Nompumelelo Ngwenya

Training a machine learning model using the Employee Attrition data.

Introduction and Overview:

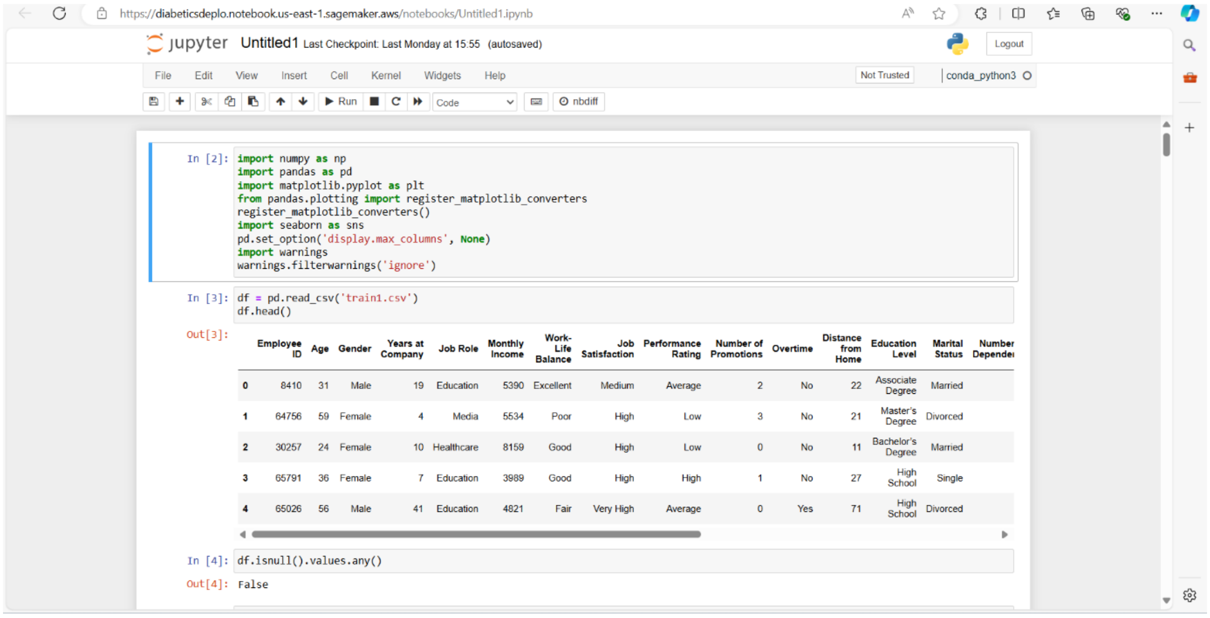
This document aims to guide stakeholders through the process of deploying a machine learning model trained on the Employees Attrition (Train) dataset on AWS.

The dataset comprises 74,498 samples and It contains detailed information about various aspects of an employee's profile, including demographics, job-related features, and personal circumstances. Each record includes a unique Employee ID and features that influence employee attrition. The aim is to understand the factors contributing to attrition and develop predictive models to identify at-risk employees.

This dataset is ideal for HR analytics, machine learning model development, and demonstrating advanced data analysis techniques. It provides a comprehensive and realistic view of the factors affecting employee retention, making it a valuable resource for researchers and practitioners in the field of human resources and organizational development.

Data source: (Employee Attrition Classification Dataset (kaggle.com))

Importing the necessary python libraries and data extracting from the data source


https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter
Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
conda_python3
Run
Code
nbdiff

```
In [2]: import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        from pandas.plotting import register_matplotlib_converters
        register_matplotlib_converters()
        import seaborn as sns
        pd.set_option('display.max_columns', None)
        import warnings
        warnings.filterwarnings('ignore')
```

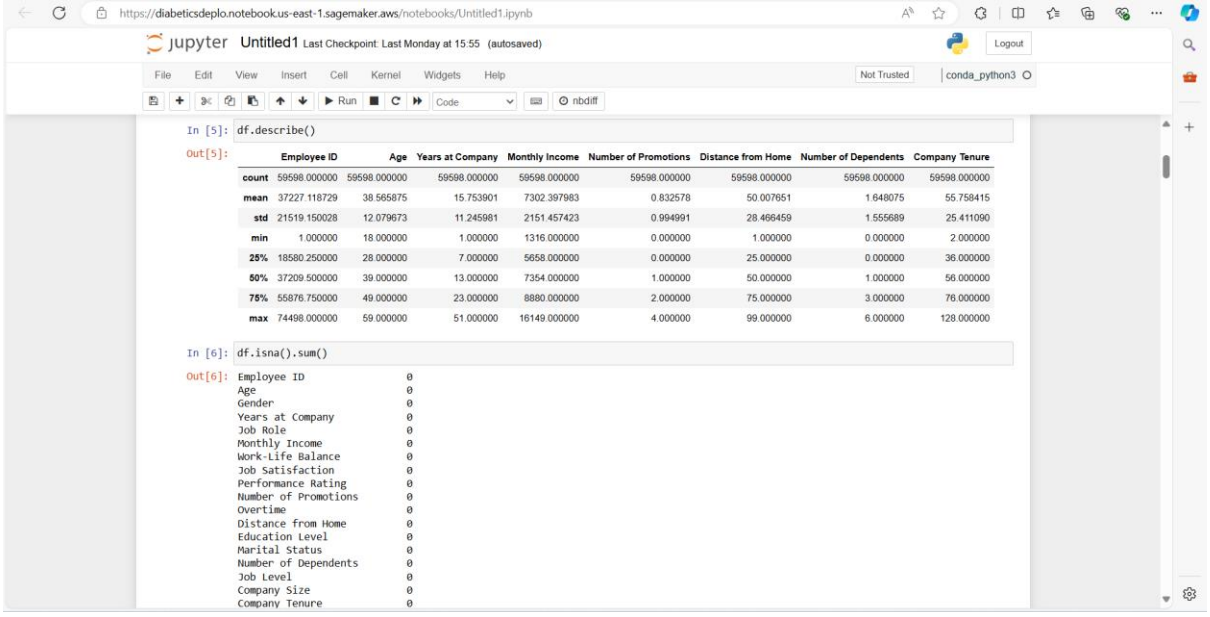
```
In [3]: df = pd.read_csv('train1.csv')
        df.head()
```

