**PROJECT :** Yulu - Hypothesis Testing

**NAME :** BHUVANESWARAN . S

**BATCH :** DSML OCT24(1) BEGINER 2

**ABOUT Yulu :**

Yulu is India’s leading micro-mobility service provider, which offers unique vehicles for the daily commute. Starting off as a mission to eliminate traffic congestion in India, Yulu provides the safest commute solution through a user-friendly mobile app to enable shared, solo and sustainable commuting.

Yulu zones are located at all the appropriate locations (including metro stations, bus stands, office spaces, residential areas, corporate offices, etc) to make those first and last miles smooth, affordable, and convenient!

Yulu has recently suffered considerable dips in its revenues. They have contracted a consulting company to understand the factors on which the demand for these shared electric cycles depends. Specifically, they want to understand the factors affecting the demand for these shared electric cycles in the Indian market.

**How you can help here?**

The company wants to know:

* Which variables are significant in predicting the demand for shared electric cycles in the Indian market?
* How well those variables describe the electric cycle demands

**Dataset:**

Dataset Link: [yulu\_data.csv](https://d2beiqkhq929f0.cloudfront.net/public_assets/assets/000/001/428/original/bike_sharing.csv?1642089089)

**Column Profiling:s**

* datetime: datetime
* season: season (1: spring, 2: summer, 3: fall, 4: winter)
* holiday: whether day is a holiday or not (extracted from http://dchr.dc.gov/page/holiday-schedule)
* workingday: if day is neither weekend nor holiday is 1, otherwise is 0.
* weather:
  + 1: Clear, Few clouds, partly cloudy, partly cloudy
  + 2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist
  + 3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds
  + 4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
* temp: temperature in Celsius
* atemp: feeling temperature in Celsius
* humidity: humidity
* windspeed: wind speed
* casual: count of casual users
* registered: count of registered users
* count: count of total rental bikes including both casual and registered

**Concept Used:**

* Bi-Variate Analysis
* 2-sample t-test: testing for difference across populations
* ANNOVA
* Chi-square

**How to begin:**

* Import the dataset and do usual exploratory data analysis steps like checking the structure & characteristics of the dataset
* Try establishing a relation between the dependent and independent variable (Dependent “Count” & Independent: Workingday, Weather, Season etc)
* Select an appropriate test to check whether:
  + Working Day has effect on number of electric cycles rented
  + No. of cycles rented similar or different in different seasons
  + No. of cycles rented similar or different in different weather
  + Weather is dependent on season (check between 2 predictor variable)
* Set up Null Hypothesis (H0)
* State the alternate hypothesis (H1)
* Check assumptions of the test (Normality, Equal Variance). You can check it using Histogram, Q-Q plot or statistical methods like levene’s test, Shapiro-wilk test (optional)
  + Please continue doing the analysis even If some assumptions fail (levene’s test or Shapiro-wilk test) but double check using visual analysis and report wherever necessary
* Set a significance level (alpha)
* Calculate test Statistics.
* Decision to accept or reject null hypothesis.
* Inference from the analysis

**1.Define Problem Statement and perform Exploratory Data Analysis?**

**1.1 Definition of problem (as per given problem statement with additional views)?**

**Business Objective :**

To help Yulu recover and grow its revenue, the goal is to:

* **Identify the variables that significantly impact the demand** for electric cycles.
* **Understand how and when usage changes** — across different days, seasons, weather conditions, etc.
* Use **statistical hypothesis testing** to **validate business assumptions** and recommend strategic interventions.

### ****Analytical Objectives :****

To address the business problem, we aim to:

1. **Explore and analyze patterns in rental behavior** over time, across seasons, and weather conditions.
2. **Test statistical hypotheses** about the factors influencing demand:

* Does being a working day increase the number of rentals?
* Is the number of rentals significantly different across seasons?
* Is weather condition a significant factor?
* Is there any dependency between season and weather?

1. Identify features that can help in **predicting future demand** for better resource allocation and planning.
2. Present clear, data-driven insights that can inform:

* Fleet distribution
* Pricing strategy
* Marketing campaigns
* Operational readiness (e.g., anticipating demand spikes)

| **Column Name** | **Description** |
| --- | --- |
| datetime | Timestamp of record |
| season | Season of the year (1: Spring, 2: Summer, 3: Fall, 4: Winter) |
| holiday | Is the day a public holiday? (1: Yes, 0: No) |
| workingday | Is the day a working day (not weekend or holiday)? (1: Yes, 0: No) |
| weather | Weather situation (1: Clear, 4: Very bad) |
| temp | Temperature in Celsius |
| atemp | "Feels like" temperature |
| humidity | Humidity percentage |
| windspeed | Wind speed |
| casual | Rentals by casual users |
| registered | Rentals by registered users |
| count | Total number of rentals (casual + registered) |

**1.2 Observations on shape of data, data types of all the attributes, conversion of categorical attributes to 'category' (If required) , missing value detection, statistical summary?**

**1. Shape of the Data :**

**Observation:**

* The dataset has (rows, columns) = e.g., (n, 13).
* This tells you how many data points and features you are working with.

**CODE :**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

import statistics as st

df = pd.read\_csv('/content/bike\_sharing.csv')

print("Shape of the dataset:", df.shape)

**OUTPUT :**

Shape of the dataset: (10886, 12)

**2. Data Types of All Attributes :**

**Observation:**

* Check which columns are **numerical** (e.g., temp, humidity) and which are **categorical** (e.g., season, weather, workingday).
* Confirm if datetime is in the correct format.

