PRODUCT SALES ANALYSIS

Problem Definition

Consider incorporating machine learning algorithms to predict future sales trends or customer behaviors

Data visualization:

Data visualization plays a crucial role in data analysis, enabling the extraction of meaningful insights from complex datasets. This abstract outlines the process of creating informative and visually appealing data visualizations using Python's Matplotlib library within the Jupyter Anaconda environment.

The dataset chosen for this project is various parameter used for analysing water quality for domestic purpose, and the goal is to effectively communicate key trends, patterns, and relationships within the data. Matplotlib, a versatile and widely used data visualization library, offers a wide range of tools for creating various types of plots, such as line charts, bar graphs, scatter plots, heatmaps, and more.

In this project, we will walk through the step-by-step process of data visualization, including data pre-processing, selecting appropriate plot types, customizing aesthetics (colours, labels, legends, etc.), and adding context to the visualizations with titles, annotations, and captions. We will also explore how to handle common data visualization challenges, such as handling missing data, creating subplots, and generating interactive visualizations.

In [3]:
```
import pandas as pd
import numpy as ny
from statsmodels.tsa.arima.model import ARIMA
```

In [4]:
```
df=pd.read_csv('ste.csv')
df
```

Out[4]:

	Unnamed: 0	Date	Q-P1	Q-P2	Q-P3	Q-P4	S-P1	S-P2	S-P3	S-P4
0	0	13-06-2010	5422	3725	576	907	17187.74	23616.50	3121.92	6466.91
1	1	14-06-2010	7047	779	3578	1574	22338.99	4938.86	19392.76	11222.62
2	2	15-06-2010	1572	2082	595	1145	4983.24	13199.88	3224.90	8163.85
3	3	16-06-2010	5657	2399	3140	1672	17932.69	15209.66	17018.80	11921.36
4	4	17-06-2010	3668	3207	2184	708	11627.56	20332.38	11837.28	5048.04
...	...	...	...	...	...	...	...	...	...	...
4595	4595	30-01-2023	2476	3419	525	1359	7848.92	21676.46	2845.50	9689.67
4596	4596	31-01-2023	7446	841	4825	1311	23603.82	5331.94	26151.50	9347.43
4597	4597	01-02-2023	6289	3143	3588	474	19936.13	19926.62	19446.96	3379.62
4598	4598	02-02-2023	3122	1188	5899	517	9896.74	7531.92	31972.58	3686.21
4599	4599	03-02-2023	1234	3854	2321	406	3911.78	24434.36	12579.82	2894.78

4600 rows × 10 columns

In [5]:
```
from sklearn.linear_model import LinearRegression
lin_model=LinearRegression()
```

In [6]:
```
from sklearn.ensemble import RandomForestRegressor
model=RandomForestRegressor(n_estimators=100,max_features=3, random_state=1)
```

In [7]:
```
import numpy as np
x1,x2,x3,y=df['Q-P1'],df['Q-P2'],df['Q-P3'],df['Q-P4']
x1,x2,x3,y=np.array(x1),np.array(x2),np.array(x3),np.array(y)
x1,x2,x3,y=x1.reshape(-1,1),x2.reshape(-1,1),x3.reshape(-1,1),y.reshape(-1,1)
final_x=np.concatenate((x1,x2,x3),axis=1)
print(final_x)
```

```
[[5422 3725  576]
 [7047  779 3578]
 [1572 2082  595]
 ...
 [6289 3143 3588]
 [3122 1188 5899]
 [1234 3854 2321]]
```

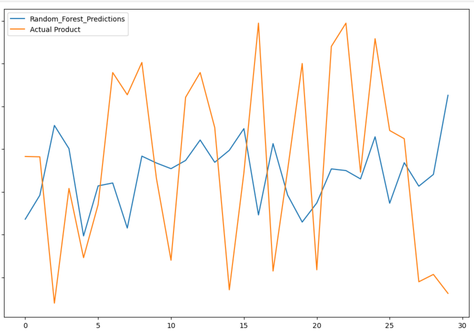
In [8]:
```
X_train,X_test,y_train,y_test=final_x[:-30],final_x[-30:],y[:-30],y[-30:]
```

