UNIT 3 - POWER BI

TEJAS MISHRA – 2021UCS1587
SHRISHIR SRIVATSA – 2021UCS1590
PRAJWAL BHAT – 2021UCS1609

INTRODUCTION AND MOTIVATION

In the dynamic landscape of data-driven decision-making, the ability to extract meaningful insights from complex datasets is a critical skill. As organizations continue to amass vast amounts of data, the need for robust tools and methodologies to analyse and visualize this information becomes paramount. harnessing the power of your information is crucial for gaining a competitive edge and making informed decisions. Microsoft Power BI, a versatile business intelligence tool designed to transform your data into actionable insights. Power BI helps to simplify data, it improves its accessibility and provides an AI powered insight. It is a part of Microsoft ecosystem.

Visualizations play a crucial role in conveying complex information in a comprehensible manner. Through compelling charts, graphs, and dashboards, data scientists and analysts can communicate findings to stakeholders more effectively. Power BI, particularly with its ability to offer insights, offers sophisticated and customizable visualization options and makes it easier for improved decision making skills.

From sales figures to customer demographics, Power BI empowers to:

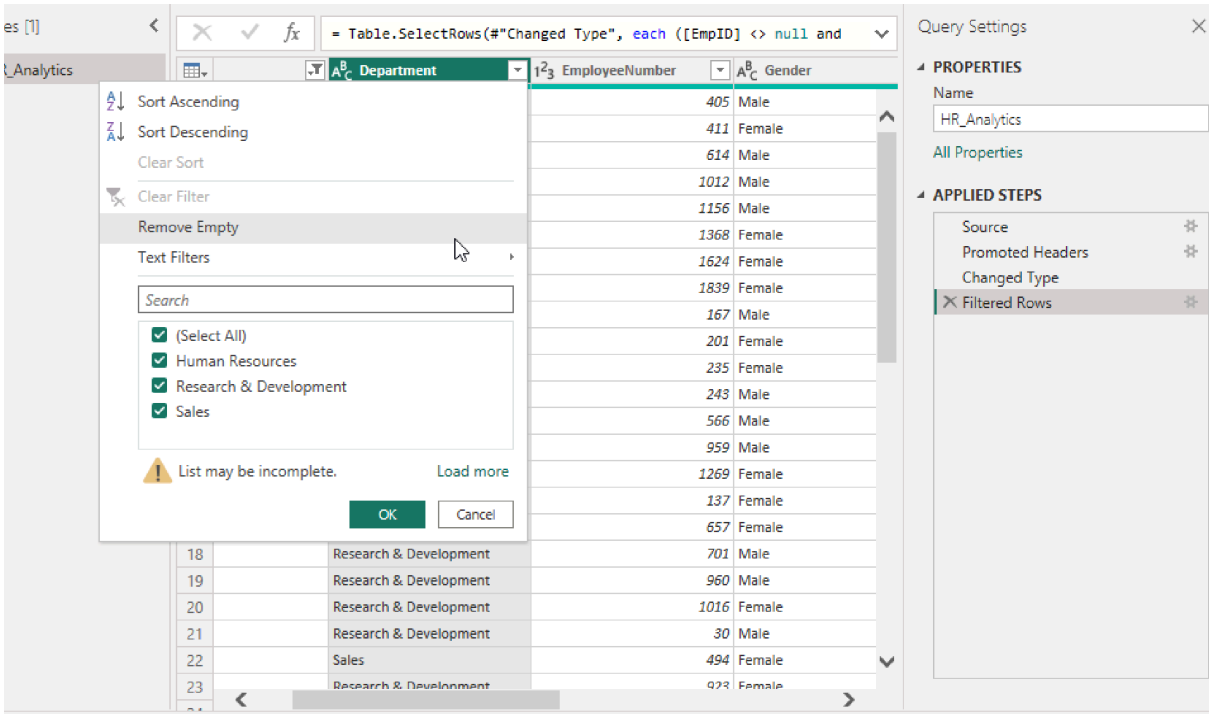
- Gain a holistic view of a business performance.
- Identify areas for improvement and make data-backed decisions.
- Spot emerging trends and stay ahead of the competition.
- Communicate insights effectively to stakeholders through visually compelling reports.
- Power BI seamlessly integrates with other Microsoft products like Excel, Teams, and Dynamics 365, streamlining workflow and to maintain security and protect data within the ecosystem.
- Built-in artificial intelligence features that automatically identify patterns and suggest to make better data-driven predictions.

Software Modules

- **Power BI** - The business intelligence powerhouse from Microsoft, transforms your scattered data into interactive dashboards and impactful reports. Imagine clear insights revealed through eye-catching charts, maps, and graphs, empowering you to make data-driven decisions with confidence. This user-friendly platform, even offering a free version, seamlessly integrates with your Microsoft ecosystem and is mobile-ready for on-the-go brilliance. Unleash the hidden potential within your data and propel your business forward with the power of Power BI.

- **Power Query Editor –** Its workspace where you can connect to diverse data sources, grab information from anywhere, and reshape it to your liking. That's the magic of Power Query Editor, a built-in tool in Power BI and Excel. Drag, drop, and click your way through intuitive transformations, cleaning, merging, and sculpting your data until it's perfectly tailored for analysis.