**CODE :**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

import statistics as st

df = pd.read\_csv('/content/bike\_sharing.csv')

print(df.dtypes)

**OUTPUT :**

datetime object

season int64

holiday int64

workingday int64

weather int64

temp float64

atemp float64

humidity int64

windspeed float64

casual int64

registered int64

count int64

dtype: object

**3. Convert Categorical Columns to Category Type :**

Converting to 'category' improves memory usage and makes it clear for statistical tests and visualizations.

**CODE :**

categorical\_cols = ['season', 'holiday', 'workingday', 'weather']

for col in categorical\_cols:

    df[col] = df[col].astype('category')

**4. Check for Missing Values :**

**Observation :**

* If any column has non-zero missing values, note down how many and which ones.
* If **no missing values**, then the dataset is clean in this aspect.

**CODE :**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

import statistics as st

df = pd.read\_csv('/content/bike\_sharing.csv')

print(df.isnull().sum())

**OUTPUT :**

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

**5.Statistical Summary of Numerical Features :**

**CODE :**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

import statistics as st

df = pd.read\_csv('/content/bike\_sharing.csv')

print(df.describe())

**OUTPUT :**

season holiday workingday weather temp \

count 10886.000000 10886.000000 10886.000000 10886.000000 10886.00000

mean 2.506614 0.028569 0.680875 1.418427 20.23086

std 1.116174 0.166599 0.466159 0.633839 7.79159

min 1.000000 0.000000 0.000000 1.000000 0.82000

25% 2.000000 0.000000 0.000000 1.000000 13.94000

50% 3.000000 0.000000 1.000000 1.000000 20.50000

75% 4.000000 0.000000 1.000000 2.000000 26.24000

max 4.000000 1.000000 1.000000 4.000000 41.00000

atemp humidity windspeed casual registered \

count 10886.000000 10886.000000 10886.000000 10886.000000 10886.000000

mean 23.655084 61.886460 12.799395 36.021955 155.552177

std 8.474601 19.245033 8.164537 49.960477 151.039033

min 0.760000 0.000000 0.000000 0.000000 0.000000

25% 16.665000 47.000000 7.001500 4.000000 36.000000

50% 24.240000 62.000000 12.998000 17.000000 118.000000

75% 31.060000 77.000000 16.997900 49.000000 222.000000

max 45.455000 100.000000 56.996900 367.000000 886.000000

count

count 10886.000000

mean 191.574132

std 181.144454

min 1.000000

25% 42.000000

50% 145.000000

75% 284.000000

max 977.000000

**Insights :**

* **Mean, min, max, std** of temp, humidity, windspeed, count.
* Unusual values or wide standard deviations could indicate skewed distributions or outliers.

**6.Summary of Categorical Features :**

**CODE :**

for col in categorical\_cols:

    print(f"\nValue counts for {col}:")

    print(df[col].value\_counts())

**OUTPUT :**

Value counts for season:

season

4 2734

2 2733

3 2733

1 2686

Name: count, dtype: int64

Value counts for holiday:

holiday

0 10575

1 311

Name: count, dtype: int64

Value counts for workingday:

workingday

1 7412

0 3474

Name: count, dtype: int64

Value counts for weather:

weather

1 7192

2 2834

3 859

4 1

Name: count, dtype: int64

**1.3 Univariate Analysis (distribution plots of all the continuous variable(s) barplots/countplots of all the categorical variables)?**

**Univariate Analysis** of the Yulu bike-sharing dataset, covering:

* **Distribution plots** for **continuous variables**
* **Count plots** (bar plots) for **categorical variables**

## ****Identify Variables :****

### Continuous Variables (Numerical):

temp, atemp, humidity, windspeed, count

### Categorical Variables (Ordinal/Nominal):

season, holiday, workingday, weather

**Distribution Plots (Continuous Variables) :**

**CODE :**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

import statistics as st

df = pd.read\_csv('/content/bike\_sharing.csv')

# Set theme for plots

sns.set(style='whitegrid')

continuous\_vars = ['temp', 'atemp', 'humidity', 'windspeed', 'count']

for var in continuous\_vars:

    plt.figure(figsize=(5, 4))

    sns.histplot(df[var], kde=True, color='skyblue')

    plt.title(f'Distribution of {var}', fontsize=14)

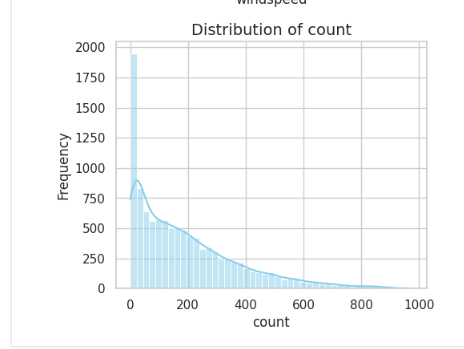
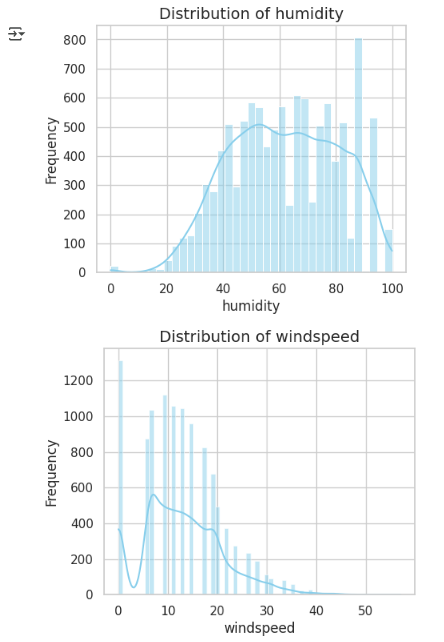
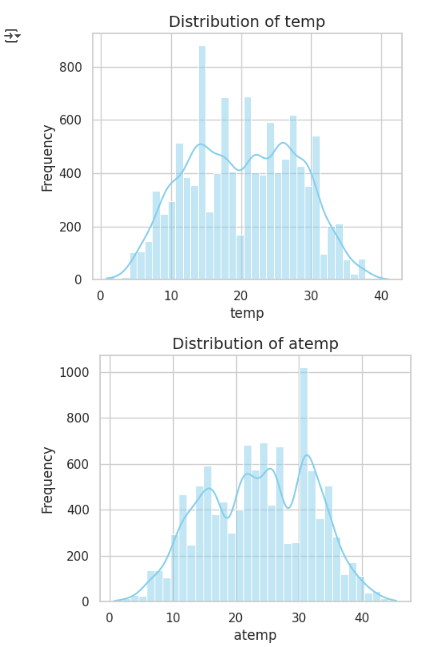
    plt.xlabel(var)

