of electric cycles rented.

Validating T-Test Results Using the Mann–Whitney U Test

```
m1_samp = yulu_df[yulu_df['workingday']=='Workingday']['count']
m2_samp = yulu_df[yulu_df['workingday']=='Nonworkingday']['count']
```

Framimg Hypothesis

Null hypothesis: Working day has no effect on the number of electric cycles rented.

Alternative hypothesis: Working day has an effect on the number of electric cycles rented.

```
from scipy.stats import mannwhitneyu
alpha = 0.05
stat,p = mannwhitneyu(m1_samp,m2_samp)
print('alpha: {}\nstat: {:.2f}\np-value: {:.2f}'.format(alpha, stat, p))
if p > alpha:
  print('Fail to reject null hypothesis')
  print('Working day has no effect on the number of electric cycles rented')
else:
  print('Reject null hypothesis')
  print('Working day has an effect on the number of electric cycles rented')
```

```
alpha: 0.05
stat: 12868495.50
p-value: 0.97
Fail to reject null hypothesis
Working day has no effect on the number of electric cycles rented
```

Conclusion

Even though the data is non-normal, the t-test result was correct in this case because the sample sizes are large, and the Mann–Whitney U test confirms it.

Is there a difference in the number of cycles rented across weather?

```
yulu_df.groupby('weather',observed=True)['count'].describe()
```

weather	count	mean	std	min	25%	50%	75%	max
Clear	7192.0	205.236791	187.959566	1.0	48.0	161.0	305.0	977.0
Heavy Rain	1.0	164.000000	NaN	164.0	164.0	164.0	164.0	164.0
Light Snow	859.0	118.846333	138.581297	1.0	23.0	71.0	161.0	891.0
Mist	2834.0	178.955540	168.366413	1.0	41.0	134.0	264.0	890.0

```
wea_1=yulu_df[yulu_df['weather']=='Clear']['count']
wea_2=yulu_df[yulu_df['weather']=='Mist']['count']
wea_3=yulu_df[yulu_df['weather']=='Light Snow']['count']
wea_4=yulu_df[yulu_df['weather']=='Heavy_Rain']['count']
```

STEP 1: Set up Null Hypothesis

- **Null Hypothesis (Ho)**: The number of cycles rented is similar across different weather conditions.
- **Alternative Hypothesis (Ha)**: The number of cycles rented differs significantly based on weather conditions.

STEP-2: Check assumptions for the test

- QQ plots to check for normality of the data distributions.
- Levene's test to check for homogeneity of variances between groups.

STEP-3: Set the Significance Level (α)