Out[3]:

	Employee ID	Age	Gender	Years at Company	Job Role	Monthly Income	Work-Life Balance	Job Satisfaction	Performance Rating	Number of Promotions	Overtime	Distance from Home	Education Level	Marital Status	Number Depende
0	8410	31	Male	19	Education	5390	Excellent	Medium	Average	2	No	22	Associate Degree	Married	
1	64756	59	Female	4	Media	5534	Poor	High	Low	3	No	21	Master's Degree	Divorced	
2	30257	24	Female	10	Healthcare	8159	Good	High	Low	0	No	11	Bachelor's Degree	Married	
3	65791	36	Female	7	Education	3989	Good	High	High	1	No	27	High School	Single	
4	65026	56	Male	41	Education	4821	Fair	Very High	Average	0	Yes	71	High School	Divorced	

```
In [4]: df.isnull().values.any()
Out[4]: False
```

Exploratory data analysis(EDA) and data cleaning by checking for any null values

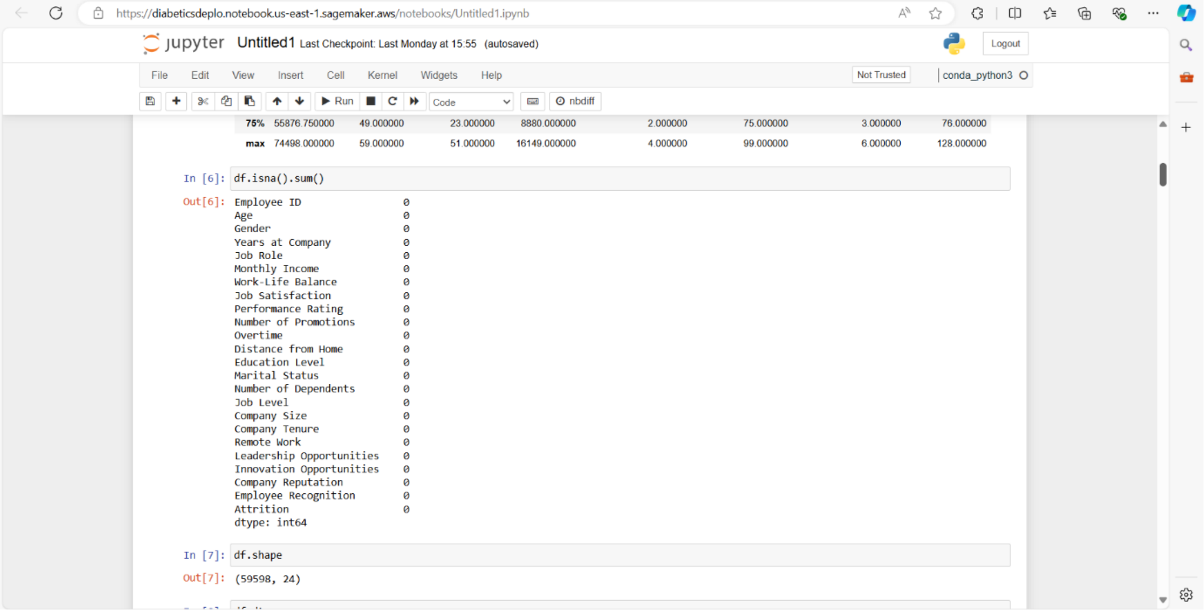

https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
conda_python3
Run
Code
nbdiff

```
In [5]: df.describe()
```

Out[5]:

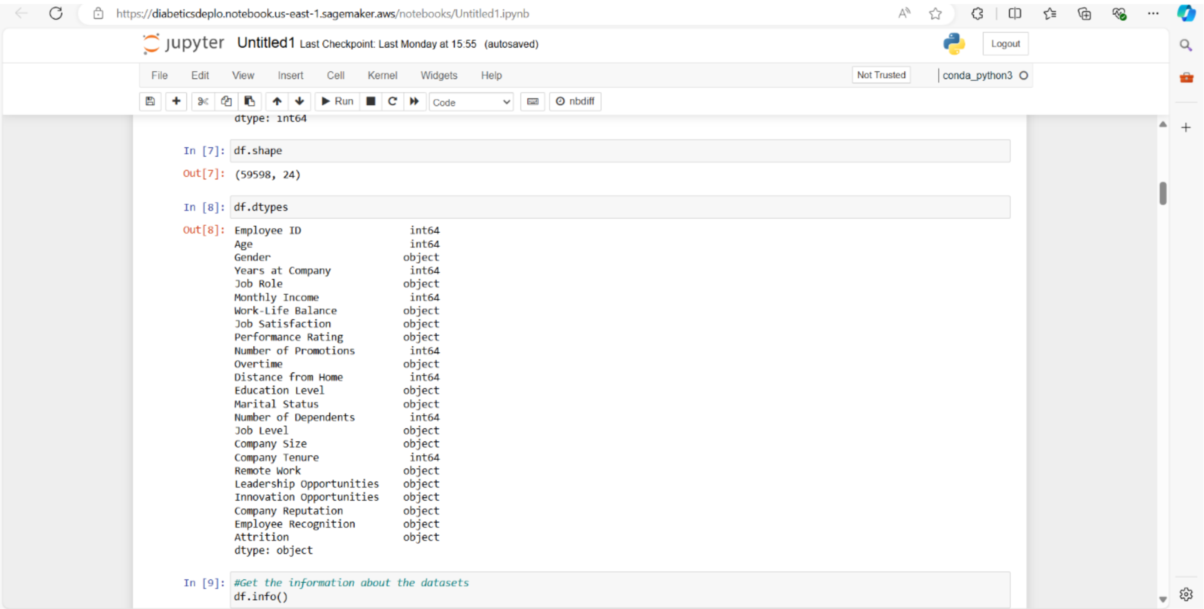
	Employee ID	Age	Years at Company	Monthly Income	Number of Promotions	Distance from Home	Number of Dependents	Company Tenure
count	59598.000000	59598.000000	59598.000000	59598.000000	59598.000000	59598.000000	59598.000000	59598.000000
mean	37227.118729	38.565875	15.753901	7302.397983	0.832578	50.007651	1.648075	55.758415
std	21519.150028	12.079673	11.245981	2151.457423	0.994991	28.466459	1.555689	25.411090
min	1.000000	18.000000	1.000000	1316.000000	0.000000	1.000000	0.000000	2.000000
25%	18580.250000	28.000000	7.000000	5658.000000	0.000000	25.000000	0.000000	36.000000
50%	37209.500000	39.000000	13.000000	7354.000000	1.000000	50.000000	1.000000	56.000000
75%	55876.750000	49.000000	23.000000	8880.000000	2.000000	75.000000	3.000000	76.000000
max	74498.000000	59.000000	51.000000	16149.000000	4.000000	99.000000	6.000000	128.000000

```
In [6]: df.isna().sum()

Out[6]: Employee ID                 0
        Age                         0
        Gender                      0
        Years at Company            0
        Job Role                    0
        Monthly Income              0
        Work-Life Balance           0
        Job Satisfaction            0
        Performance Rating          0
        Number of Promotions        0
        Overtime                    0
        Distance from Home          0
        Education Level             0
        Marital Status              0
        Number of Dependents        0
        Job Level                   0
        Company Size                0
        Company Tenure              0
```

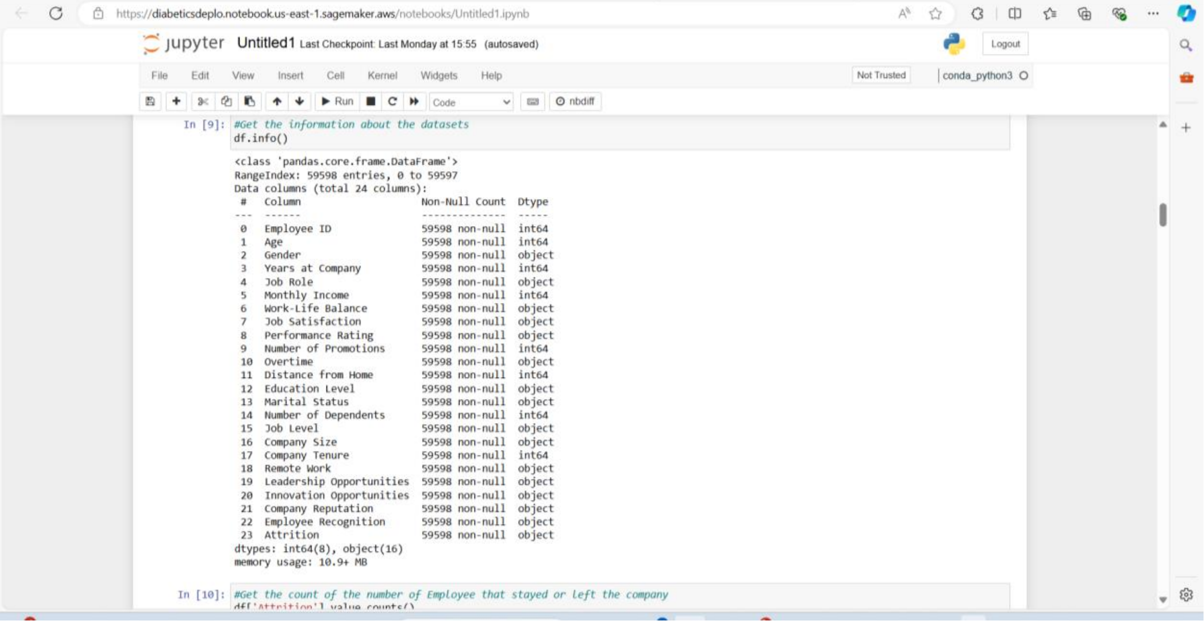

https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
conda_python3
Run
Code
nbdiff

75%	55876.750000	49.000000	23.000000	8880.000000	2.000000	75.000000	3.000000	76.000000
max	74498.000000	59.000000	51.000000	16149.000000	4.000000	99.000000	6.000000	128.000000