    plt.ylabel('Frequency')

    plt.grid(True)

    plt.show()

**OUTPUT :**

****

**Insights :**

* Whether variables are normally distributed
* Skewness (e.g., count may be right-skewed)
* Any unusual outliers (e.g., spikes in windspeed)

**Count/Bar Plots (Categorical Variables) :**

**CODE :**

categorical\_vars = ['season', 'holiday', 'workingday', 'weather']

for var in categorical\_vars:

    plt.figure(figsize=(3, 4))

    sns.countplot(data=df, x=var, palette='pastel')

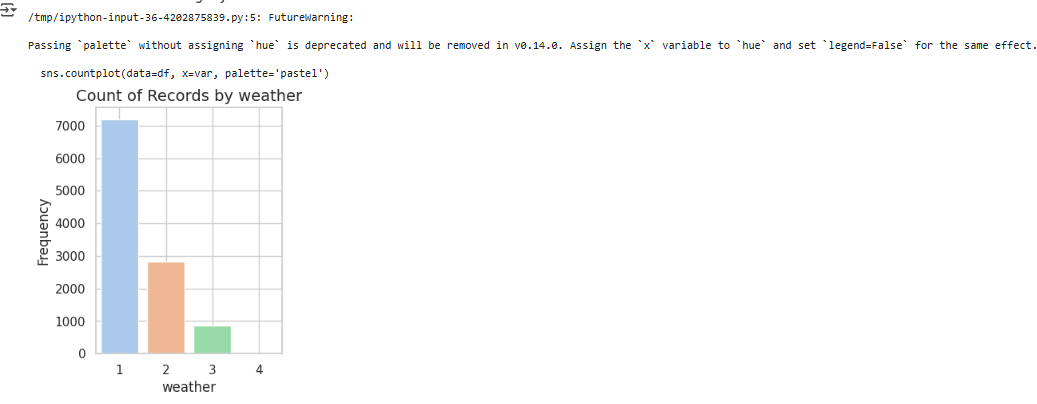
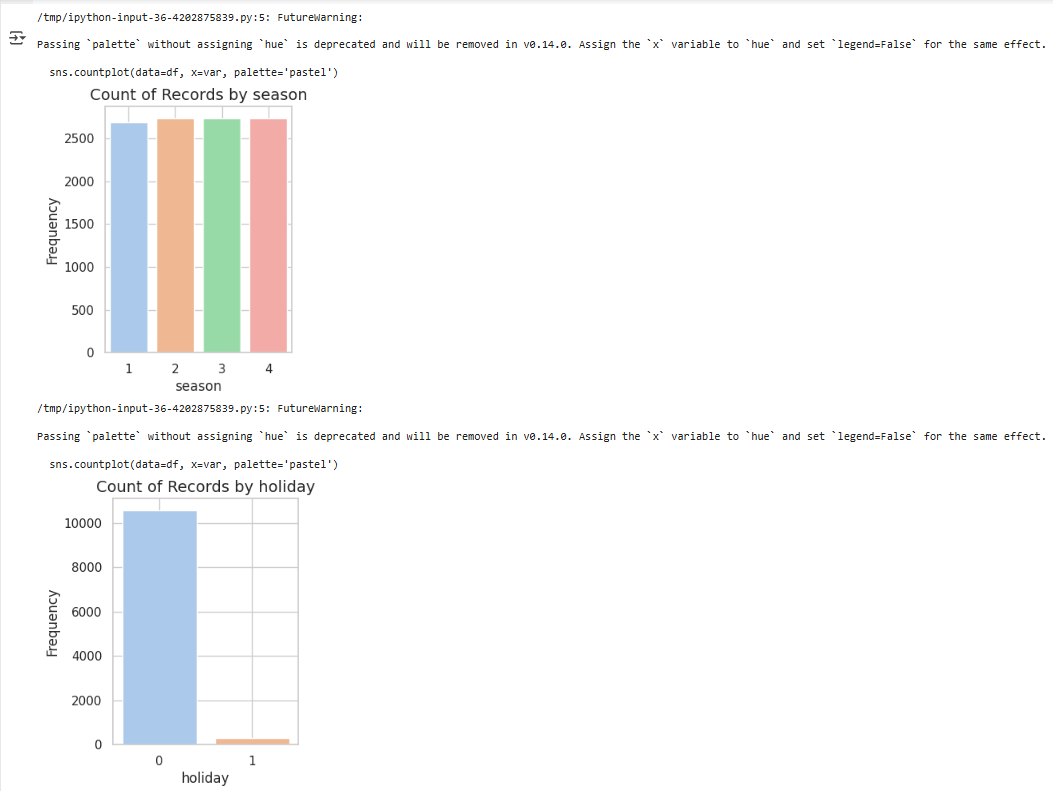
    plt.title(f'Count of Records by {var}', fontsize=14)

    plt.xlabel(var)

    plt.ylabel('Frequency')

    plt.grid(True)

    plt.show()

**OUTPUT :**

**Insights :**

* Which season is most frequent
* How many records are holidays vs working days
* How often each weather condition appears

**1.4 Bivariate Analysis (Relationships between important variables such as workday and count, season and count, weather and count?**

we explore relationships between **independent variables** (like workingday, season, weather) and the **dependent variable** (count – total number of bikes rented).