Removing rows with empty values of gender and department



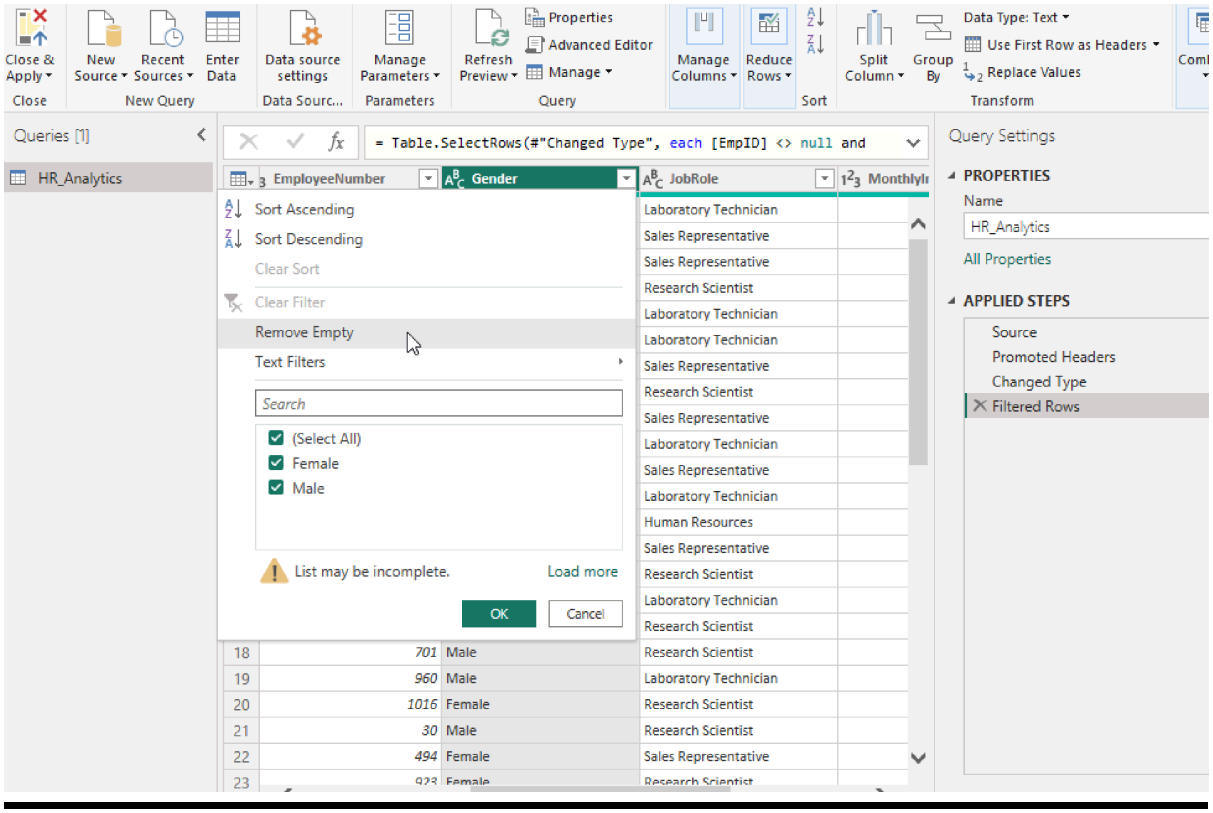
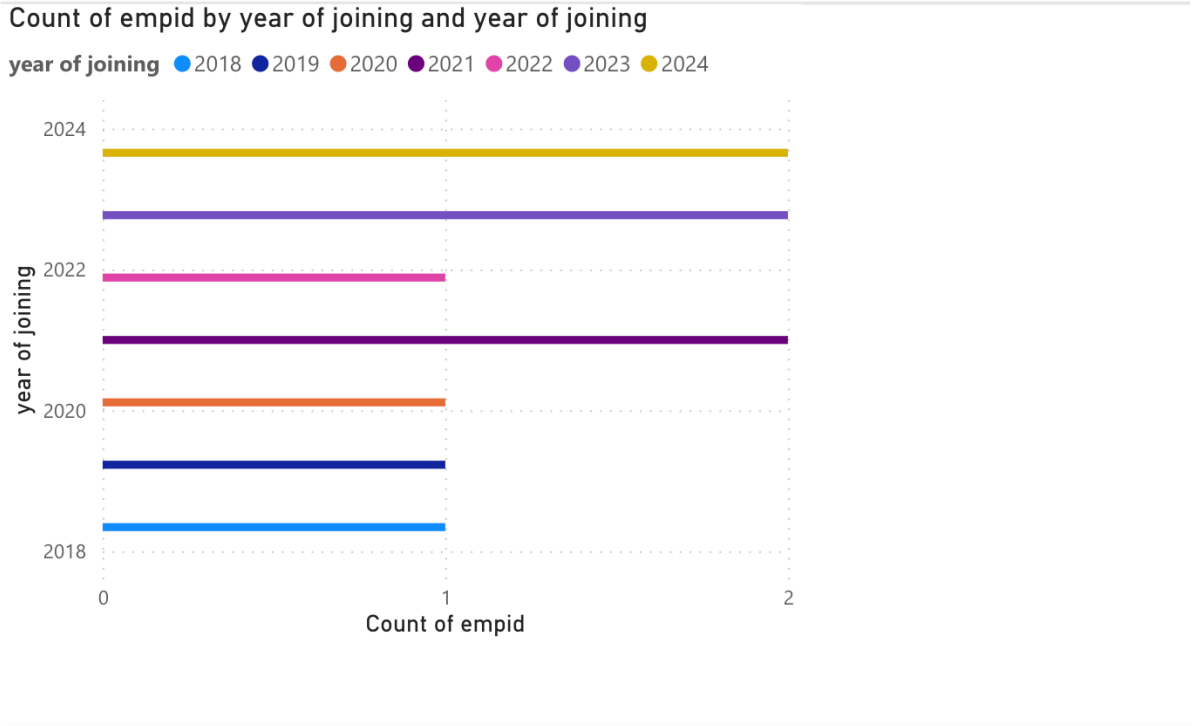


