Phase - 4 Submission Document

Project Title: Product Sales Analysis

IBM Cognos is a robust business intelligence and data analytics platform that allows organizations to create interactive visualizations and gain valuable insights from their data. To create visualizations and gain insights from the dataset provided by REC corp LTD (product sales and revenue data), you can follow these steps:

Description of the Process:

1. **Data Import:**

Start by importing the dataset into the IBM Cognos environment. The dataset contains eight numerical parameters, including total unit sales (Q1, Q2, Q3, Q4) and total revenue (S1, S2, S3, S4) for each of the four products (P1, P2, P3, P4). Ensure that the dataset is structured appropriately.

2. **Data Preparation:**

Perform data preparation tasks, such as data cleansing, handling missing values, and ensuring data quality. This step is essential to ensure that your visualizations and insights are based on reliable data.

3. **Data Exploration:**

Explore the dataset to get a better understanding of the data distribution, summary statistics, and relationships between variables. You can use IBM Cognos tools to perform data exploration efficiently.

4. **Visualization Creation:**

- Create visualizations to gain insights into product sales and revenue. IBM Cognos provides various chart types, such as bar charts, line charts, and pie charts, which you can use to visualize the data.

- Create bar charts to compare the total unit sales of different products. This can help identify the best-selling product.

- Generate line charts to analyze the sales trends over the ten-year period for each product.
- Create pie charts to visualize the proportion of revenue generated by each product.

5. **Interactive Dashboards:**

Utilize IBM Cognos to build interactive dashboards that allow users to interact with the data. You can add filters, slicers, and drill-through capabilities to explore data at different levels of granularity.

6. **Advanced Analytics:**

Implement advanced analytics techniques, such as forecasting and clustering, to gain deeper insights. For instance, you can forecast future sales trends based on historical data.

7. **Collaboration and Sharing:**

IBM Cognos allows users to collaborate and share insights. You can publish reports, dashboards, and visualizations to share with stakeholders.

8. **Performance Monitoring:**

Continuously monitor the performance of products and sales by setting up key performance indicators (KPIs). You can track metrics like sales growth, revenue, and customer preferences over time.

9. **Automated Reporting:**

Set up automated reporting to generate regular reports and alerts based on specific conditions or thresholds.

Gain Insights:

- Identify the product with the highest sales by analyzing the total unit sales over the ten years.
- Analyze peak sales periods by visualizing sales trends over time.

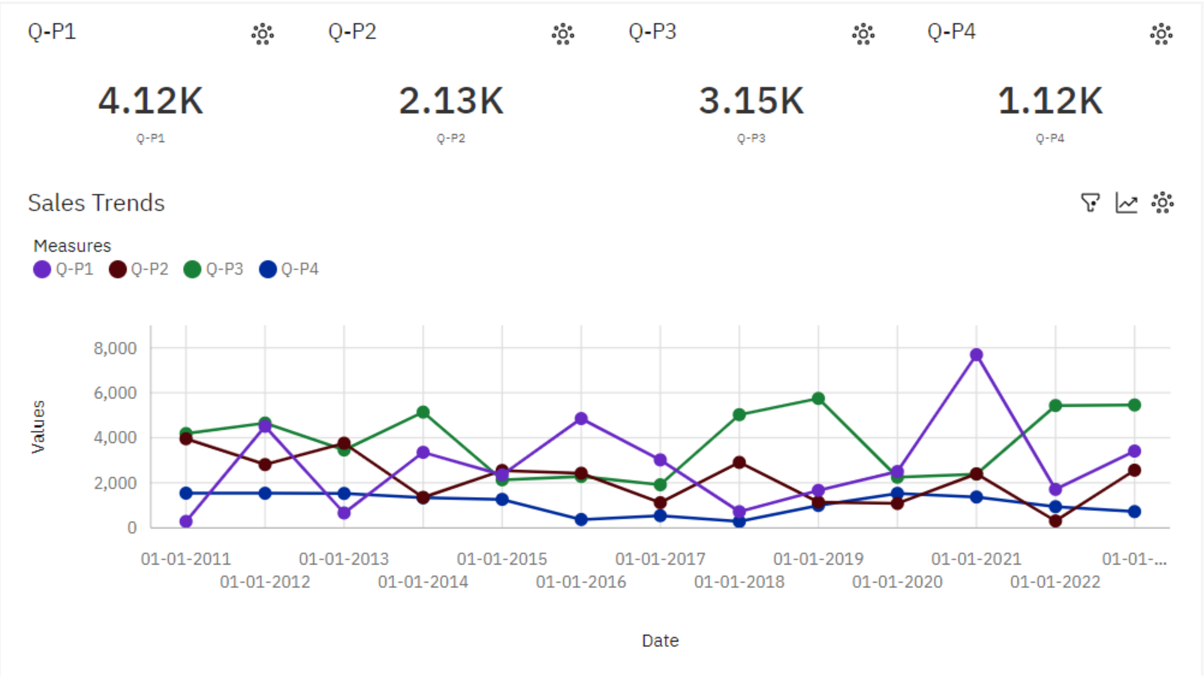
- Gain insights into customer preferences by examining which products contribute the most to total revenue.