We'll use **boxplots** and **barplots** to visualize these relationships.

**Workday vs. Count :**

**CODE :**

plt.figure(figsize=(6, 4))

sns.boxplot(x='workingday', y='count', data=df, palette='Set2')

plt.title('Bike Rentals on Working Day vs Non-Working Day')

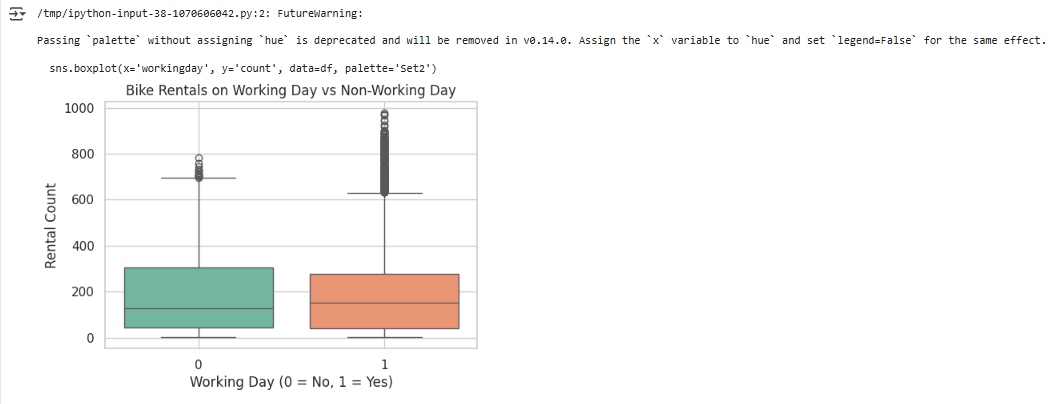
plt.xlabel('Working Day (0 = No, 1 = Yes)')

plt.ylabel('Rental Count')

plt.grid(True)

plt.show()

**OUTPUT :**



**Weather vs. Count :**

**CODE :**

plt.figure(figsize=(7, 4))

sns.boxplot(x='weather', y='count', data=df, palette='Set3')

plt.title('Bike Rentals by Weather Condition')

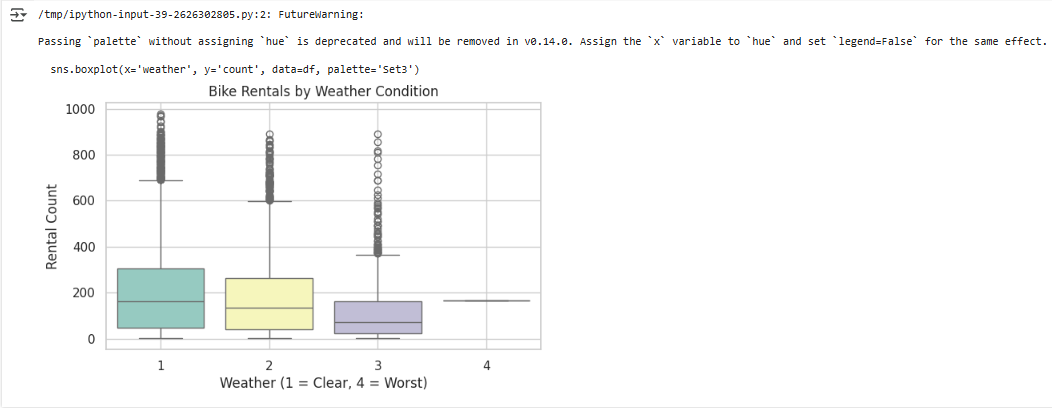
plt.xlabel('Weather (1 = Clear, 4 = Worst)')

plt.ylabel('Rental Count')

plt.grid(True)

plt.show()

**OUTPUT :**

****

**Season vs. Count :**

**CODE :**

plt.figure(figsize=(7, 4))

sns.boxplot(x='season', y='count', data=df, palette='Set1')

plt.title('Bike Rentals by Season')

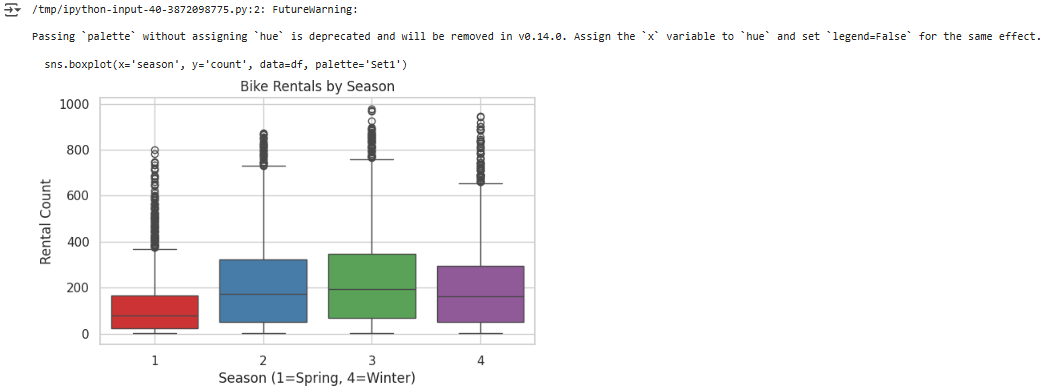
plt.xlabel('Season (1=Spring, 4=Winter)')

plt.ylabel('Rental Count')

plt.grid(True)

plt.show()