In [9]:
```
model.fit(X_train,y_train)
lin_model.fit(X_train,y_train)
```

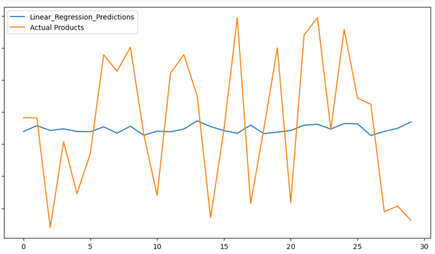
```
C:\Users\Dell\anaconda3\envs\fre\lib\site-packages\sklearn\base.py:1151: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples,), for example using ravel().
  return fit_method(estimator, *args, **kwargs)
```

Out[9]:
```
LinearRegression()
```

In [10]:
```
pred=model.predict(X_test)
import matplotlib.pyplot as plt
plt.rcParams["figure.figsize"] = (12,8)
plt.plot(pred,label='Random_Forest_Predictions')
plt.plot(y_test,label='Actual Product')
plt.legend(loc="upper left")
plt.show()
```



In [11]:
```
lin_pred=lin_model.predict(X_test)
import matplotlib.pyplot as plt
plt.rcParams["figure.figsize"] = (11,6)
plt.plot(lin_pred,label='Linear_Regression_Predictions')
plt.plot(y_test,label='Actual Products')
plt.legend(loc="upper left")
plt.show()
```



In [12]:
```
import numpy as np
x1,x2,x3,y=df['S-P1'],df['S-P2'],df['S-P3'],df['S-P4']
x1,x2,x3,y=np.array(x1),np.array(x2),np.array(x3),np.array(y)
x1,x2,x3,y=x1.reshape(-1,1),x2.reshape(-1,1),x3.reshape(-1,1),y.reshape(-1,1)
final_x=np.concatenate((x1,x2,x3),axis=1)
print(final_x)
```

```
[[17187.74 23616.5   3121.92]
 [22338.99  4938.86 19392.76]
 [ 4983.24 13199.88  3224.9 ]
 ...
 [19936.13 19926.62 19446.96]
 [ 9896.74  7531.92 31972.58]
 [ 3911.78 24434.36 12579.82]]
```

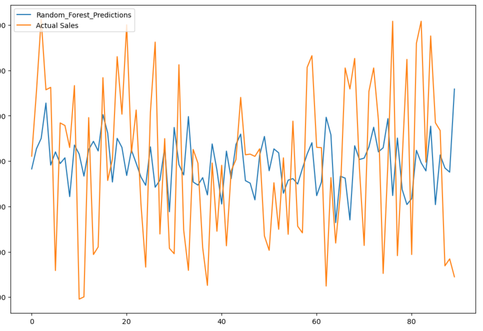
In [13]:
```
X_train,X_test,y_train,y_test=final_x[:-90],final_x[-90:],y[:-90],y[-90:]
```

In [14]:
```
model.fit(X_train,y_train)
lin_model.fit(X_train,y_train)
```

```
C:\Users\Dell\anaconda3\envs\fre\lib\site-packages\sklearn\base.py:1151: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples,), for example using ravel().
  return fit_method(estimator, *args, **kwargs)
```

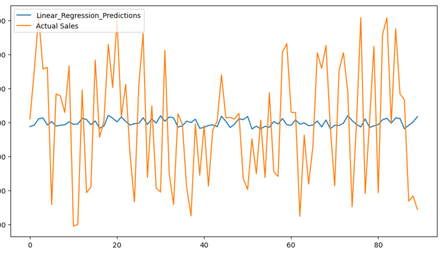
Out[14]:
```
LinearRegression()
```

In [15]:
```
pred=model.predict(X_test)
import matplotlib.pyplot as plt
plt.rcParams["figure.figsize"] = (12,8)
plt.plot(pred,label='Random_Forest_Predictions')
plt.plot(y_test,label='Actual Sales')
plt.legend(loc="upper left")
plt.show()
```



In [32]:
```
from sklearn.metrics import mean_squared_error
from math import sqrt
rmse_rf=sqrt(mean_squared_error(pred,y_test))
rmse_lr=sqrt(mean_squared_error(lin_pred,y_test))
```

In [19]:
```
lin_pred=lin_model.predict(X_test)
import matplotlib.pyplot as plt
plt.rcParams["figure.figsize"] = (11,6)
plt.plot(lin_pred,label='Linear_Regression_Predictions')
plt.plot(y_test,label='Actual Sales')
plt.legend(loc="upper left")
plt.show()
```



In [ ]:
```
```

In [ ]:
```
```