Table Created after removing rows with empty values

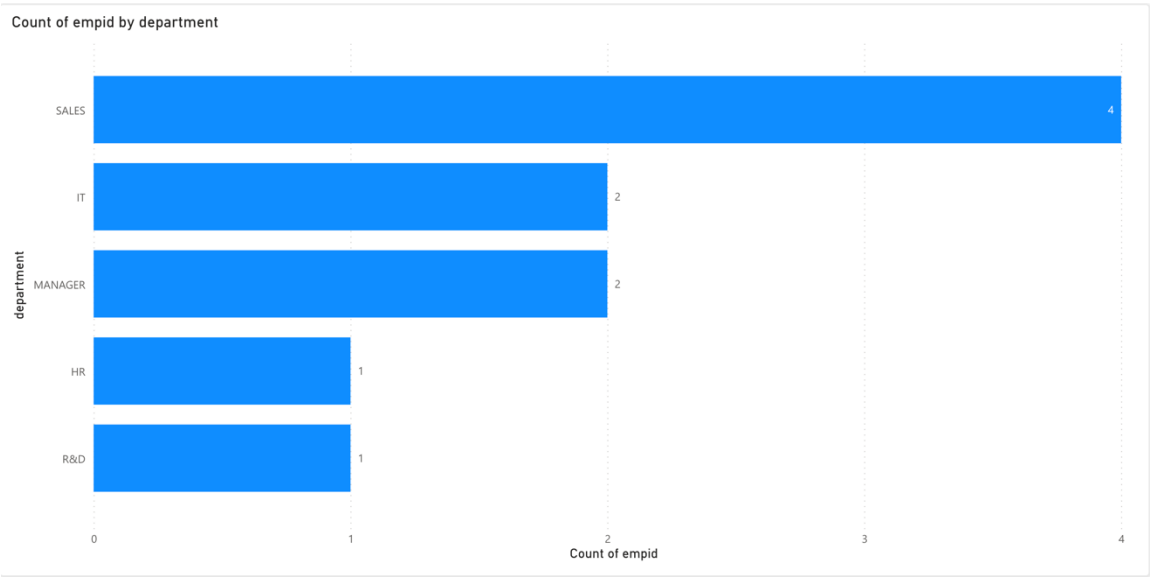
Table

empid	gender	department	salary	year of joining
E1	M	MANAGER	60000	2024
E10	M	IT	25000	2018
E2	M	SALES	30000	2023
E3	F	SALES	25000	2022
E4	F	HR	50000	2021
E5	F	SALES	40000	2019

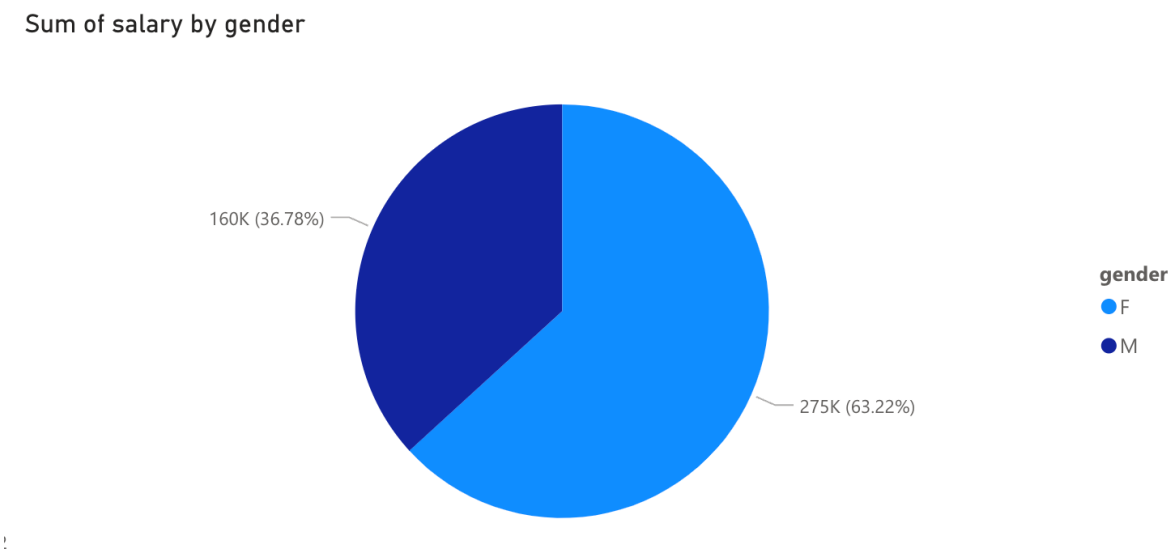
Extract year_of_joining column and visualize number of employees w.r.t year of experience in the company.