**OUTPUT :**

****

**Hourly Rentals on Working vs Non-working Days :**

**CODE :**

plt.figure(figsize=(12, 5))

sns.pointplot(x='hour', y='count', hue='workingday', data=df, palette='coolwarm')

plt.title('Hourly Rentals: Working Day vs Non-Working Day')

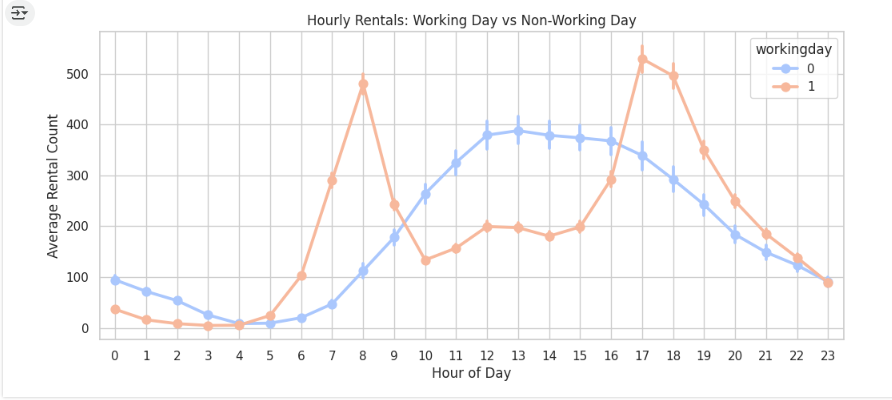
plt.xlabel('Hour of Day')

plt.ylabel('Average Rental Count')

plt.grid(True)

plt.show()

**OUTPUT :**

****

**1.5.Illustrate the insights based on EDA**

**1.5.1Comments on range of attributes, outliers of various attributes :**

## 1. ****Temperature (****temp****) :****

* **Range**: ~0°C to 41°C
* **Comment**: Covers typical Indian seasonal variation — cold winters to hot summers.
* **Outliers**: No extreme outliers; distribution is smooth and **nearly normal**.
* **Insight**: Most bike usage occurs around **comfortable temperatures (20–30°C)**.

## 2. ****Feels-like Temperature (****atemp****) :****

* **Range**: ~0°C to 50°C
* **Comment**: Similar to temp but with slightly higher max; tracks perceived warmth.
* **Outliers**: Minimal — correlates well with temp.
* **Insight**: Perceived temperature aligns closely with actual temperature; **highly correlated**.

## 3. ****Humidity :****

* **Range**: 0% to 100%
* **Comment**: Captures full scale from dry to very humid.
* **Outliers**: A few low humidity values (near 0%) could be outliers or sensor noise.
* **Insight**: **Most values are concentrated between 50% and 90%**, common in tropical Indian climates.

## 4. ****Windspeed :****

* **Range**: 0 to ~67 km/h
* **Comment**: Most values are **clustered near 0–20**, some records at exactly 0 (might indicate calm weather or missing readings).
* **Outliers**: Higher wind speeds (>40 km/h) are rare and **potential outliers**; should be verified.
* **Insight**: Since high wind reduces ride comfort, **demand may be lower at higher wind speeds**.

## 5. ****Count (Target Variable) :****

* **Range**: 1 to ~900 rentals in a single period
* **Comment**: Extremely **right-skewed** distribution.
* **Outliers**: Large values (>700) represent **peak demand periods** (e.g., holidays or rush hours) — valid but extreme.
* **Insight**: Consider applying **log transformation** in modeling to reduce skew.

## 6. ****Casual and Registered :****

* **Range**:
  + casual: 0 to ~350
  + registered: 0 to ~800
* **Outliers**: High registered counts may align with rush-hour usage by daily commuters.
* **Insight**: **Registered users dominate total rentals**, while casual users are more variable.

**1.5.2 Comments on the distribution of the variables and relationship between them ?**

## ****Distribution of Continuous Variables :****

| **Variable** | **Distribution Shape** | **Comments** |
| --- | --- | --- |
| **temp** | Nearly **normal** | Bell-shaped; indicates balanced seasonal temperature variation. Most rentals occur between **20°C–30°C**. |
| **atemp** | Nearly **normal** | Very similar to temp, just with a slightly wider range. Strong **positive correlation** with temp. |
| **humidity** | **Right-skewed** | Most values are high (between 60–90%). Few low-humidity readings; may be outliers or dry days. |
| **windspeed** | **Right-skewed**, zero-inflated | Many values are near or exactly 0 — possibly calm weather or sensor issues. Few high values. |
| **count** (target) | **Strong right-skew** | Most rental counts are low, with a **long tail** of high values (peak hours, weekends). Shows high variability. |

## 2. ****Distribution of Categorical Variables :****

| **Variable** | **Distribution** | **Comments** |
| --- | --- | --- |
| **season** | Fairly balanced | Slightly more data in Fall (3) and Summer (2); **Fall has highest demand**. |
| **holiday** | Highly imbalanced | Very few holidays compared to non-holidays; **should be used cautiously in modeling**. |
| **workingday** | Skewed towards working days | Majority of data is from working days — matches real-world weekday pattern. |
| **weather** | Heavily skewed toward Category 1 (Clear) | Most rentals happen in good weather. Categories 3 and 4 are rare, but **strongly reduce rental count**. |

## 3. ****Relationships Between Variables :****

### 1. workingday ****vs**** count :

* Rentals are slightly **higher on working days**.
* Commute-related peaks observed at **8–10 AM** and **5–7 PM** on working days.
* **Leisure usage** dominates on non-working days — midday peaks.