- Discover correlations and patterns within the dataset that may inform strategic decisions.

- Use advanced analytics to make forecasts for future sales and revenue.

By following these steps and leveraging the capabilities of IBM Cognos, you can create visualizations and gain valuable insights from the dataset to make informed business decisions.

Dashboard using IBM Cognos



Insights:

Over all dates, the average of Q-P1 is nearly three thousand.

Across all dates, the average of Q-P2 is over two thousand.

Over all dates, the average of Q-P3 is nearly four thousand.

Across all dates, the average of Q-P4 is over a thousand.

The total number of results for Q-P1, across all dates, is 13.

The total number of results for Q-P2, across all dates, is 13.

The total number of results for Q-P3, across all dates, is 13.

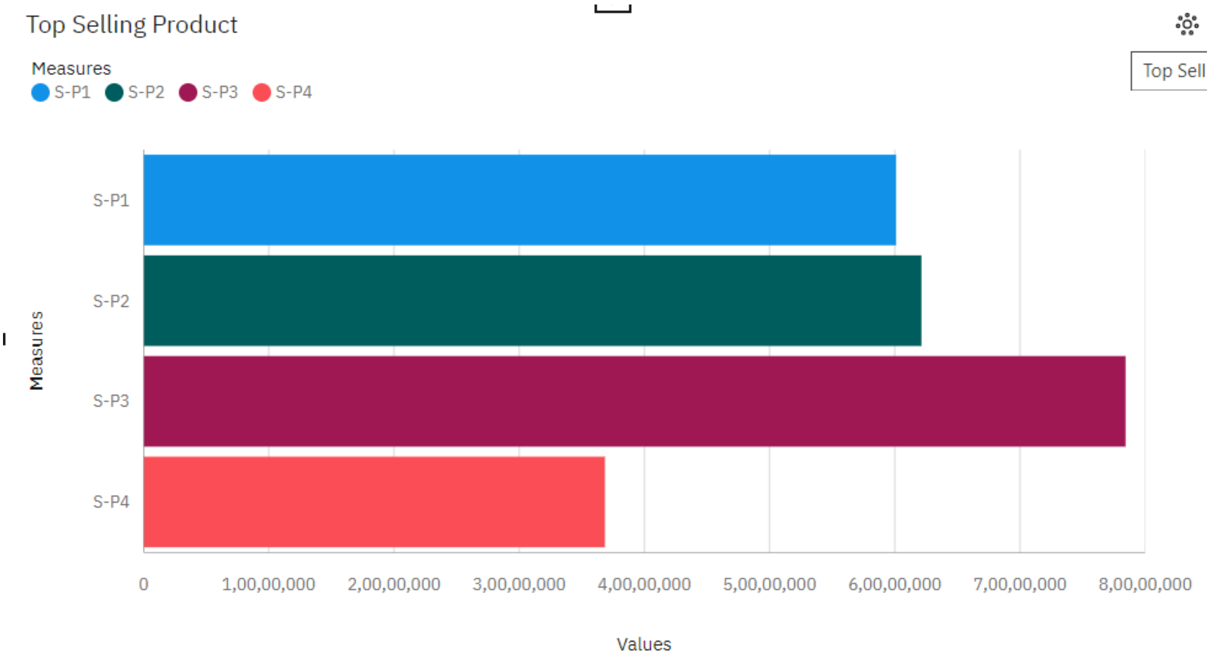
The total number of results for Q-P4, across all dates, is 13.

