Multiple Linear Regression

Import libraries

In [ ]:

```
import pandas as pd
import numpy as np
from sklearn.linear_model import LinearRegression
```

Import your Data

In [ ]:

```
df=pd.read_csv('ml_data_salary.csv')
df.head()
```

Out[ ]:

	age	distance	YearsExperience	Salary
0	31.1	77.75	1.1	39343
1	31.3	78.25	1.3	46205
2	31.5	78.75	1.5	37731
3	32.0	80.00	2.0	43525
4	32.2	80.50	2.2	39891

In [ ]:

```
x = df[['age','distance', 'YearsExperience']]
x.head()
```

Out[ ]:

	age	distance	YearsExperience
0	31.1	77.75	1.1
1	31.3	78.25	1.3
2	31.5	78.75	1.5
3	32.0	80.00	2.0
4	32.2	80.50	2.2

In [ ]:

```
y = df[['Salary']]
y.head()
```

Out[ ]:

	Salary
0	39343
1	46205
2	37731
3	43525
4	39891

Create and fit the model

In [ ]:

```
model = LinearRegression().fit(x,y)
model
```

Out[ ]:

```
▾ LinearRegression
LinearRegression()
```

In [ ]:

```
model.coef_
```

Out[ ]:

```
array([[-2.79782201e+15,  1.10953700e+15,  2.39795093e+13]])
```

In [ ]:

```
model.intercept_
```

Out[ ]:

```
array([7.19385278e+14])
```

In [ ]:

```
model.predict([[33.1, 77.75, 1.1]])
```

```
c:\Users\Dell\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\base.
py:450: UserWarning: X does not have valid feature names, but LinearRegression was fi
tted with feature names
  warnings.warn(
```

Out[ ]:

```
array([-5.59564401e+15])
```

In [ ]:

```
reg = LinearRegression().fit(x_test, y_test)
print("Test Score: ", reg.score(x_test, y_test))
print("Train Score: ", reg.score(x_train, y_train))
```

```
Test Score:  0.9933289271009905
Train Score:  0.930628181708933
```

In [ ]:

```
model.score(x,y)
```

Out[ ]:

```
0.9569516748144675
```

In [ ]:

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train, y_test= train_test_split(x,y, test_size=0.2, random_state=0)
from sklearn.linear_model import LinearRegression
model= LinearRegression().fit(x_train, y_train)
model
model.predict(x_test)
model.predict([[31.1,77.75,1.1]])
```

```
c:\Users\Dell\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\base.
py:450: UserWarning: X does not have valid feature names, but LinearRegression was fi
tted with feature names
  warnings.warn(
```

Out[ ]:

```
array([[37024.]])
```

In [ ]:

```
import matplotlib.pyplot as plt
plt.scatter(x_test, y_test)
plt.plot(x_test, model.predict(x_test), color="green")
plt.xlabel("Experience")
plt.ylabel("Salary")
plt.title("Test Plot")
plt.show()
```

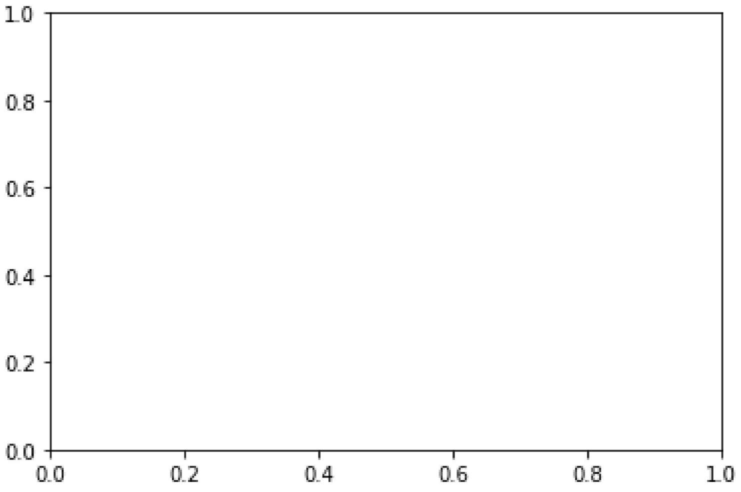
```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
c:\Users\Dell\Desktop\New folder\ML_03\ml_4.ipynb Cell 16 in <cell line: 2>()
      <a href='vscode-notebook-cell:/c%3A/Users/Dell/Desktop/New%20folder/ML_03/ml_4.
ipynb#ch0000015?line=0'>1</a> import matplotlib.pyplot as plt
----> <a href='vscode-notebook-cell:/c%3A/Users/Dell/Desktop/New%20folder/ML_03/ml_4.
ipynb#ch0000015?line=1'>2</a> plt.scatter(x_test, y_test)
      <a href='vscode-notebook-cell:/c%3A/Users/Dell/Desktop/New%20folder/ML_03/ml_4.
ipynb#ch0000015?line=2'>3</a> plt.plot(x_test, model.predict(x_test), color="green")
      <a href='vscode-notebook-cell:/c%3A/Users/Dell/Desktop/New%20folder/ML_03/ml_4.
ipynb#ch0000015?line=3'>4</a> plt.xlabel("Experience")

File c:\Users\Dell\AppData\Local\Programs\Python\Python310\lib\site-packages\matplotl
ib\pyplot.py:2819, in scatter(x, y, s, c, marker, cmap, norm, vmin, vmax, alpha, line
widths, edgecolors, plotnonfinite, data, **kwargs)
   2814 @_copy_docstring_and_deprecators(Axes.scatter)
   2815 def scatter(
   2816         x, y, s=None, c=None, marker=None, cmap=None, norm=None,
   2817         vmin=None, vmax=None, alpha=None, linewidths=None, *,
   2818         edgecolors=None, plotnonfinite=False, data=None, **kwargs):
-> 2819     __ret = gca().scatter(
   2820         x, y, s=s, c=c, marker=marker, cmap=cmap, norm=norm,
   2821         vmin=vmin, vmax=vmax, alpha=alpha, linewidths=linewidths,
   2822         edgecolors=edgecolors, plotnonfinite=plotnonfinite,
   2823         **({"data": data} if data is not None else {}), **kwargs)
   2824     sci(__ret)
   2825     return __ret

File c:\Users\Dell\AppData\Local\Programs\Python\Python310\lib\site-packages\matplotl
ib\__init__.py:1412, in _preprocess_data.<locals>.inner(ax, data, *args, **kwargs)
   1409 @functools.wraps(func)
   1410 def inner(ax, *args, data=None, **kwargs):
   1411     if data is None:
-> 1412         return func(ax, *map(sanitize_sequence, args), **kwargs)
   1414     bound = new_sig.bind(ax, *args, **kwargs)
   1415     auto_label = (bound.arguments.get(label_namer)
   1416                   or bound.kwargs.get(label_namer))

File c:\Users\Dell\AppData\Local\Programs\Python\Python310\lib\site-packages\matplotl
ib\axes\_axes.py:4362, in Axes.scatter(self, x, y, s, c, marker, cmap, norm, vmin, vm
ax, alpha, linewidths, edgecolors, plotnonfinite, **kwargs)
   4360 y = np.ma.ravel(y)
   4361 if x.size != y.size:
-> 4362     raise ValueError("x and y must be the same size")
   4364 if s is None:
   4365     s = (20 if rcParams['_internal.classic_mode'] else
   4366          rcParams['lines.markersize'] ** 2.0)

ValueError: x and y must be the same size
```


1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0