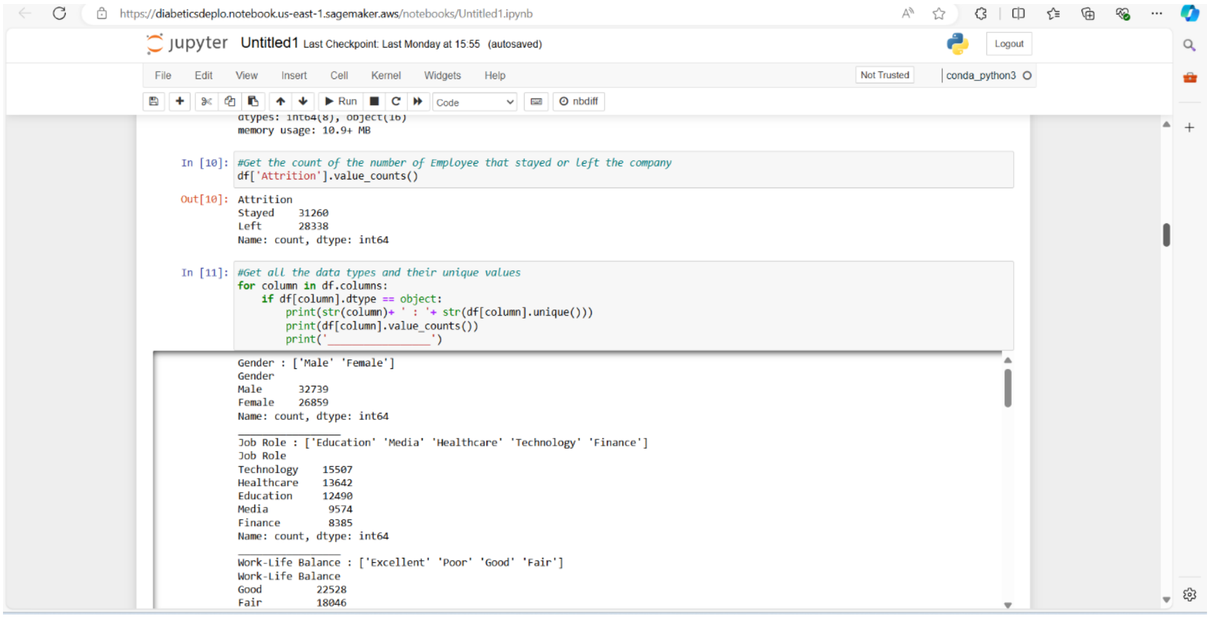
```
In [6]: df.isna().sum()

Out[6]: Employee ID                 0
        Age                         0
        Gender                      0
        Years at Company            0
        Job Role                    0
        Monthly Income              0
        Work-Life Balance           0
        Job Satisfaction            0
        Performance Rating          0
        Number of Promotions        0
        Overtime                    0
        Distance from Home          0
        Education Level             0
        Marital Status              0
        Number of Dependents        0
        Job Level                   0
        Company Size                0
        Company Tenure              0
        Remote Work                 0
        Leadership Opportunities    0
        Innovation Opportunities    0
        Company Reputation          0
        Employee Recognition        0
        Attrition                   0
        dtype: int64

In [7]: df.shape

Out[7]: (59598, 24)
```


https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
conda_python3
Run
Code
nbdiff

```
dtype: int64

In [7]: df.shape

Out[7]: (59598, 24)

In [8]: df.dtypes

Out[8]: Employee ID                    int64
        Age                            int64
        Gender                        object
        Years at Company               int64
        Job Role                      object
        Monthly Income                 int64
        Work-Life Balance             object
        Job Satisfaction              object
        Performance Rating            object
        Number of Promotions           int64
        Overtime                      object
        Distance from Home             int64
        Education Level               object
        Marital Status                object
        Number of Dependents           int64
        Job Level                     object
        Company Size                  object
        Company Tenure                 int64
        Remote Work                   object
        Leadership Opportunities      object
        Innovation Opportunities      object
        Company Reputation            object
        Employee Recognition          object
        Attrition                     object
        dtype: object

In [9]: #Get the information about the datasets
        df.info()
```