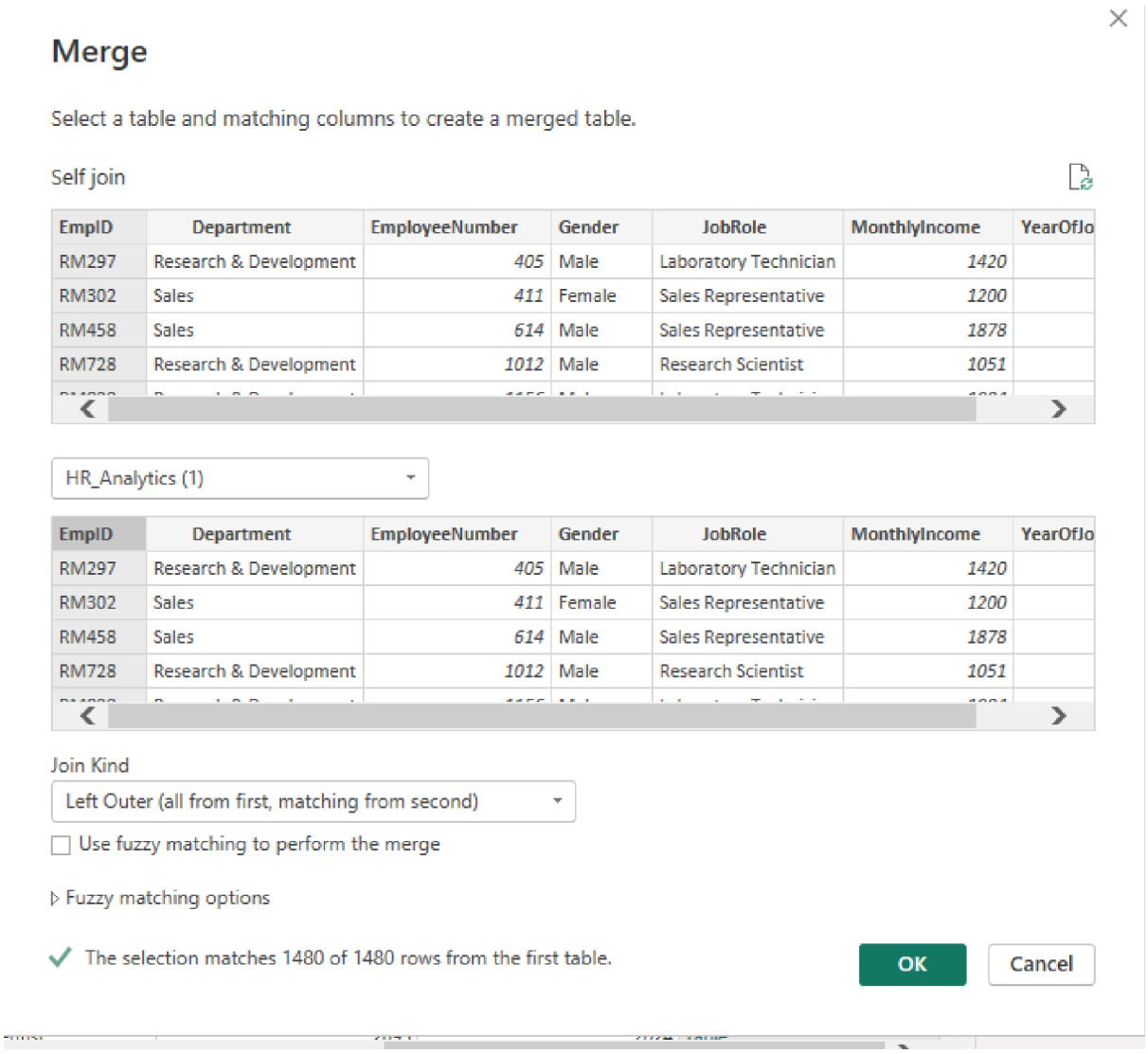
Count of employee per department

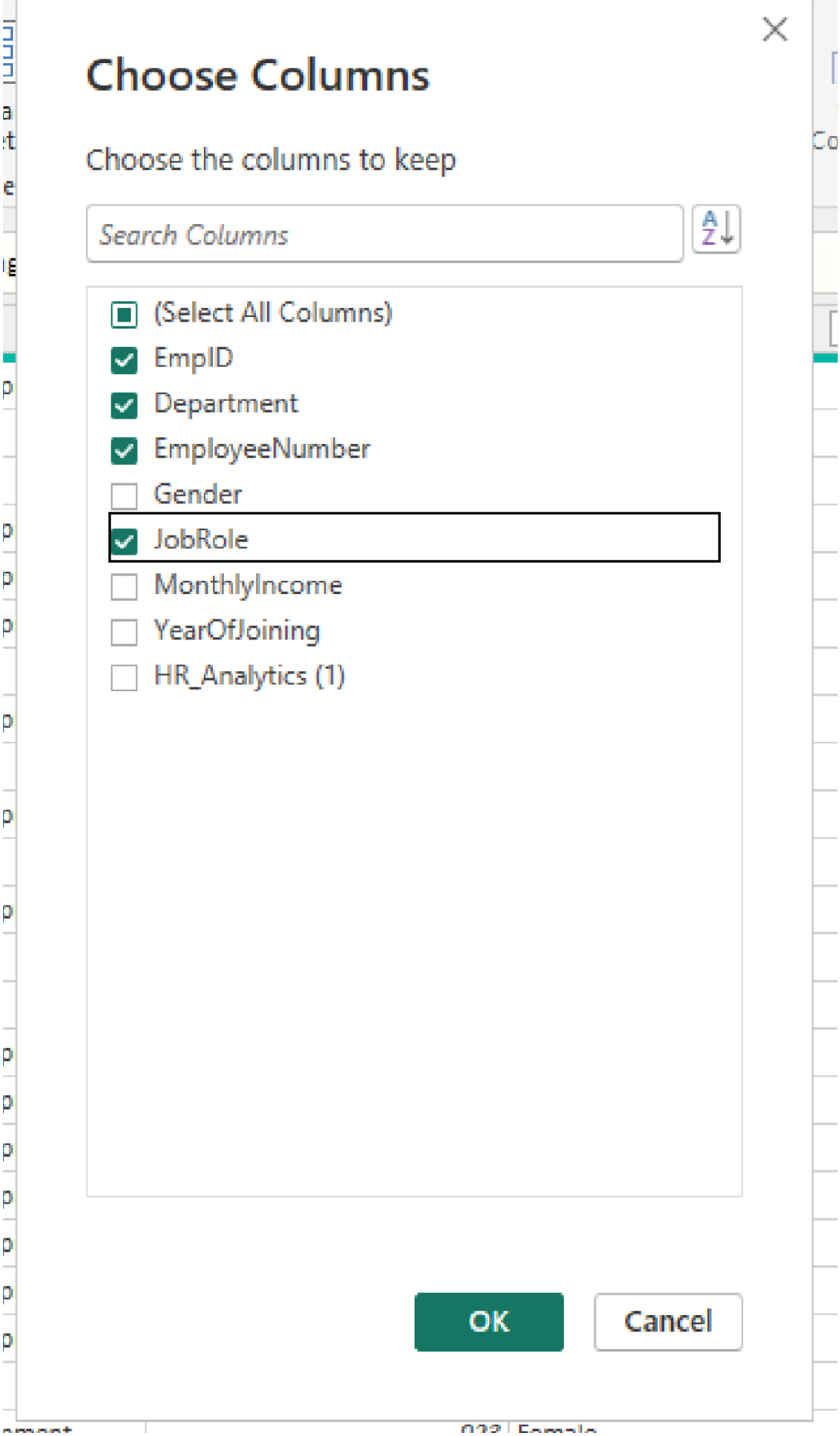


Aggregating Salary with gender



Performing self-join using Power Query.

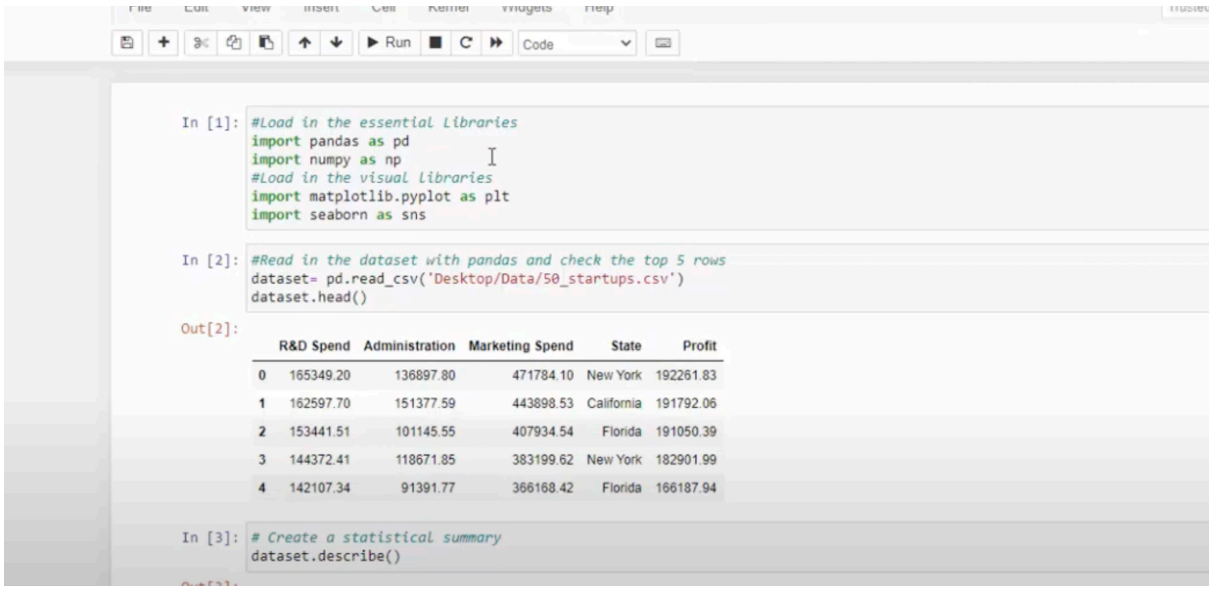



Choose Columns
Choose the columns to keep
Search Columns
(Select All Columns)
EmpID
Department
EmployeeNumber
Gender
JobRole
MonthlyIncome
YearOfJoining
HR_Analytics (1)
OK
Cancel

Visualize the result of any Machine Learning algorithm on any dataset of your choice in PowerBI.

We will be predicting the profit of an organization using the Linear regression machine learning model.

