**Top 10 Indian Companies Sales & Revenue Analysis**

**Step 1: Prepare Your Environment**

1. **Install Power BI Desktop**: Ensure you have the latest version of Power BI Desktop installed.
2. **Install Python**: Download and install Python from the official website. Install necessary libraries like pandas, matplotlib, seaborn, etc.

bash

Copy code

pip install pandas matplotlib seaborn

**Step 2: Load Data into Power BI**

1. **Open Power BI Desktop**: Launch Power BI Desktop.
2. **Get Data**: Click on “Get Data” and choose the source of your dataset (Excel, CSV, database, etc.). Import the top 50 Indian companies dataset.

**Step 3: Use Python for Data Manipulation**

1. **Transform Data**:
   * Go to the “Home” tab and click on “Transform Data” to open the Power Query Editor.
   * Select “Transform” and then “Run Python Script”.
2. **Write Python Script**:
   * In the Python script editor, write your script to manipulate the data. For example:

python

Copy code

import pandas as pd

import matplotlib.pyplot as plt

# Assuming 'dataset' is the dataframe imported in Power BI

top\_50 = dataset

# Data transformation (if needed)

top\_50['Revenue Growth %'] = top\_50['Revenue Growth %'].str.rstrip('%').astype('float') / 100.0

# Return the transformed dataframe

dataset = top\_50

* + Click “OK” to run the script. Power BI will execute the script and load the transformed data.

**Step 4: Create Visualizations with Python**

1. **Create Python Visuals**:
   * Go back to the main Power BI report view.
   * Click on the Python visual icon in the Visualizations pane (a Python logo).
2. **Add Data Fields**:
   * Drag the necessary fields from your dataset into the Values section of the Python visual.
   * In the Python script editor, use the provided dataset (usually named dataset) to create your visualizations. For example:

python

Copy code

import matplotlib.pyplot as plt

import seaborn as sns

# Sample Data

top\_10 = dataset.nlargest(10, 'Revenue').sort\_values(by='Revenue', ascending=False)

plt.figure(figsize=(10, 6))

sns.barplot(x='Revenue', y='Company Name', data=top\_10, palette='viridis')

plt.title('Top 10 Companies by Revenue')

plt.xlabel('Revenue (in millions)')

plt.ylabel('Company Name')

plt.show()

* + Click “OK” to run the script and generate the visual.

**Step 5: Build the Dashboard**

1. **Combine Visuals**: Combine Python visuals with native Power BI visuals (bar charts, line charts, pie charts, maps, etc.) to create a comprehensive dashboard.
2. **Add Interactivity**:
   * Use slicers, filters, and other interactive elements to allow users to interact with the data.
   * Link visuals together using Power BI’s built-in functionality to create a dynamic and interactive dashboard.

**Step 6: Publish and Share**

1. **Publish**: Once your dashboard is complete, publish it to Power BI Service by clicking the “Publish” button.
2. **Share**: Share the dashboard with stakeholders by providing access through the Power BI Service. You can also embed the dashboard into other applications if needed.

**Example Use Cases**

1. **Revenue and Profit Comparison**:
   * Use Python visuals to create detailed bar charts or scatter plots comparing revenue and profits.
   * Combine with native Power BI visuals for summary insights.
2. **Industry Analysis**:
   * Create a Python pie chart for industry distribution.
   * Use a Power BI bar chart to compare average revenues and profits across industries.
3. **Geographic Distribution**:
   * Use Power BI’s map visual to show the location of companies.
   * Enhance with Python visuals for detailed regional analysis.
4. **Growth Analysis**:
   * Use a Python scatter plot to analyze the relationship between revenue growth percentage and profit.

By integrating Python with Power BI, you can leverage the power of both tools to create a rich, interactive, and insightful dashboard. If you need further assistance or have specific questions about any step, feel free to ask!

1. **Top 10 Companies by Revenue bar chart**

import matplotlib.pyplot as plt

import seaborn as sns

# Sample Data

top\_10 = dataset.nlargest(10, 'Revenue').sort\_values(by='Revenue', ascending=False)

# Adjust figure size and font size

plt.figure(figsize=(14, 8))  # Increase figure size

sns.set(font\_scale=1.2)  # Increase font size

# Create bar plot

sns.barplot(x='Revenue', y='Name', data=top\_10, palette='viridis')

# Add title and labels

plt.title('Top 10 Companies by Revenue')

plt.xlabel('Revenue (in millions)')

plt.ylabel('Company Name')

# Adjust layout to ensure full company names are visible

plt.tight\_layout(pad=3.0)

# Display plot

plt.show()

1. **Top 10 Companies by Revenue and Profit bar chart**

import matplotlib.pyplot as plt

import seaborn as sns

import pandas as pd

# Sample Data (replace with your dataset)

# dataset = pd.DataFrame({

#     'Company Name': [...],

#     'Revenue': [...],

#     'Profit': [...]

# })

# Extract top 10 companies by revenue

top\_10 = dataset.nlargest(10, 'Revenue').sort\_values(by='Revenue', ascending=False)

# Adjust figure size and font size

plt.figure(figsize=(14, 8))

sns.set(font\_scale=1.2)