STEP-4: Calculate the p_value

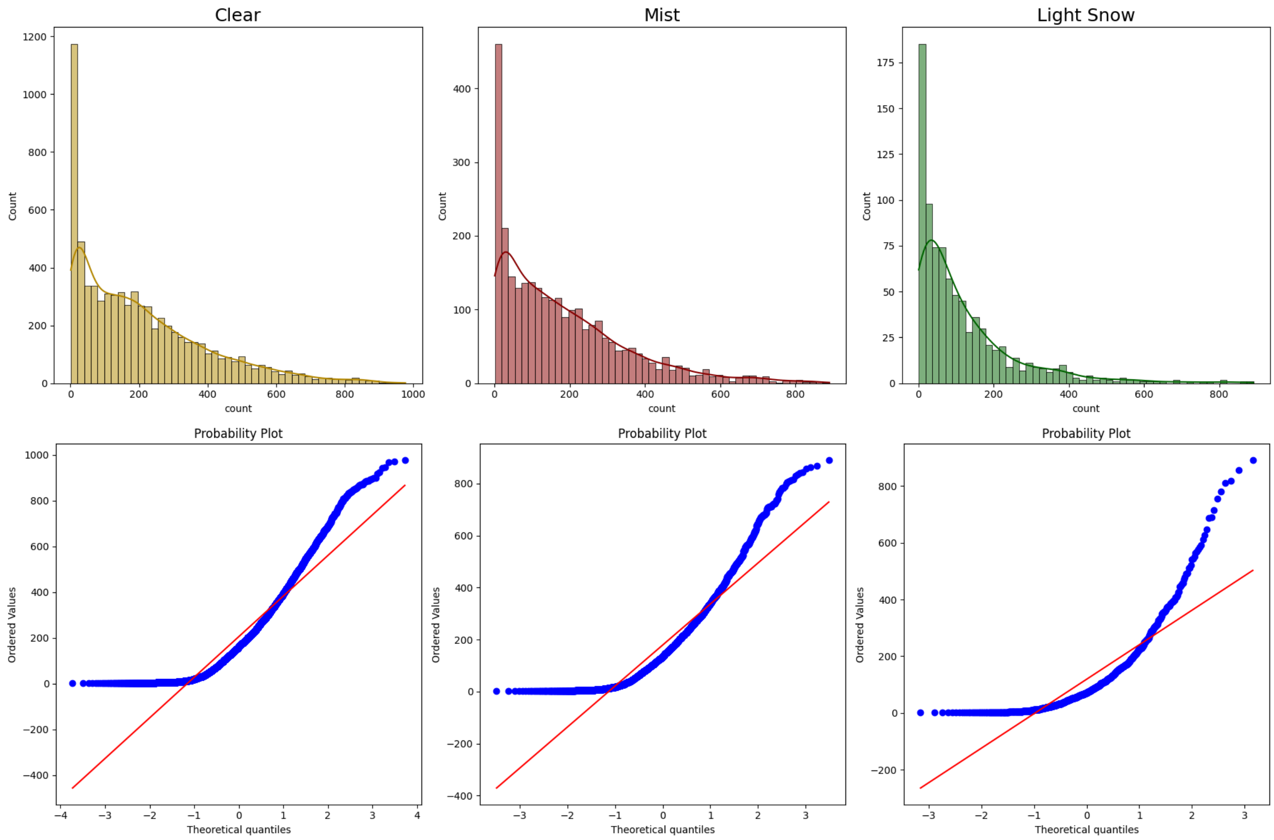
STEP-5: Interpret the Results

Normality test using QQ plot

```
weather = [wea_1, wea_2, wea_3]
labels = ["Clear", "Mist", "Light Snow"]
colors =  ["#b58900", "#8b0000", "#006400"]

# Histograms with KDE
fig, axes = plt.subplots(1, 3, figsize=(18, 6))
for ax, data, label, color in zip(axes.flat, weather, labels, colors):
    sns.histplot(data, kde=True, bins=50, ax=ax, color=color)
    ax.set_title(label,fontsize=18)
plt.tight_layout()
plt.show()

# Q-Q Plots
fig, axes = plt.subplots(1, 3, figsize=(18, 6))
for ax, data, label in zip(axes.flat, weather, labels):
    stats.probplot(data, dist="norm", plot=ax)
plt.tight_layout()
plt.show()
```



Obeservation

From the above plots we can clearly see that the data is right skewed and not normally distributed.

Homogeneity of Variances using Levene's test

```
leve_stat,p_value=stats.levene(wea_1,wea_2,wea_3)
alpha=0.05
print("alpha:",alpha,"\nleve_stat :", leve_stat,"\np_value :", p_value)
if p_value > alpha:
  print("Fail to Reject Null Hypothesis")
  print("The variance are equal across all weather conditions.")
else:
  print("Reject Null Hypothesis:")
  print("Atleast one season has different variance.")
```

```
alpha: 0.05
leve_stat : 81.67574924435011
p_value : 6.198278710731511e-36
Reject Null Hypothesis:
Atleast one season has different variance.
```

Since the assumption of homogeneity of variances (equal variances across groups) is violated, we cannot proceed with ANOVA. Instead, we use the Kruskal-Wallis test.

Kruskal-Wallis test

```
stat,p_value=stats.kruskal(wea_1,wea_2,wea_3)
alpha=0.05
print("alpha:",alpha,"\nstat :", stat,"\np_value :", p_value)
if p_value > alpha:
  print("Fail to Reject Null Hypothesis")
  print("The number of cycles rented is similar across different weather conditions")
else:
```

```
    print("Reject Null Hypothesis")
    print("The number of cycles rented differs significantly based on weather conditions")
```

```
alpha: 0.05
stat : 204.95566833068537
p_value : 3.122066178659941e-45
Reject Null Hypothesis
The number of cycles rented differs significantly based on weather conditions
```

Conclusion

Since p_value is less than alpha value we reject the Null hypothesis. Therefore,The number of cycles rented differs significantly based on weather conditions.