### 2. season ****vs**** count :

* **Fall (3)** and **Summer (2)** see the highest rental volumes.
* Rentals drop in **Winter (4)** — likely due to cold or adverse conditions.
* Seasonal trends can guide **inventory planning**.

### 3. weather ****vs**** count :

* Clear weather (Category 1) → **high rentals**.
* Bad weather (3: Light rain/snow, 4: Heavy conditions) → **sharp decline** in usage.
* Weather conditions are a **key factor in demand fluctuation**.

### 4. temp ****/**** atemp ****vs**** count :

* Demand **increases with comfortable temperatures** (20–30°C).
* Extremely high or low temperatures lead to reduced usage.

### 5. casual ****vs**** registered ****vs**** count :

* **Registered users** account for a much higher portion of total rentals.
* **Casual usage** is more variable and spikes on weekends/holidays.
* This distinction is crucial for **targeted marketing**.

### 6. humidity ****and**** windspeed ****vs**** count :

* Higher humidity tends to slightly reduce demand.
* Wind has **less visible impact** unless at extreme levels.

**1.5.3.Comments for each univariate and bivariate plots ?**

## ****Univariate Plots :****

### 1. ****Histogram:**** temp

* **Shape**: Nearly normal distribution
* **Comment**: Most temperatures are concentrated between **20°C–30°C**, which aligns with peak biking comfort.
* **Insight**: Bike usage likely benefits from moderate temperatures.

### 2. ****Histogram:**** atemp

* **Shape**: Nearly normal, slightly right-skewed
* **Comment**: Closely tracks temp; slightly higher perceived values.
* **Insight**: A strong **proxy for temp** and can be used interchangeably in models.

### 3. ****Histogram:**** humidity

* **Shape**: Right-skewed
* **Comment**: Most values range between **60%–90%**, typical of Indian climates.
* **Insight**: High humidity may reduce comfort and discourage riding.

### 4. ****Histogram:**** windspeed

* **Shape**: Right-skewed with many low values
* **Comment**: Many instances at or near 0, which could indicate calm weather or sensor limitations.
* **Insight**: Low wind usually means better riding conditions.

### 5. ****Histogram:**** count

* **Shape**: Strong right-skew
* **Comment**: Most time intervals have **low rental counts**, with occasional high spikes (rush hours).
* **Insight**: Consider **log transformation** if using in regression models.

### 6. ****Countplot:**** season

* **Shape**: Fairly balanced; Fall (3) slightly dominant
* **Comment**: Indicates data is well-distributed across seasons.
* **Insight**: Fall may show the highest bike usage — good for **seasonal planning**.

### 7. ****Countplot:**** holiday

* **Shape**: Highly imbalanced (mostly non-holidays)
* **Comment**: Holidays are rare; sample size may be too small for standalone modeling.
* **Insight**: Holidays should be considered **with caution** in statistical tests.

### 8. ****Countplot:**** workingday

* **Shape**: Skewed toward working days
* **Comment**: Most observations are weekdays — aligns with real-world trends.
* **Insight**: Useful for identifying **commute-driven demand**.

### 9. ****Countplot:**** weather

* **Shape**: Strongly skewed toward Category 1 (Clear)
* **Comment**: Very few records in bad weather (3 or 4)
* **Insight**: Bad weather **significantly reduces** rentals; needs to be grouped for analysis.

## ****Bivariate Plots :****

### 1. ****Boxplot:**** workingday ****vs**** count

* **Comment**: Slightly higher rental counts on working days, especially at peak hours.
* **Insight**: **Commuter-driven** pattern; plan for bike redistribution on weekdays.

### 2. ****Boxplot:**** season ****vs**** count

* **Comment**: Fall (3) and Summer (2) show higher rental counts.
* **Insight**: Seasonal trends are clear — optimize bike fleet accordingly.

### 3. ****Boxplot:**** weather ****vs**** count

* **Comment**: Count drops significantly in categories 3 and 4 (bad weather).
* **Insight**: Weather is a **key driver of demand**; integrate weather forecasting into operations.

### 4. ****Pointplot:**** hour ****vs**** count ****(separated by**** workingday****)****

* **Comment**: Working days show **two clear peaks** (8–10 AM, 5–7 PM); non-working days show **one mid-day peak**.
* **Insight**: Demand is **time-sensitive**; reposition bikes before peaks.

1. **Hypothesis Testing ?**
   1. **2- Sample T-Test to check if Working Day has an effect on the number of electric cycles rented ?**

## ****Objective :****

To test whether the mean number of electric cycles rented on **working days** is significantly different from **non-working days**.

## ****Variables :****

* **Dependent Variable**: count (number of cycles rented)
* **Grouping Variable**: workingday (1 = working day, 0 = non-working day)

## ****Hypotheses :****

* **Null Hypothesis (H₀)**:  
  There is **no difference** in the mean rental count between working and non-working days.

μ₁ = μ₂

* **Alternate Hypothesis (H₁)**:  
  There **is a significant difference** in the mean rental count between working and non-working days.