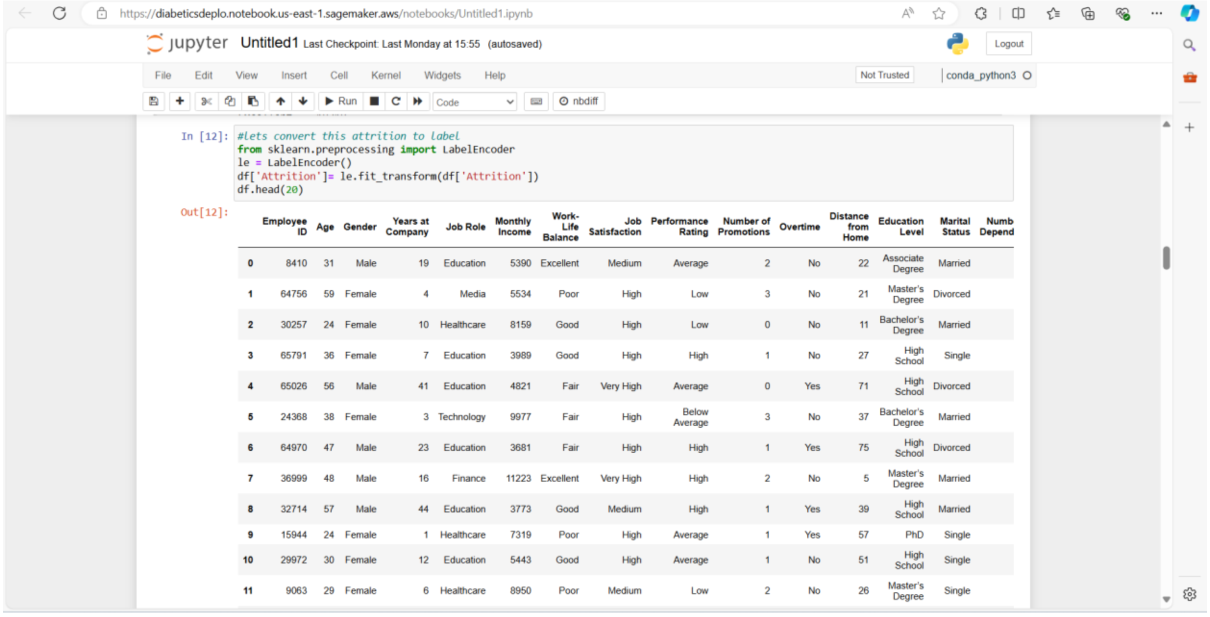

https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
conda_python3
Run
Code
nbdiff

```
In [9]: #Get the information about the datasets
        df.info()

        <class 'pandas.core.frame.DataFrame'>
        RangeIndex: 59598 entries, 0 to 59597
        Data columns (total 24 columns):
         #   Column                    Non-Null Count  Dtype
        ---  ------                    --------------  -----
         0   Employee ID               59598 non-null  int64
         1   Age                       59598 non-null  int64
         2   Gender                    59598 non-null  object
         3   Years at Company          59598 non-null  int64
         4   Job Role                  59598 non-null  object
         5   Monthly Income            59598 non-null  int64
         6   Work-Life Balance         59598 non-null  object
         7   Job Satisfaction          59598 non-null  object
         8   Performance Rating        59598 non-null  object
         9   Number of Promotions      59598 non-null  int64
         10  Overtime                  59598 non-null  object
         11  Distance from Home        59598 non-null  int64
         12  Education Level           59598 non-null  object
         13  Marital Status            59598 non-null  object
         14  Number of Dependents      59598 non-null  int64
         15  Job Level                 59598 non-null  object
         16  Company Size              59598 non-null  object
         17  Company Tenure            59598 non-null  int64
         18  Remote Work               59598 non-null  object
         19  Leadership Opportunities  59598 non-null  object
         20  Innovation Opportunities  59598 non-null  object
         21  Company Reputation        59598 non-null  object
         22  Employee Recognition      59598 non-null  object
         23  Attrition                 59598 non-null  object
        dtypes: int64(8), object(16)
        memory usage: 10.9+ MB

In [10]: #Get the count of the number of Employee that stayed or left the company
         df['Attrition'].value_counts()
```


https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
conda_python3
Run
Code
nbdiff

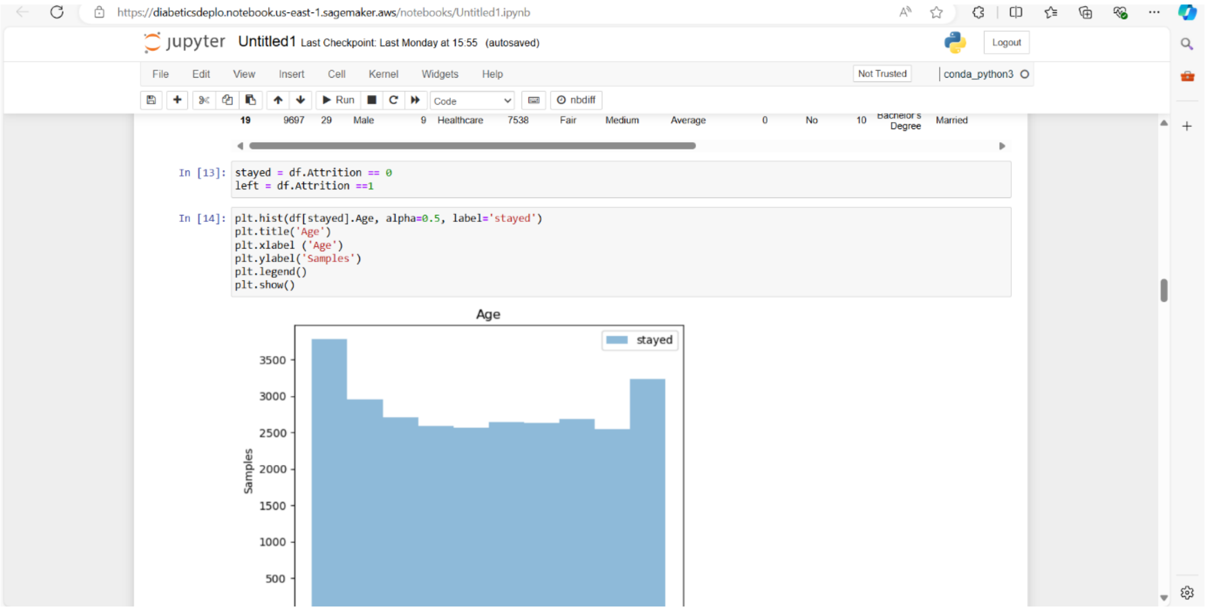
```
dtypes: int64(8), object(16)
memory usage: 10.9+ MB

In [10]: #Get the count of the number of Employee that stayed or left the company
         df['Attrition'].value_counts()

Out[10]: Attrition
         Stayed    31260
         Left      28338
         Name: count, dtype: int64

In [11]: #Get all the data types and their unique values
         for column in df.columns:
             if df[column].dtype == object:
                 print(str(column)+ ' : '+ str(df[column].unique()))
                 print(df[column].value_counts())
                 print('_________________')

         Gender : ['Male' 'Female']
         Gender
         Male      32739
         Female    26859
         Name: count, dtype: int64
         _________________
         Job Role : ['Education' 'Media' 'Healthcare' 'Technology' 'Finance']
         Job Role
         Technology    15507
         Healthcare    13642
         Education     12490
         Media          9574
         Finance        8385
         Name: count, dtype: int64
         _________________
         Work-Life Balance : ['Excellent' 'Poor' 'Good' 'Fair']
         Work-Life Balance
         Good         22528
         Fair         18046
```

Encoding the attrition. Employees that stayed as 1, employees who left as 0


https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
conda_python3
Run
Code
nbdiff

```
In [12]: #lets convert this attrition to label
         from sklearn.preprocessing import LabelEncoder
         le = LabelEncoder()
         df['Attrition']= le.fit_transform(df['Attrition'])
         df.head(20)

Out[12]:
```