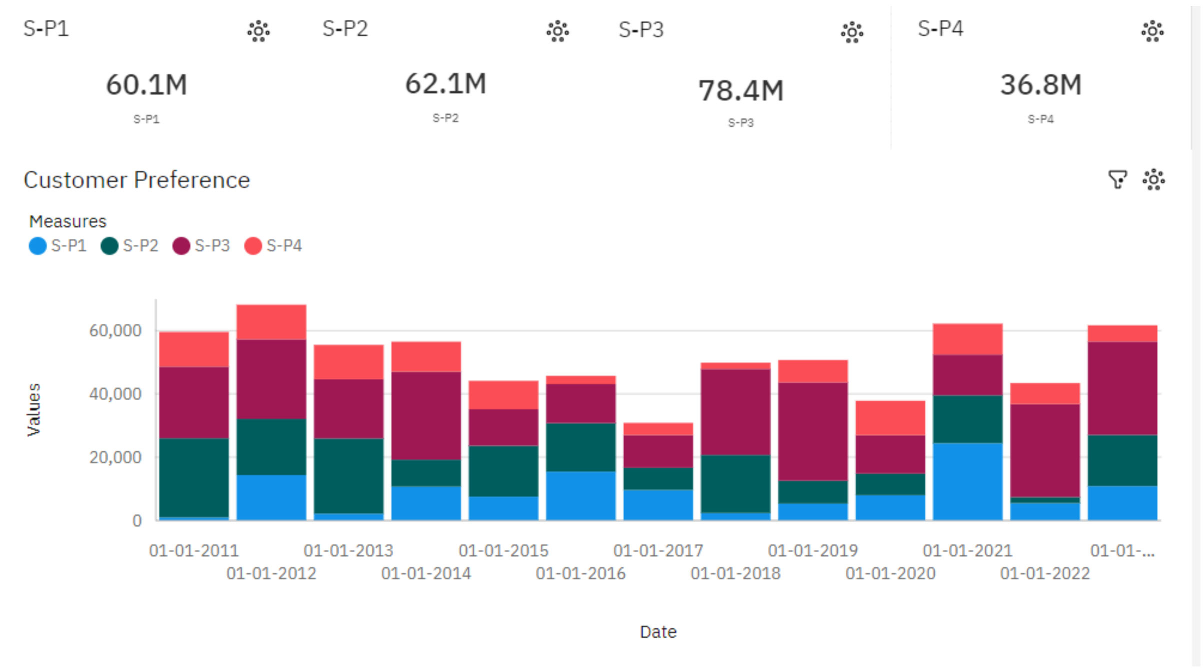
Q-P1 ranges from 281, when Date is 01-01-2011, to over 7500, when Date is 01-01-2021.

Q-P2 ranges from 302, when Date is 01-01-2022, to nearly four thousand, when Date is 01-01-2011.

Q-P3 ranges from nearly two thousand, when Date is 01-01-2017, to over 5500, when Date is 01-01-2019.

Q-P4 ranges from 276, when Date is 01-01-2018, to over 1500, when Date is 01-01-2011.





Insights:

Over all dates, the average of S-P1 is nearly nine thousand.

Over all dates, the average of S-P2 is nearly fourteen thousand.

Over all dates, the average of S-P3 is almost 21 thousand.

Across all dates, the average of S-P4 is over 7500.

The total number of results for S-P1, across all dates, is 13.

The total number of results for S-P2, across all dates, is 13.

The total number of results for S-P3, across all dates, is 13.

The total number of results for S-P4, across all dates, is 13.

S-P1 ranges from 890.8, when Date is 01-01-2011, to over 24 thousand, when Date is 01-01-2021.

S-P2 ranges from nearly two thousand, when Date is 01-01-2022, to over 25 thousand, when Date is 01-01-2011.

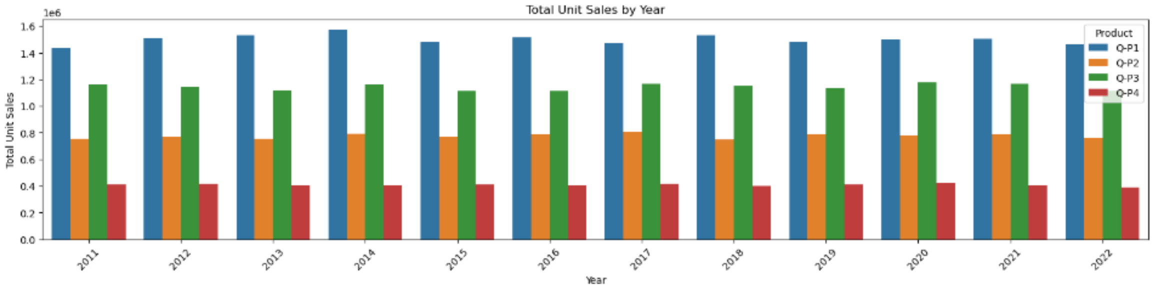
S-P3 ranges from over 10 thousand, when Date is 01-01-2017, to over 31 thousand, when Date is 01-01-2019.