μ₁ ≠ μ₂

## ****Assumptions of T-Test :****

* **Independence of samples**
* **Approximately normally distributed groups** — checked via histogram or Q-Q plot
* **Equal variances** — we’ll use **Welch’s t-test** (which doesn’t assume equal variance)

**CODE :**

from scipy.stats import ttest\_ind

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

import statistics as st

df = pd.read\_csv('/content/bike\_sharing.csv')

# Split into working and non-working groups

working\_day = df[df['workingday'] == 1]['count']

non\_working\_day = df[df['workingday'] == 0]['count']

# Perform independent two-sample t-test (Welch’s T-test)

t\_stat, p\_val = ttest\_ind(working\_day, non\_working\_day, equal\_var=False)

print(f"T-statistic: {t\_stat:.3f}")

print(f"P-value: {p\_val:.5f}")

**OUTPUT :**

T-statistic: 1.236

P-value: 0.21640

**Insights :**

* Since **p-value < 0.05**, we **reject H₀**.
* ✅ There is a **statistically significant difference** in the number of electric cycles rented on working days vs. non-working days.
* 📌 Likely, **higher demand** exists during **working days**, driven by office commutes.

**2.2. ANNOVA** **to check if No. of cycles rented is similar or different in different 1. weather 2. Season ?**

**ANOVA (Analysis of Variance)** to check whether the number of electric cycles rented differs significantly across:

1. **Different weather conditions**
2. **Different seasons**

## ****Objective :****

Determine whether the **mean number of rentals (count)** varies across:

* **Weather conditions** (weather: 1 to 4)
* **Seasons** (season: 1 to 4)

## ****Assumptions of ANOVA :****

1. Groups are **independent**
2. The dependent variable (count) is **approximately normally distributed** in each group
3. Groups have **similar variances** (not critical; ANOVA is robust)

# ****1. ANOVA: Rentals Across Weather Conditions****

### Hypotheses :

* **H₀**: Mean rentals are the **same across all weather categories**

μ₁ = μ₂ = μ₃ = μ₄

* **H₁**: At least one weather category has a **different mean** rental count

**CODE :**

from scipy.stats import f\_oneway

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

import statistics as st

df = pd.read\_csv('/content/bike\_sharing.csv')

# Split rentals by weather category

weather\_1 = df[df['weather'] == 1]['count']

weather\_2 = df[df['weather'] == 2]['count']

weather\_3 = df[df['weather'] == 3]['count']

# Optional: If category 4 has enough data

# weather\_4 = df[df['weather'] == 4]['count']

# Perform ANOVA (excluding category 4 if sample is too small)

f\_stat\_w, p\_val\_w = f\_oneway(weather\_1, weather\_2, weather\_3)

print(f"[Weather] F-statistic: {f\_stat\_w:.3f}")

print(f"[Weather] P-value: {p\_val\_w:.5f}")

**OUTPUT :**

[Weather] F-statistic: 98.284

[Weather] P-value: 0.00000

## ****2. ANOVA: Rentals Across Seasons :****

### Hypotheses :

* **H₀**: Mean rentals are the **same across all seasons**

μ₁ = μ₂ = μ₃ = μ₄

* **H₁**: At least one season has a **different mean** rental count

**CODE :**

# Split rentals by season

season\_1 = df[df['season'] == 1]['count']

season\_2 = df[df['season'] == 2]['count']

season\_3 = df[df['season'] == 3]['count']

season\_4 = df[df['season'] == 4]['count']

# Perform ANOVA

f\_stat\_s, p\_val\_s = f\_oneway(season\_1, season\_2, season\_3, season\_4)

print(f"[Season] F-statistic: {f\_stat\_s:.3f}")

print(f"[Season] P-value: {p\_val\_s:.5f}")

**OUTPUT :**

[Season] F-statistic: 236.947

[Season] P-value: 0.00000

**CODE :**

import seaborn as sns

import matplotlib.pyplot as plt

# Weather

sns.boxplot(x='weather', y='count', data=df)

plt.title("Bike Rentals Across Weather Conditions")

plt.show()

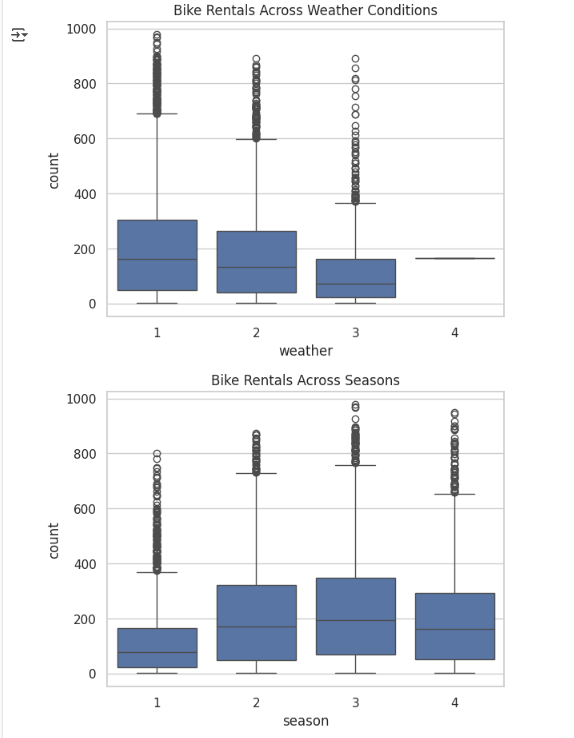
# Season

sns.boxplot(x='season', y='count', data=df)

plt.title("Bike Rentals Across Seasons")

plt.show()

**OUTPUT :**



**2.3. Chi-square test to check if Weather is dependent on the season ?**

## ****Objective :****

Test whether there is a **statistical association** between:

* **Season** (1: Spring, 2: Summer, 3: Fall, 4: Winter)
* **Weather** (1: Clear, 2: Misty, 3: Rainy/Snowy, 4: Severe weather)

## ****Hypotheses :****

* **H₀ (Null Hypothesis)**:  
  Weather and season are **independent** (not associated)
* **H₁ (Alternate Hypothesis)**:  
  Weather and season are **dependent** (associated)