Is there a difference in the number of cycles rented across seasons?

```
yulu_df.groupby('season',observed=True)['count'].describe()
```

season	count	mean	std	min	25%	50%	75%	max
fall	2733.0	234.417124	197.151001	1.0	68.0	195.0	347.0	977.0
spring	2686.0	116.343261	125.273974	1.0	24.0	78.0	164.0	801.0
summer	2733.0	215.251372	192.007843	1.0	49.0	172.0	321.0	873.0
winter	2734.0	198.988296	177.622409	1.0	51.0	161.0	294.0	948.0

STEP-1 : Set up Null Hypothesis

- **Null Hypothesis (Ho)**: The number of cycles rented is similar across different seasons.
- **Alternate Hypothesis (Ha)**: The number of cycles rented differs significantly based on different seasons.

STEP-2: Define Test statistics

- Since we are comparing multiple independent groups, ANOVA test will be appropriate.

STEP-3: Checking assumpitons of the test

- Normality test using QQ plot
- Homegenety of variance

STEP-4: Calculate the p_value

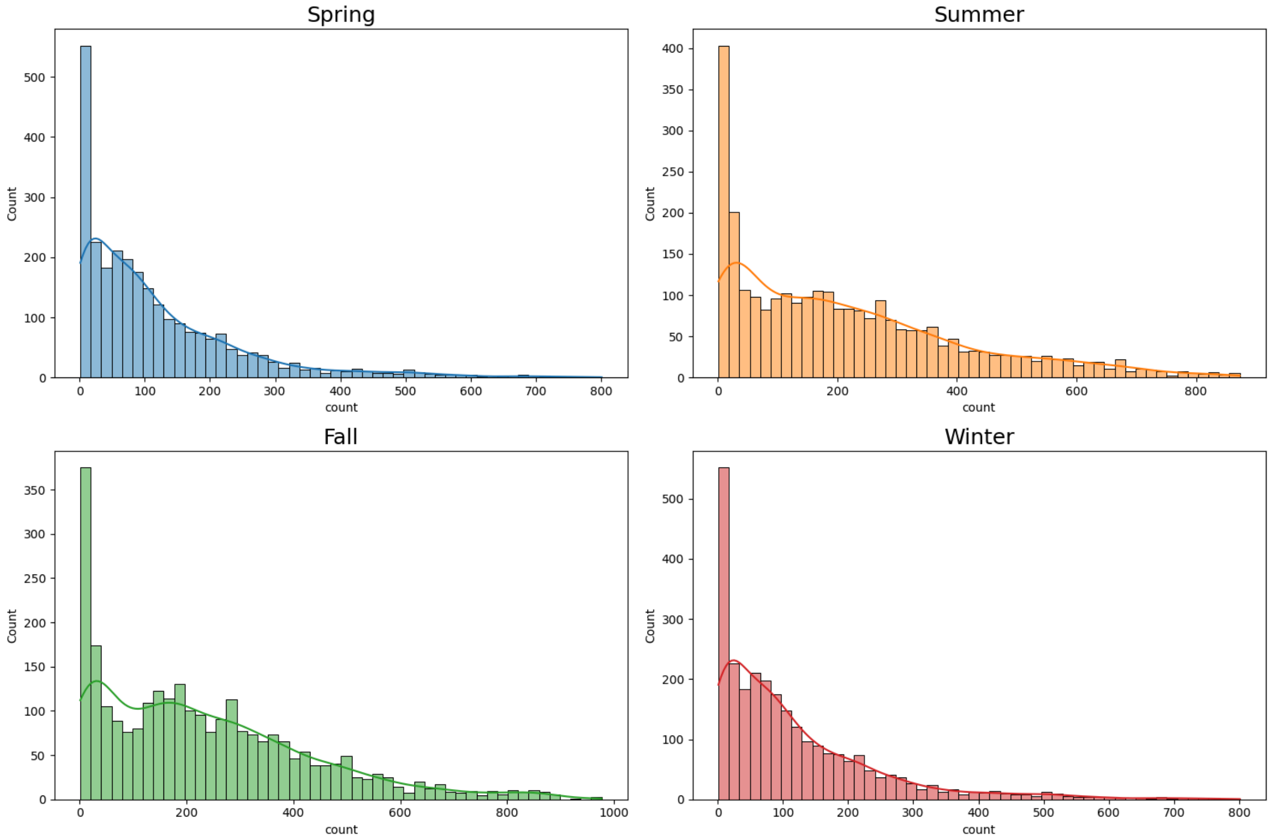
STEP-5: Interpret the Results

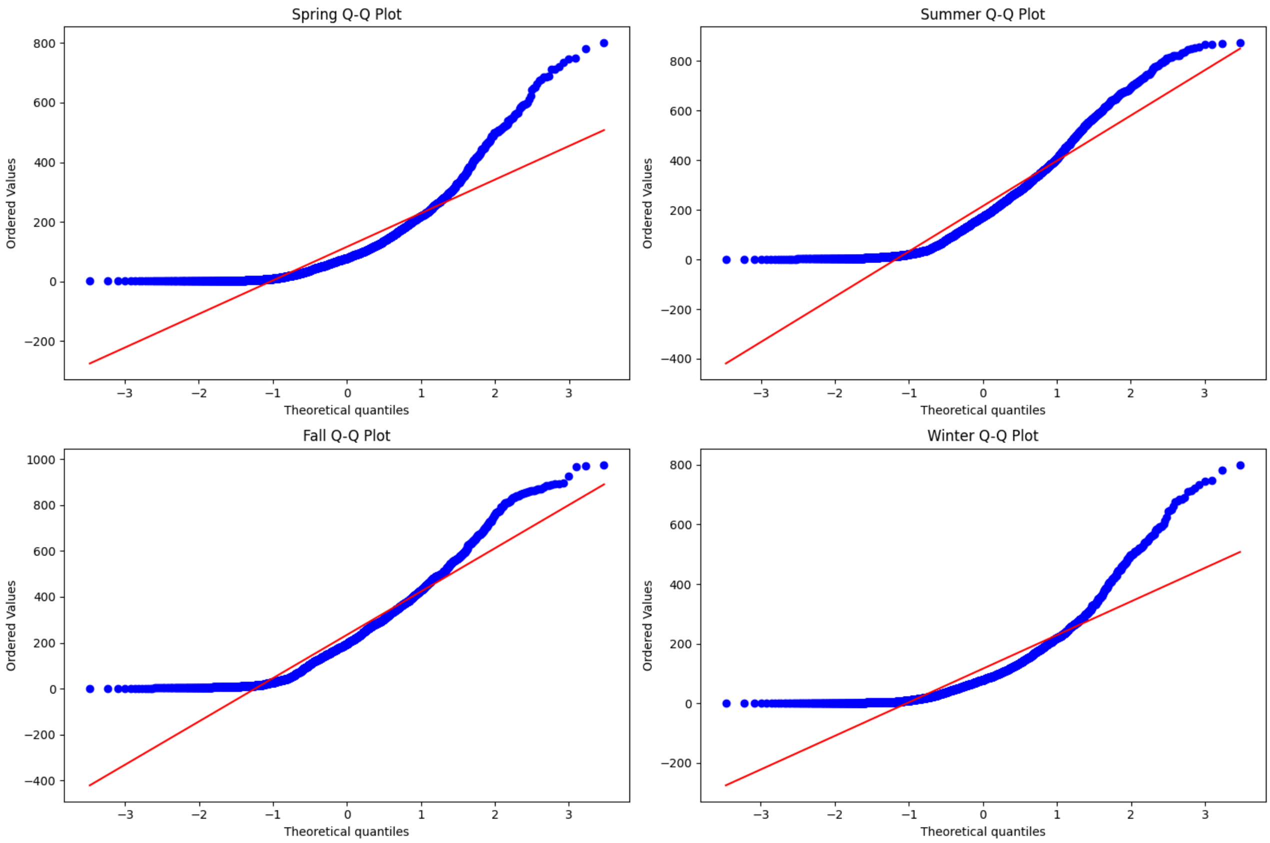
```
spring=yulu_df[yulu_df['season']=='spring']['count']
summer=yulu_df[yulu_df['season']=='summer']['count']
fall=yulu_df[yulu_df['season']=='fall']['count']
winter=yulu_df[yulu_df['season']=='spring']['count']
```