	Employee ID	Age	Gender	Years at Company	Job Role	Monthly Income	Work-Life Balance	Job Satisfaction	Performance Rating	Number of Promotions	Overtime	Distance from Home	Education Level	Marital Status	Numb Depend
0	8410	31	Male	19	Education	5390	Excellent	Medium	Average	2	No	22	Associate Degree	Married	
1	64756	59	Female	4	Media	5534	Poor	High	Low	3	No	21	Master's Degree	Divorced	
2	30257	24	Female	10	Healthcare	8159	Good	High	Low	0	No	11	Bachelor's Degree	Married	
3	65791	36	Female	7	Education	3989	Good	High	High	1	No	27	High School	Single	
4	65026	56	Male	41	Education	4821	Fair	Very High	Average	0	Yes	71	High School	Divorced	
5	24368	38	Female	3	Technology	9977	Fair	High	Below Average	3	No	37	Bachelor's Degree	Married	
6	64970	47	Male	23	Education	3681	Fair	High	High	1	Yes	75	High School	Divorced	
7	36999	48	Male	16	Finance	11223	Excellent	Very High	High	2	No	5	Master's Degree	Married	
8	32714	57	Male	44	Education	3773	Good	Medium	High	1	Yes	39	High School	Married	
9	15944	24	Female	1	Healthcare	7319	Poor	High	Average	1	Yes	57	PhD	Single	
10	29972	30	Female	12	Education	5443	Good	High	Average	1	No	51	High School	Single	
11	9063	29	Female	6	Healthcare	8950	Poor	Medium	Low	2	No	26	Master's Degree	Single	



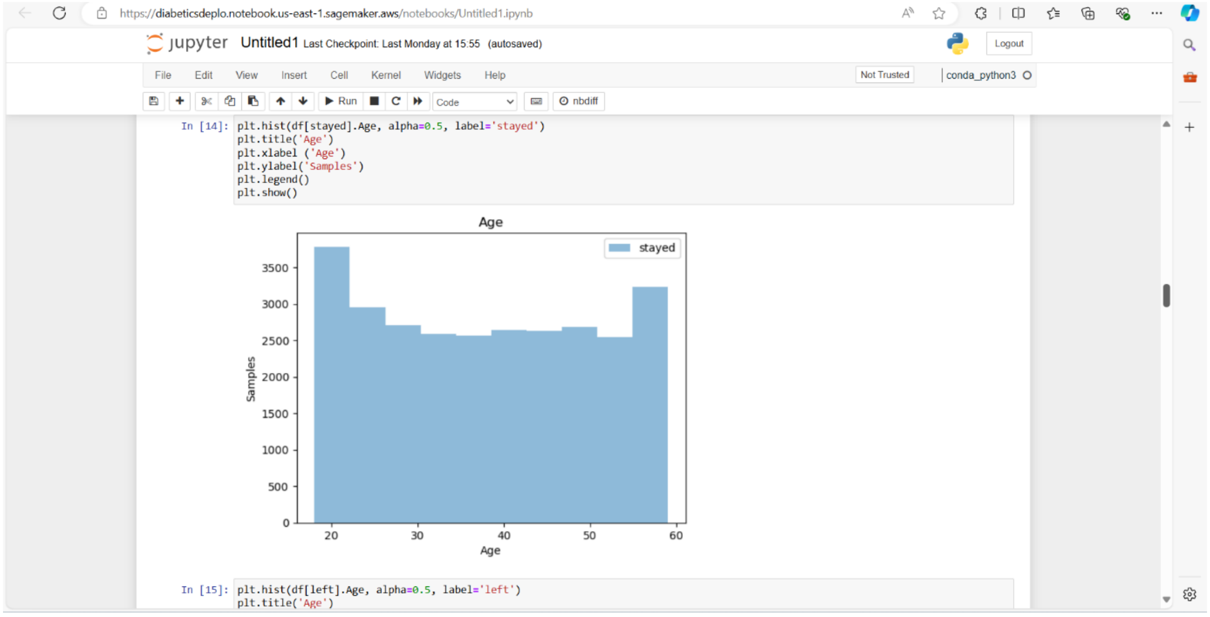
https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter
Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
conda_python3
Run
Code
nbdiff

```
19    9697    29    Male    9    Healthcare    7538    Fair    Medium    Average    0    No    10    Bachelor's Degree    Married
```

In [13]:
```
stayed = df.Attrition == 0
left = df.Attrition ==1
```

In [14]:
```
plt.hist(df[stayed].Age, alpha=0.5, label='stayed')
plt.title('Age')
plt.xlabel ('Age')
plt.ylabel('Samples')
plt.legend()
plt.show()
```

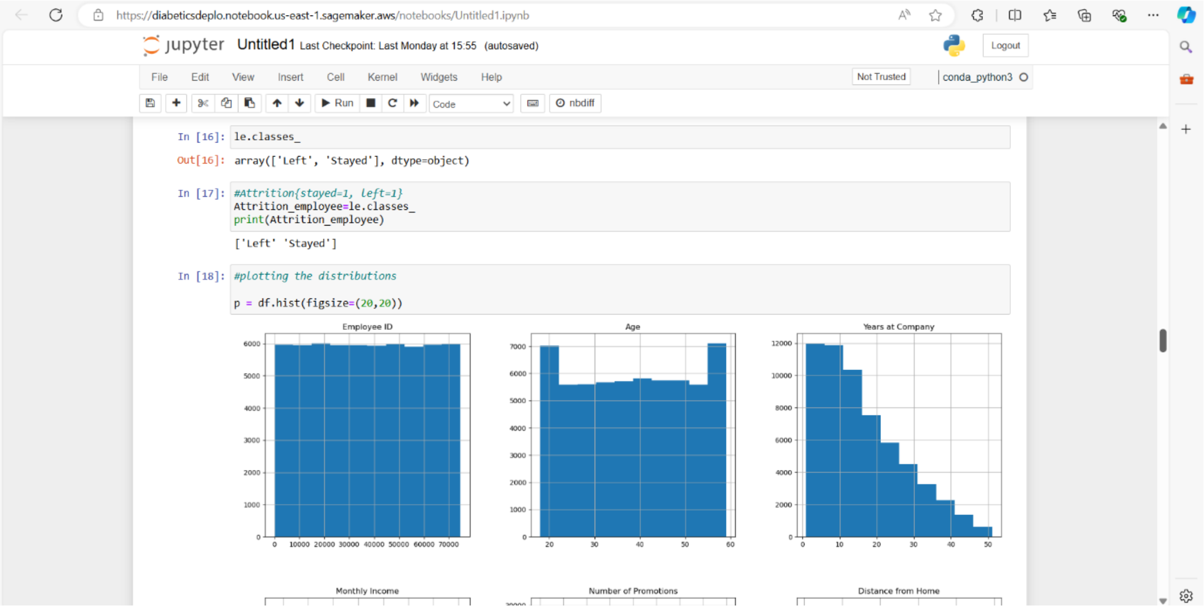
Age
stayed
Samples
3500
3000
2500
2000
1500
1000
500