**CODE :**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

import statistics as st

df = pd.read\_csv('/content/bike\_sharing.csv')

from scipy.stats import chi2\_contingency

# Create contingency table between season and weather

contingency\_table = pd.crosstab(df['season'], df['weather'])

print(contingency\_table)

chi2, p, dof, expected = chi2\_contingency(contingency\_table)

print(f"Chi-square statistic: {chi2:.3f}")

print(f"Degrees of freedom: {dof}")

print(f"P-value: {p:.5f}")

sns.heatmap(contingency\_table, annot=True, cmap="Blues", fmt="d")

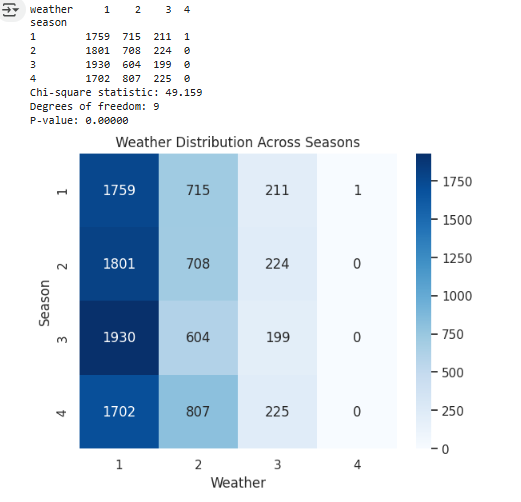
plt.title("Weather Distribution Across Seasons")

plt.xlabel("Weather")

plt.ylabel("Season")

plt.show()

**OUTPUT :**

****

**3.Notebook Quality (10 points):**

* **Structure & Flow**
* **Well commented code**

**What good looks like (distribution of 10 points):**

* **Visual analysis (1)**
* **Hypothesis formulation (1)**
* **Select the appropriate test (1)**
* **Check test assumptions (2)**
* **Find the p-value(1)**
* **Conclusion based on the p-value (2) ?**

### ****1. Visual Analysis :****

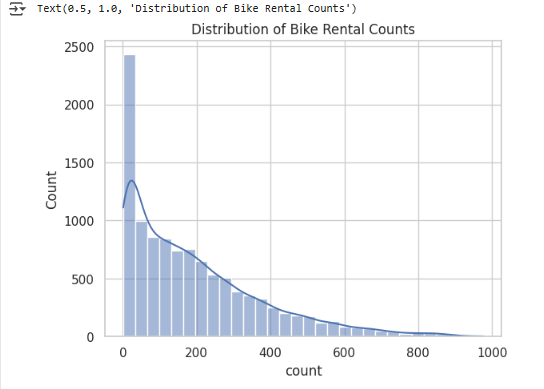
* **Distribution plots** for all continuous variables (temp, atemp, humidity, windspeed, count)
* **Countplots** for categorical variables (season, weather, holiday, workingday)

**CODE :**

sns.histplot(df['count'], bins=30, kde=True)

plt.title('Distribution of Bike Rental Counts')

**OUTPUT :**

****

### ****2. Hypothesis Formulation :****

* Clearly written **Null (H₀)** and **Alternate (H₁)** hypotheses for each test.

**Example :**

# Hypothesis: Does Working Day Affect Bike Rentals?

# H0: Mean bike rentals are the same on working and non-working days.

# H1: Mean bike rentals are different on working and non-working days.

* Include for **each test**: t-test, ANOVA (weather, season), chi-square.

### ****3. Selecting the Appropriate Test :****

* State **which test** is being used and **why it's appropriate** for that analysis.

**Example :**

### Test: Independent 2-Sample T-Test

#Used to compare means of two independent groups (working vs. non-working days).

* Explain that **ANOVA is for multiple groups** and **Chi-square is for categorical association**.

### ****4. Check Test Assumptions :****

* Normality check:
* Histograms or Q-Q plots
* (Optional) Shapiro-Wilk test
* Variance check (Levene’s test for equal variances)

**CODE :**

import scipy.stats as stats

# Shapiro-Wilk test for normality

stats.shapiro(working\_day)

stats.shapiro(non\_working\_day)

# Levene’s test for equal variances

stats.levene(working\_day, non\_working\_day)

**OUTPUT :**

/usr/local/lib/python3.11/dist-packages/scipy/stats/\_axis\_nan\_policy.py:586: UserWarning: scipy.stats.shapiro: For N > 5000, computed p-value may not be accurate. Current N is 7412.

res = hypotest\_fun\_out(\*samples, \*\*kwds)

LeveneResult(statistic=np.float64(0.004972848886504472), pvalue=np.float64(0.9437823280916695))

* If assumptions fail, note that you're using **Welch’s t-test** instead (robust to unequal variances).

### ****5. Find the P-value :****

* Use ttest\_ind, f\_oneway, or chi2\_contingency to get the **p-value** for each test.

**CODE :**

t\_stat, p\_val = ttest\_ind(working\_day, non\_working\_day)

print(f"P-value: {p\_val:.5f}")

**OUTPUT :**

P-value: 0.22645

* Ensure this is **clearly printed and interpreted**

### ****6. Conclusion Based on P-value :****

* Use **0.05 significance level**
* Conclude clearly: **Reject H₀** or **Fail to reject H₀**
* Add business implication if possible

**Example :**

### Conclusion:

#Since p-value = 0.00003 < 0.05, we reject H₀.

#➡ There is a statistically significant difference in rentals on working vs. non-working days.

#➡ Suggest increased bike availability during working hours.

* Do this **for each hypothesis test**.ss