Normality test using QQ plot

```
seasons = [spring, summer, fall, winter]
labels = ["Spring", "Summer", "Fall", "Winter"]
colors = ["#1f77b4", "#ff7f0e", "#2ca02c", "#d62728"]
# Histograms with KDE
fig, axes = plt.subplots(2, 2, figsize=(15, 10))
for ax, season, label, color in zip(axes.flat, seasons, labels, colors):
    sns.histplot(season, kde=True, bins=50, ax=ax, color=color)
    ax.set_title(label,fontsize=18)
plt.tight_layout()
plt.show()
```

```
# Q-Q Plots
fig, axes = plt.subplots(2, 2, figsize=(15, 10))
for ax, season, label in zip(axes.flat, seasons, labels):
    stats.probplot(season, dist="norm", plot=ax)
    ax.set_title(f"{label} Q-Q Plot")
plt.tight_layout()
plt.show()
```





Observation

It can be inferred that the data is not normaly distributed.

Homogeneity of Variances using Levene's test

```
leve_stat,p_value=stats.levene(spring,summer,fall,winter)
alpha=0.05
print("alpha:",alpha,"\nleve_stat :", leve_stat,"\np_value :", p_value)
if p_value > alpha:
  print("Fail to Reject Null Hypothesis")
  print("The variance are equal across all the seasons.")
else:
  print("Reject Null Hypothesis:")
  print("Atleast one season has different variance.")
```

```
alpha: 0.05
leve_stat : 305.0383395048151
p_value : 3.969591085599068e-190
Reject Null Hypothesis:
Atleast one season has different variance.
```

Since the assumption of homogeneity of variances (equal variances across groups) is violated, we cannot proceed with ANOVA. Instead, we use the Kruskal-Wallis test.

Kruskal-Wallis Test

```
stat,p_value=stats.kruskal(spring,summer,fall,winter)
alpha=0.05
print("alpha:",alpha,"\nstat :", stat,"\np_value :", p_value)
if p_value > alpha:
  print("Fail to Reject Null Hypothesis")
  print("The number of cycles rented is similar in different seasons")
else:
  print("Reject Null Hypothesis")
  print("The number of cycles rented differs significantly based on different seasons.")
```

```
alpha: 0.05
stat : 998.0818170635649
p_value : 4.690777043783735e-216
Reject Null Hypothesis
The number of cycles rented differs significantly based on different seasons.
```

Conclusion

Since the p value is less than alpha, we reject the null hypothesis. Therefore, we can conclude that the total number of bikes rented is differs based seasons.

Is weather dependent on the season?

STEP-1 : Set up Null and Alternative Hypothesis

- **Null Hypothesis (Ho):** Season and weather are not associated.
- **Alternative Hypothesis (Ha):** Season and weather are associated.

STEP-2: Define Test statistics

- Since both features are categorical, the Chi-Square test will be appropriate.

STEP-3: Set a significance level (alpha) = 0.05

```
chi_cross=pd.crosstab(yulu_df['weather'],yulu_df['season'])
chi_cross
```

season	spring	summer	fall	winter
weather				
Clear	1759	1801	1930	1702
Mist	715	708	604	807
Light Snow	211	224	199	225
Heavy Rain	1	0	0	0

Next steps: Generate code with chi_cross New interactive sheet

STEP-4: Calculating test Statistics (p_value)