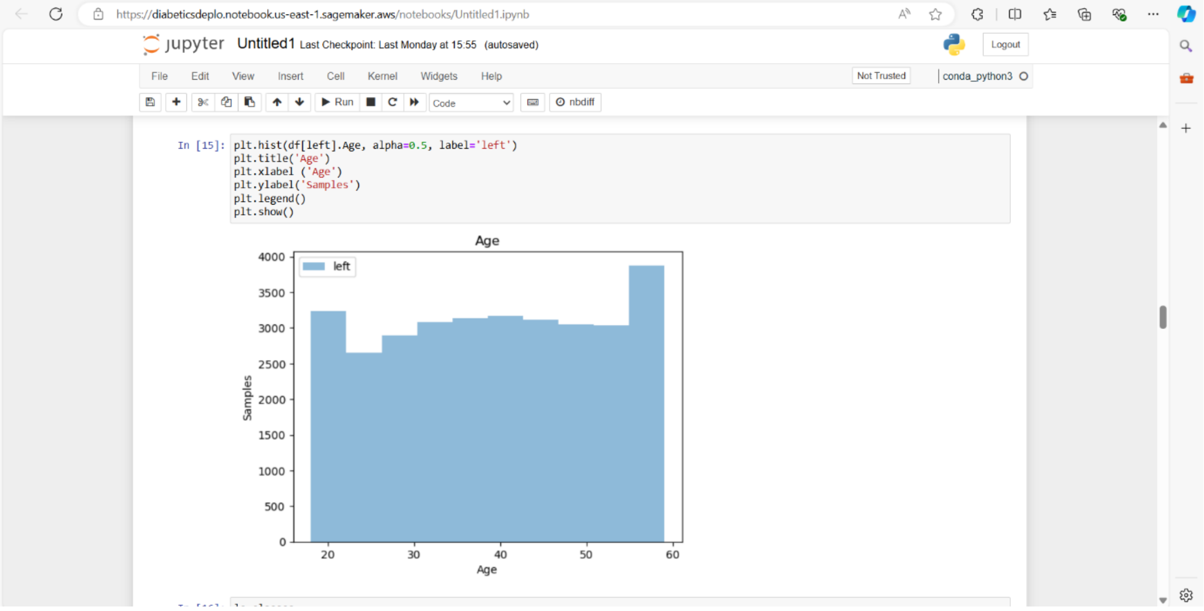
https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter
Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
conda_python3
Run
Code
nbdiff

In [14]:
```
plt.hist(df[stayed].Age, alpha=0.5, label='stayed')
plt.title('Age')
plt.xlabel ('Age')
plt.ylabel('Samples')
plt.legend()
plt.show()
```

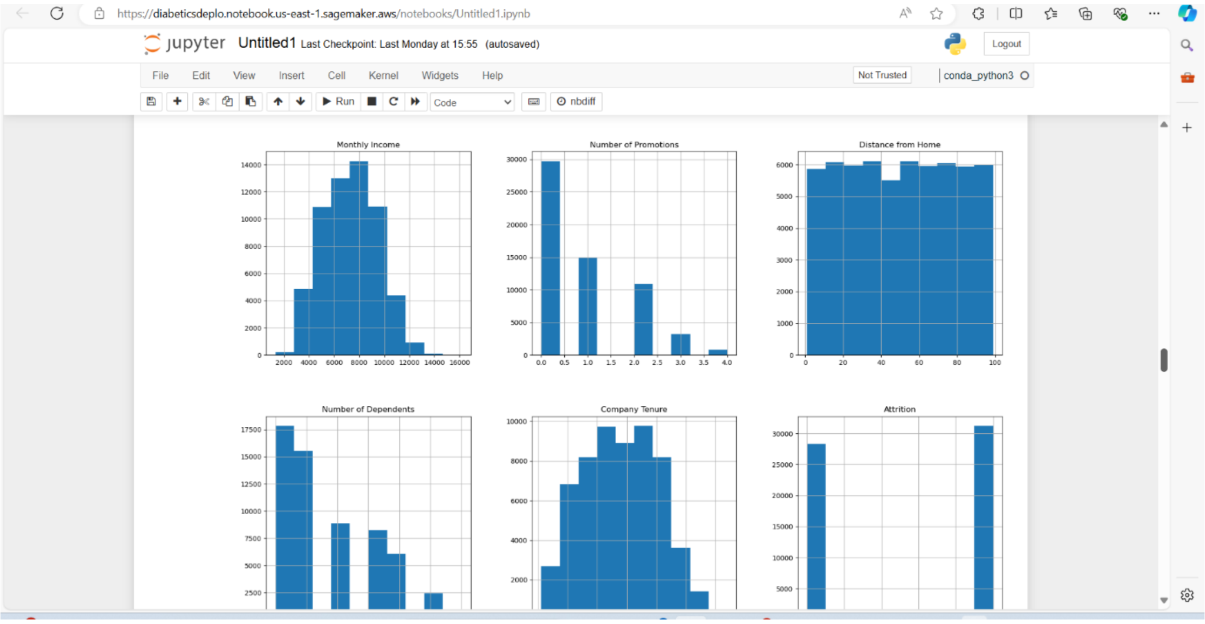
In [15]:
```
plt.hist(df[left].Age, alpha=0.5, label='left')
plt.title('Age')
```