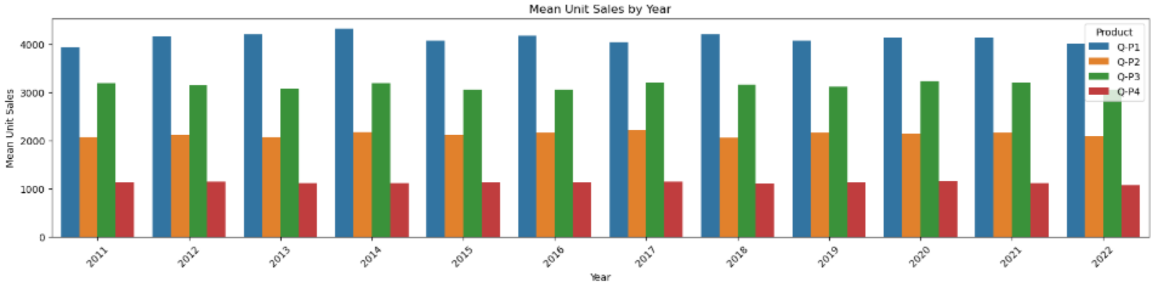
S-P4 ranges from nearly two thousand, when Date is 01-01-2018, to nearly eleven thousand, when Date is 01-01-2011.

DATA VISUALISATION USING PYTHON:

Graph our TOTAL & MEAN unit sold for each product using a histogram.

```
#Create a function that allows us to plot a bar chart for the 4 products
def plot_bar_chart(df, columns, stri, str1, val):
    # Aggregate sales for each product by year, by sum or mean
    if val == 'sum':
        sales_by_year = df.groupby('Year')[columns].sum().reset_index()
    elif val == 'mean':
        sales_by_year = df.groupby('Year')[columns].mean().reset_index()

    # Melt the data to make it easier to plot
    sales_by_year_melted = pd.melt(sales_by_year, id_vars='Year', value_vars=columns, var_name='Product', value_name='Sales')

    # Create a bar chart
    plt.figure(figsize=(20,4))
    sns.barplot(data=sales_by_year_melted, x='Year', y='Sales', hue='Product') #,palette="cividis")
    plt.xlabel('Year')
    plt.ylabel(stri)
    plt.title(f'{stri} by {str1}')
    plt.xticks(rotation=45)
    plt.show()plot_bar_chart(data_reduced, ['Q-P1', 'Q-P2', 'Q-P3', 'Q-P4'],'Total Unit Sales', 'Year', 'sum')

plot_bar_chart(data_reduced, ['Q-P1', 'Q-P2', 'Q-P3', 'Q-P4'],'Mean Unit Sales', 'Year', 'mean')
```