# Create bar plot for Revenue

sns.barplot(x='Revenue', y='Name', data=top\_10, color='b', label='Revenue')

# Add Profit bars on the same plot

sns.barplot(x='Profits', y='Name', data=top\_10, color='r', label='Profit')

# Add title and labels

plt.title('Top 10 Companies by Revenue and Profit')

plt.xlabel('Value (in millions)')

plt.ylabel('Company Name')

# Add legend

plt.legend()

# Adjust layout to ensure full company names are visible

plt.tight\_layout(pad=3.0)

# Display plot

plt.show()

1. **Revenue vs Profit Scatter Plot**

import matplotlib.pyplot as plt

import seaborn as sns

import pandas as pd

# Sample Data (replace with your dataset)

# dataset = pd.DataFrame({

#     'Company Name': [...],

#     'Revenue': [...],

#     'Profit': [...]

# })

# Adjust figure size and font size

plt.figure(figsize=(24, 14))  # Increase figure size

sns.set(font\_scale=1.2)

# Create scatter plot

scatter\_plot = sns.scatterplot(x='Revenue', y='Profits', data=dataset, hue='Name', palette='viridis', s=100)

# Add title and labels

plt.title('Revenue vs. Profit')

plt.xlabel('Revenue (in millions)')

plt.ylabel('Profit (in millions)')

# Move the legend outside of the plot

scatter\_plot.legend(loc='center left', bbox\_to\_anchor=(1, 0.5), ncol=1)  # Increase number of columns if needed

# Adjust layout to ensure there's space for the legend

plt.tight\_layout(pad=3.0)

# Display plot

plt.show()

1. **Industry Distributio Pie chart**

import matplotlib.pyplot as plt

import pandas as pd

# Sample Data (replace with your dataset)

# dataset = pd.DataFrame({

#     'Industry Type': [...],

#     'Company Name': [...],

#     'Revenue': [...],

#     'Profit': [...]

# })

# Calculate the distribution of companies by industry

industry\_distribution = dataset['Industry'].value\_counts()

# Adjust figure size

plt.figure(figsize=(12, 8))

# Create pie chart with detailed legend and labels

wedges, texts, autotexts = plt.pie(industry\_distribution, labels=industry\_distribution.index, autopct='%1.1f%%', startangle=140)

# Set properties for texts and autotexts

for text in texts:

    text.set\_size(12)

for autotext in autotexts:

    autotext.set\_size(12)

# Add title

plt.title('Industry Distribution', fontsize=16)

# Move the legend outside the plot

plt.legend(wedges, industry\_distribution.index, loc='center left', bbox\_to\_anchor=(1, 0.5))

# Adjust layout

plt.tight\_layout(pad=3.0)

# Display plot

plt.show()

1. **Comparison of Average Revenues and Profits Across Industries clustered bar chart**

import matplotlib.pyplot as plt

import seaborn as sns

import pandas as pd

# The dataset variable will be available in Power BI after you drag the fields into the Values area

# Ensure that the dataset includes columns: 'Industry Type', 'Average Revenue', 'Average Profit'

# Create a DataFrame from the dataset (Power BI will pass the dataset automatically)

data = dataset[['Industry', 'Average Revenue', 'Average Profit']]

# Plotting the bar chart

plt.figure(figsize=(14, 8))

sns.set(style="whitegrid")

# Melt the data for seaborn compatibility

melted\_data = pd.melt(data, id\_vars='Industry', value\_vars=['Average Revenue', 'Average Profit'],

                      var\_name='Metric', value\_name='Value')

# Create a barplot

sns.barplot(x='Value', y='Industry', hue='Metric', data=melted\_data, palette='viridis')

# Add titles and labels

plt.title('Comparison of Average Revenues and Profits Across Industries', fontsize=16)

plt.xlabel('Value (in millions)', fontsize=14)

plt.ylabel('Industry Type', fontsize=14)

# Adjust legend position

plt.legend(title='Metric', bbox\_to\_anchor=(1.05, 1), loc='upper left')

# Display the plot

plt.tight\_layout(pad=3.0)

plt.show()

1. **Revenue Growth Percentage vs Profit**

import matplotlib.pyplot as plt

import seaborn as sns

import pandas as pd

# Create a DataFrame from the dataset (Power BI will pass the dataset automatically)

data = dataset

# Initialize the plot

plt.figure(figsize=(12, 8))

sns.set(style="whitegrid")

# Create a scatter plot

scatter = sns.scatterplot(data=data, x='Revenue growth', y='Profits', hue='Profits', size='Revenue growth', palette='viridis', legend=False, sizes=(20, 200))

# Add titles and labels

plt.title('Revenue Growth Percentage vs Profit', fontsize=16)

plt.xlabel('Revenue Growth Percentage (%)', fontsize=14)

plt.ylabel('Profit (in millions)', fontsize=14)

# Optionally, add annotations for each point (company names)

#for line in range(0, data.shape[0]):

#     plt.text(data['Revenue growth'][line], data['Profits'][line],

 #             data['Name'][line], horizontalalignment='left',

#              size='medium', color='black', weight='semibold')

# Adjust layout to make room for annotations

plt.tight\_layout(pad=3.0)

# Show the plot

plt.show()