Adding essential libraries and preparing the datset



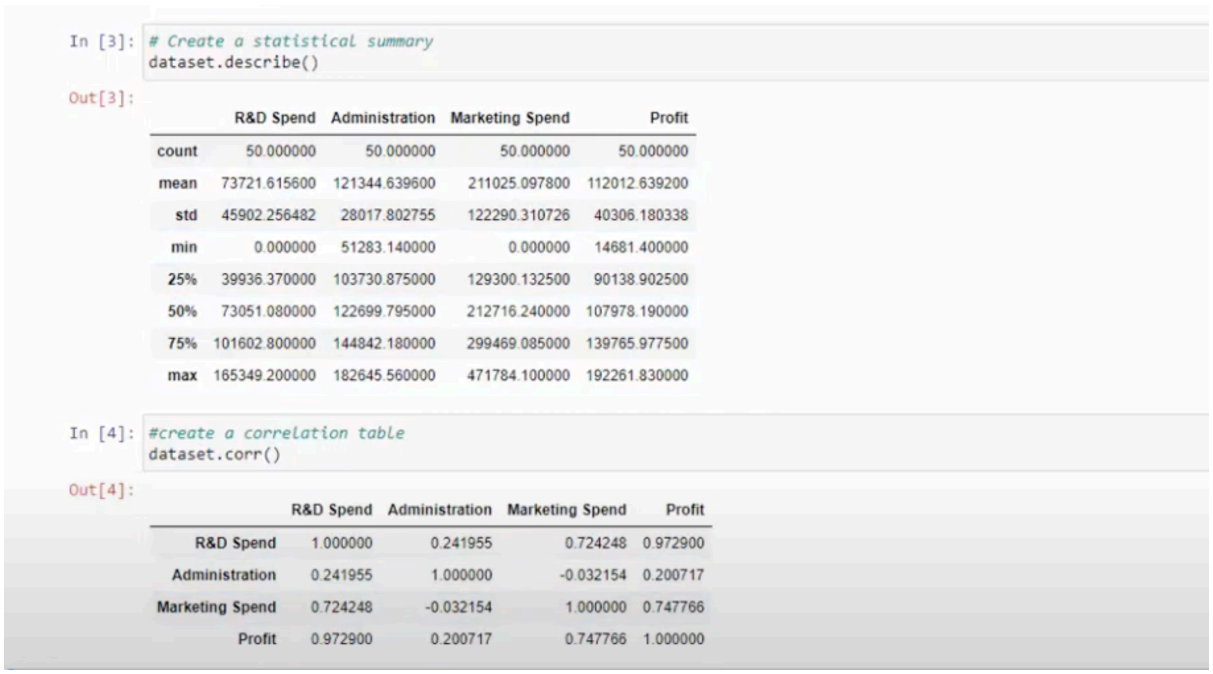
```
In [1]: #Load in the essential libraries
        import pandas as pd
        import numpy as np
        #Load in the visual libraries
        import matplotlib.pyplot as plt
        import seaborn as sns

In [2]: #Read in the dataset with pandas and check the top 5 rows
        dataset= pd.read_csv('Desktop/Data/50_startups.csv')
        dataset.head()
```

Out[2]:

	R&D Spend	Administration	Marketing Spend	State	Profit
0	165349.20	136897.80	471784.10	New York	192261.83
1	162597.70	151377.59	443898.53	California	191792.06
2	153441.51	101145.55	407934.54	Florida	191050.39
3	144372.41	118671.85	383199.62	New York	182901.99
4	142107.34	91391.77	366168.42	Florida	166187.94

```
In [3]: # Create a statistical summary
        dataset.describe()
```

Creating a statistical summary and finding correlation between the features.



```
In [3]: # Create a statistical summary
        dataset.describe()
```

Out[3]:

	R&D Spend	Administration	Marketing Spend	Profit
count	50.000000	50.000000	50.000000	50.000000
mean	73721.615600	121344.639600	211025.097800	112012.639200
std	45902.256482	28017.802755	122290.310726	40306.180338
min	0.000000	51283.140000	0.000000	14681.400000
25%	39936.370000	103730.875000	129300.132500	90138.902500
50%	73051.080000	122699.795000	212716.240000	107978.190000
75%	101602.800000	144842.180000	299469.085000	139765.977500
max	165349.200000	182645.560000	471784.100000	192261.830000

```
In [4]: #create a correlation table
        dataset.corr()
```

Out[4]:

	R&D Spend	Administration	Marketing Spend	Profit
R&D Spend	1.000000	0.241955	0.724248	0.972900
Administration	0.241955	1.000000	-0.032154	0.200717
Marketing Spend	0.724248	-0.032154	1.000000	0.747766
Profit	0.972900	0.200717	0.747766	1.000000

Checking for NULL values and removing them if found.

```
In [6]: #Look for null values
        dataset.isnull().sum()
```

Out[6]:

	R&D Spend	Administration	Marketing Spend	State	Profit
0	False	False	False	False	False
1	False	False	False	False	False
2	False	False	False	False	False
3	False	False	False	False	False
4	False	False	False	False	False
5	False	False	False	False	False
6	False	False	False	False	False
7	False	False	False	False	False
8	False	False	False	False	False
9	False	False	False	False	False
10	False	False	False	False	False

Encoding the string or state columns

```
In [7]: #count the string categories
        dataset['State'].value_counts()
```

Out[7]:

```
California    17
New York      17
Florida       16
Name: State, dtype: int64
```

```
In [9]: #encode the string or state columns
        pd.get_dummies(dataset,drop_first=True)
```

Out[9]:

	R&D Spend	Administration	Marketing Spend	Profit	State_Florida	State_New York
0	165349.20	136897.80	471784.10	192261.83	0	1
1	162597.70	151377.59	443898.53	191792.06	0	0
2	153441.51	101145.55	407934.54	191050.39	1	0
3	144372.41	118671.85	383199.62	182901.99	0	1
4	142107.34	91391.77	366168.42	166187.94	1	0
5	131876.90	99814.71	362861.36	156991.12	0	1
6	134615.46	147198.87	127716.82	156122.51	0	0
7	130298.13	145530.06	323876.68	155752.60	1	0
8	120542.52	148718.95	311613.29	152211.77	0	1
9	123334.88	108679.17	304981.62	149759.96	0	0
10	101913.08	110594.11	229160.95	146121.95	1	0
11	100671.96	91790.61	249744.55	144259.40	0	0
12	93863.75	127320.38	249839.44	141585.52	1	0
13	91992.39	135495.07	252664.93	134307.35	0	0
14	119943.24	156547.42	256512.92	132602.65	1	0
15	114523.61	122616.84	261776.23	129917.04	0	1
16	78013.11	121597.55	264346.06	126992.93	0	0

Loading ML libraries

```
In [11]: # Load in the ML libraries
         from sklearn.model_selection import train_test_split
         from sklearn.linear_model import LinearRegression
         from sklearn.metrics import r2_score, mean_squared_error
```

Creating feature set and splitting data into training and testing data

```
In [14]: #X and Y
         X = dataset[features]
         y = dataset[target]
```

```
In [15]: #create a training and test set
         X_train,X_test,y_train,y_test= train_test_split(X,y,test_size=0.2,random_state=123)
```

Out[15]:

	R&D Spend	Administration	Marketing Spend	State_Florida	State_New York
21	78389.47	153773.43	299737.29	0	1
47	0.00	135426.92	0.00	0	0
11	100671.96	91790.61	249744.55	0	0
41	27892.92	84710.77	164470.71	1	0
5	131876.90	99814.71	362861.36	0	1
1	162597.70	151377.59	443898.53	0	0
6	134615.46	147198.87	127716.82	0	0
27	72107.60	127864.55	353183.81	0	1
49	0.00	116983.80	45173.06	0	0
24	77044.01	99281.34	140574.81	0	1
31	61136.38	152701.92	88218.23	0	1
15	114523.61	122616.84	261776.23	0	1
35	46014.02	85047.44	205517.64	0	1
26	75328.87	144135.98	134050.07	1	0
7	130298.13	145530.06	323876.68	1	0
20	76253.86	113867.30	298664.47	0	0
48	542.05	51743.15	0.00	0	1

Fitting data into our model

```
In [17]: #fit the data to our ML model
         lin = LinearRegression()
         lin.fit(X_train,y_train)
         y_pred = lin.predict(X_test)
         y_pred
```

```
Out[17]: array([[133749.91948852],
                [126771.56418161],
                [ 97712.50105   ],
                [ 58138.82512327],
                [128196.53673201],
                [192274.03929239],
                [ 75126.75206534],
                [127984.52000746],
                [101453.65842151],
                [151532.50862832]])
```

Testing the data

```
In [18]: y_test
```

Out[18]:

	Profit
10	146121.95
13	134307.35
30	99937.59
46	49490.75
18	124266.90

Checking accuracy of model

```
In [19]: #check the model performance
         r2_score(y_test,y_pred)
```

```
Out[19]: 0.9667998486975283
```

Creating dataset to run in PowerBI

```
In [20]: dataset.insert(4,"Predictions", lin.predict(X))
         dataset.head()
```

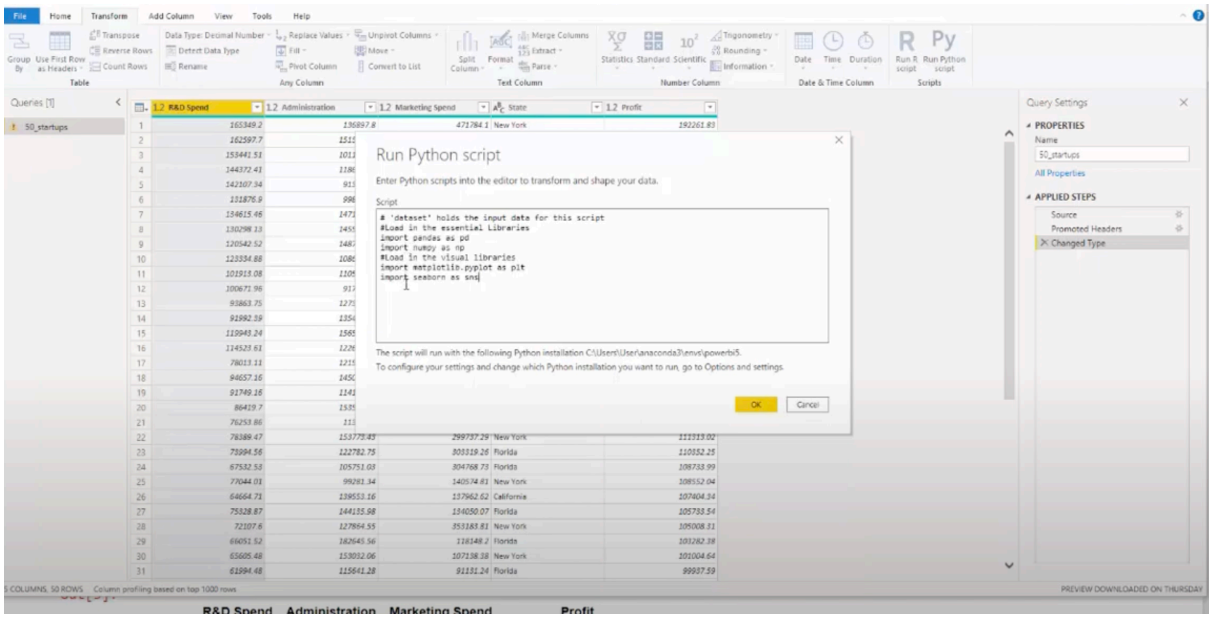
Out[20]:

	R&D Spend	Administration	Marketing Spend	Profit	Predictions	State_Florida	State_New York
0	165349.20	136897.80	471784.10	192261.83	192274.039292	0	1
1	162597.70	151377.59	443898.53	191792.06	188185.066166	0	0
2	153441.51	101145.55	407934.54	191050.39	180338.371448	1	0
3	144372.41	118671.85	383199.62	182901.99	173097.905882	0	1
4	142107.34	91391.77	366168.42	166187.94	170196.772368	1	0

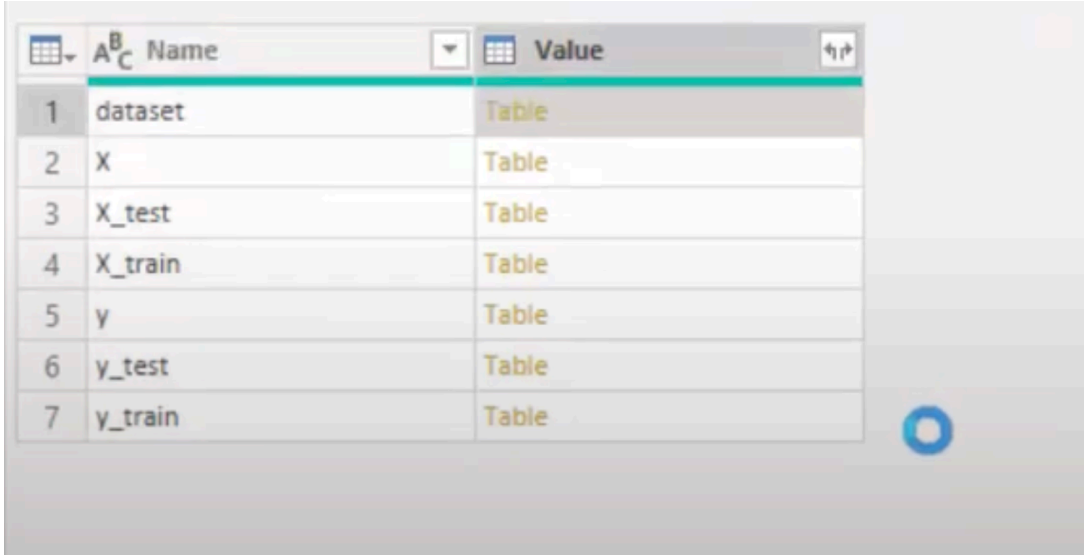
```
In [21]: #save as csv
         dataset.to_csv('full_dataset.csv')
```

```
In [ ]:
```

Loading the dataset into PowerBI and transforming



Tables are created



	R&D Spend	Administration	Marketing Spend	Profit	Predictions	State_Florida	State_New York
1	165349.2	136897.8	471784.1	192261.83	192274.0393	0	1
2	162597.7	151377.59	443898.53	191792.06	188185.0662	0	0
3	153441.51	101145.55	407934.54	191050.39	180338.3714	1	0
4	144372.41	118671.85	383199.62	182901.99	173097.9059	0	1
5	142107.34	91391.77	366168.42	166187.94	170196.7724	1	0
6	131876.9	99814.71	362861.36	156991.12	162982.9364	0	1
7	134615.46	147198.87	127716.82	156122.51	155091.0908	0	0
8	130298.13	145530.06	323876.68	155752.6	158617.0179	1	0
9	120542.52	148718.95	311613.29	152211.77	151532.5086	0	1
10	123334.88	108679.17	304981.62	149759.96	153409.1229	0	0
11	101913.08	110594.11	229160.95	146121.95	133749.9195	1	0
12	100671.96	91790.61	249744.55	144259.4	134113.4135	0	0
13	93863.75	127320.38	249839.44	141585.52	127984.52	1	0
14	91992.39	135495.07	252664.93	134307.35	126771.5642	0	0
15	119943.24	156547.42	256512.92	132602.65	147964.4971	1	0
16	114523.61	122616.84	261776.23	129917.04	145487.7744	0	1
17	78013.11	121597.55	264346.06	126992.93	116586.1569	0	0
18	94657.16	145077.58	282574.31	125370.37	130470.9661	0	1
19	91749.16	114175.79	294919.57	124266.9	128196.5367	1	0
20	86419.7	153514.11	0	122776.86	113696.0606	0	1
21	76253.86	113867.3	298664.47	118474.03	116594.2464	0	0
22	78389.47	153773.43	299737.29	111313.02	118338.6893	0	1
23	73994.56	122782.75	303319.26	110352.25	114595.0943	1	0
24	67532.53	105751.03	304768.73	108733.99	109917.9803	1	0
25	77044.01	99281.34	140574.81	108552.04	112418.4506	0	1

Visualizing the actual profit and the predicted profit