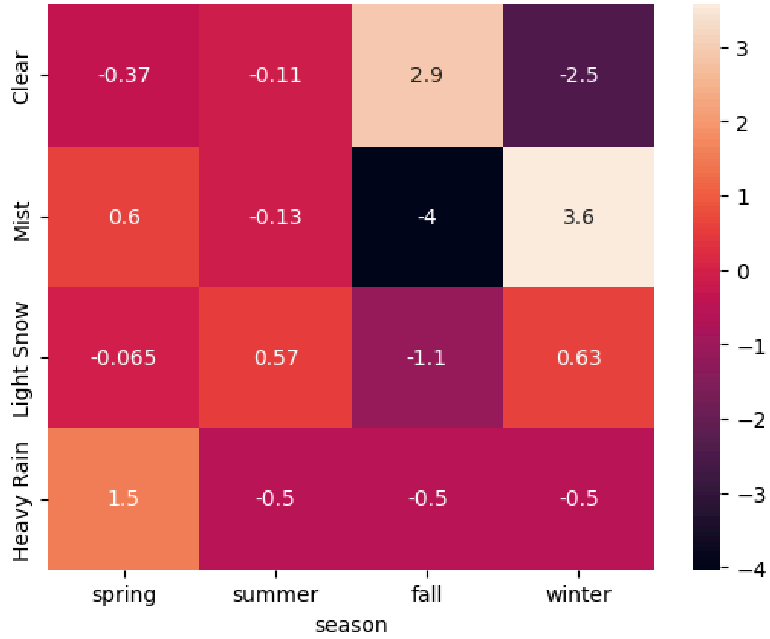
```
chi_statistic,p_value,dof, expected = stats.chi2_contingency(chi_cross)
alpha = 0.05
print("chi_statistic :",chi_statistic,"\np_value : ", p_value,"\ndof : ", dof, "\nexpected : ", expected)
if p_value > alpha:
  print("Fail to Reject Null Hypothesis.")
  print("Season and weather are not associated.")
else:
  print("Reject Null Hypothesis")
  print("Season and weather are associated.")
```

```
chi_statistic : 49.15865559689363
p_value :  1.5499250736864862e-07
dof :  9
expected :  [[1.77454639e+03 1.80559765e+03 1.80559765e+03 1.80625831e+03]
 [6.99258130e+02 7.11493845e+02 7.11493845e+02 7.11754180e+02]
 [2.11948742e+02 2.15657450e+02 2.15657450e+02 2.15736359e+02]
 [2.46738931e-01 2.51056403e-01 2.51056403e-01 2.51148264e-01]]
Reject Null Hypothesis
Season and weather are associated.
```

Standardized residuals

```
import numpy as np

observed = chi_cross.values
std_resid = (observed - expected) / np.sqrt(expected)
std_resid_df = pd.DataFrame(std_resid, index=chi_cross.index, columns=chi_cross.columns)
sns.heatmap(std_resid_df,annot=True)
plt.show()
```



Insights

- Clear weather occurs more than expected in Fall, and Mist occurs more than expected in Winter, Indicating more demand in these periods.
- Other weather-season combinations don't have much impact on demand.

STEP-5: Conclusion

Standardized residuals indicate a relationship between Clear weather in Fall and Mist in Winter. The chi-square test yields a p-value less than alpha (0.05), so we reject the null hypothesis. Therefore, weather and season are statistically associated, affecting demand.

Recommendations

Season

- No of bikes rented durin spring season is low, So yulu can provide price deduction, discount or ofers to increse the count.
- The demand for yulu bikes peaks during fall and summer seasons, Therefore they should increase the supply of bikes to meet this rising demand.
- Yulu would provide bike equipped with rain covers during raining season.

Weather

- Yulu should stock a higher number of electric cycles during clear weather to meet increased demand.
- To boost rentals during misty (light fog) and winter conditions, Yulu can invest in bikes equipped with high-intensity LED lights, reflective stickers and anti-slip tires to improve safety and rider confidence.

Workingday

- Offering flexible rental plans for daily commuters to make motorcycles more accessible.
- Partnering with companies to provide motorcycles for employee commutes at discounted rates.
- Rent motorcycles with ready-made travel routes for holiday trips. Riders get maps, tips, and scenic ride suggestions for a great experience.

- Yulu can bring limitted time discounts and cash back offers during holidays to attrack more users.

Environmental factors

- During days with lower humidity, lower windspeed, and moderate temperature, Yulu should ensure adequate availability of bikes, as these conditions are associated with higher rental demand.
- Fewer bikes are rented late at night, so Yulu can offer discounts or special deals during those hours to encourage more people to rent. Yulu can target night shift workers, by offering exclusive late-night deals or discounts to encourage more rentals during those hours.
- Yulu can utilize low-demand periods to carry out regular maintenance, servicing, and safety checks on their electric cycles. This ensures that bikes remain in optimal condition during peak demand hours and improves user satisfaction and reliability.