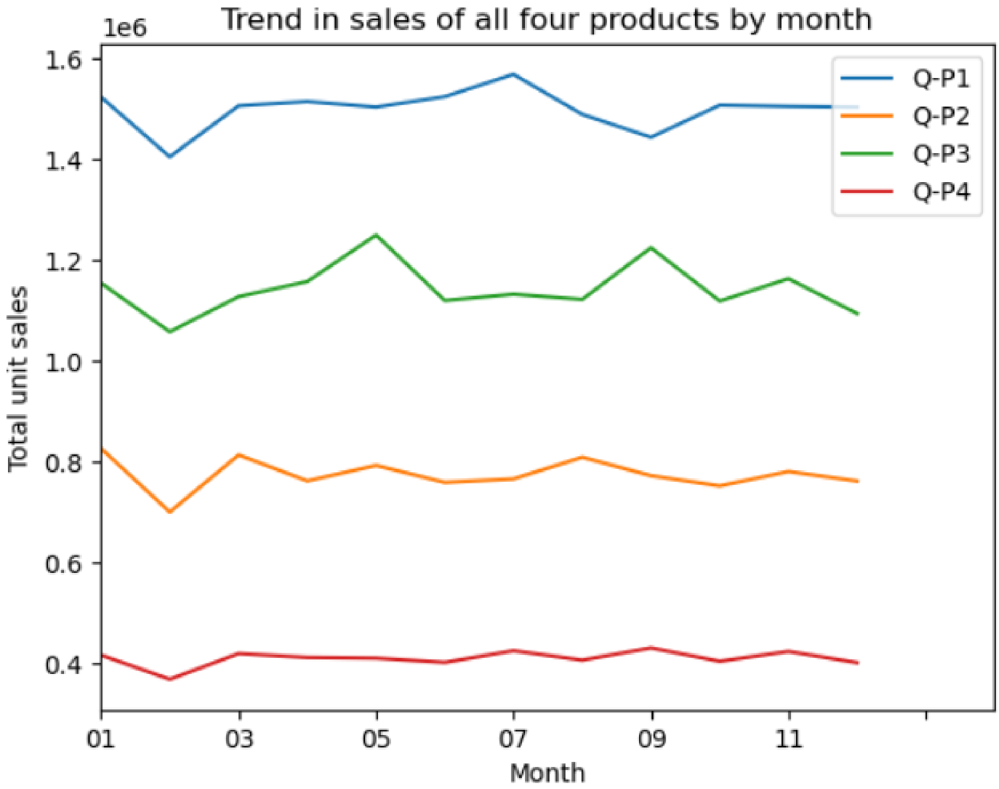


Observation

- We can observe that P1 has the highest unit sales for each year. And it's highest is in year 2014.
- We can observe taht P4 has the lowest unit sales of all the products.

Trend in sales of all four products during certain months

```
# Create a figure and axis

def month_plot():

    fig, ax = plt.subplots()



    # Plot the sales data for each product by month

    data_reduced.groupby('Month')[['Q-P1', 'Q-P2', 'Q-P3', 'Q-P4']].sum().plot(ax=ax)



    # Set the x-axis limits to only show up to December

    ax.set_xlim(left=0, right=13)



    # Set the axis labels and title

    ax.set_xlabel('Month')
    ax.set_ylabel('Total unit sales')

    ax.set_title('Trend in sales of all four products by month')


    # Show the plot

    plt.show()


month_plot()
```



Observation

- We have merged the sales for months 9 and 09.
- We can observe that Feb and Dec have the lowest sales for each product
- For P1 We can observe Mar - Jul having the highest unit sales
- For P2 We can observe Jan, Mar - Aug having the highest unit sales
- For P3 We can observe May & Sep having the highest unit sales
- For P4 We can observe uniform sales from Jan - Dec

Conclusion

Unit Sales 2011 - 2022

- P1 has the highest unit sales for each year. And it's highest is in year 2014.
- We can observe that P4 has the lowest unit sales of all the products.

Revenues 2011 - 2022

- We can observe that P3 brought in the most revenue. This could be as a result of multiple things:
    - P3 was sold for higher than the rest, as it had the second highest unit sales for each year.
- We can observe than P1 and P2 brought in similar revenues for each year. With P2 bringing in slightly more.
- P1 despite having the most unit sold, brought in the second lowest revenue each year.

Average Month Sales 2011 - 2022

- We can observe that all Products unit sales drop in Feb.
- We can observe that Feb and Dec have the lowest sales for each product
- For P1 We can observe Mar - Jul having the highest unit sales
- For P2 We can observe Jan, Mar - Aug having the highest unit sales
- For P3 We can observe May & Sep having the highest unit sales
- For P4 We can observe uniform sales from Jan - Dec

Estimated Unit Sales for 31st of Dec

This value can not be properly estimated with out Machine Learning. Currently we used the average for all the 31st days across all years for each product.

- Overall we can see that P1 has the highest unit sales on the 31st for each year, except for 2021 and 2022. (These could be as a result to Covid and other economy issues.)
- P3 has the second highest unit sales for all the 31st in each year.
- We can see that our previous observation correlate as Q-P1 has the higest estimate, followed by Q-P3
- We can approxiamte that the company will make:
    - Q-P1: 3813.74
    - Q-P2: 2058.80
    - Q-P3: 3183.88
    - Q-P4: 1098.61