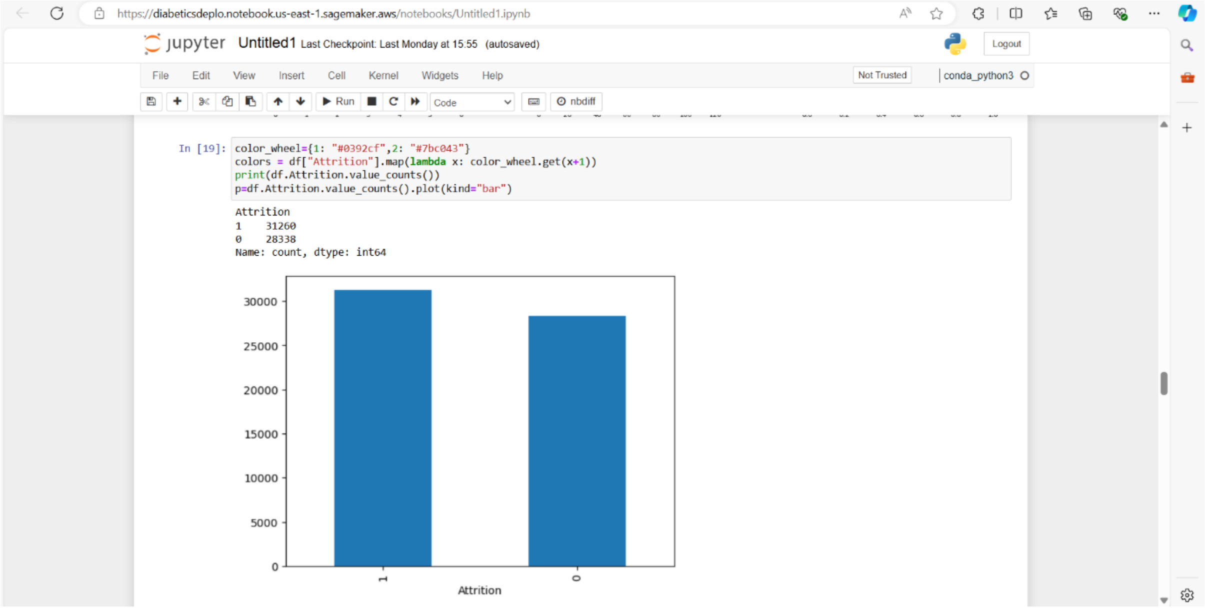


```
In [16]: le.classes_

Out[16]: array(['Left', 'Stayed'], dtype=object)

In [17]: #Attrition{stayed=1, left=1}
         Attrition_employee=le.classes_
         print(Attrition_employee)

         ['Left' 'Stayed']

In [18]: #plotting the distributions

         p = df.hist(figsize=(20,20))
```



```
In [15]: plt.hist(df[left].Age, alpha=0.5, label='left')
         plt.title('Age')
         plt.xlabel ('Age')
         plt.ylabel('Samples')
         plt.legend()
         plt.show()
```



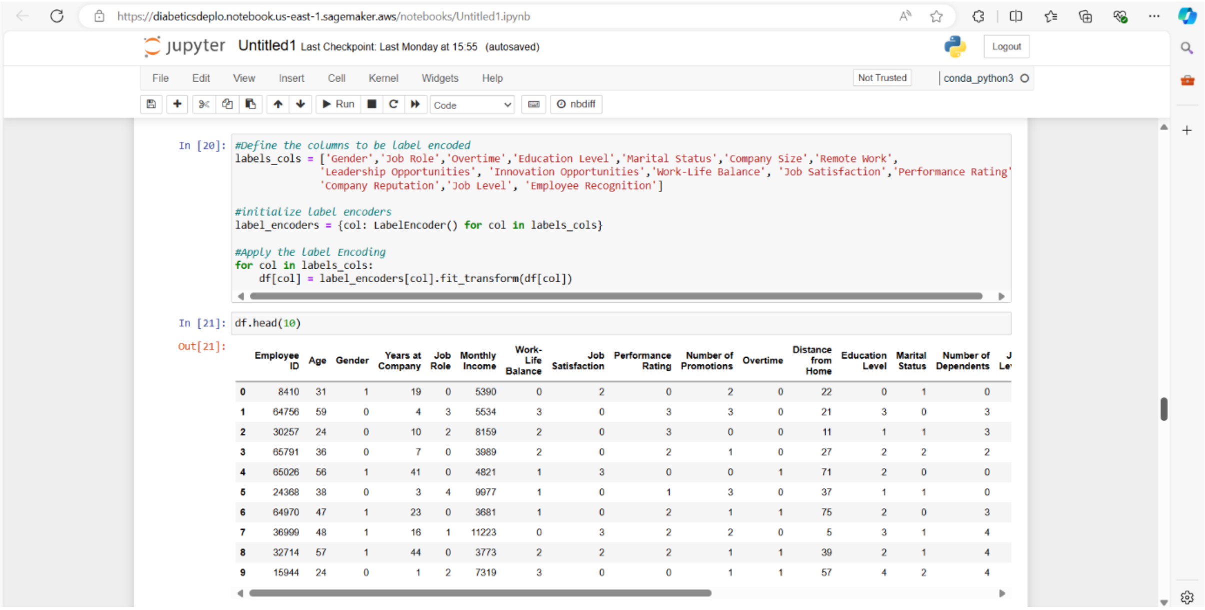

https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter
Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
conda_python3
Run
Code
nbdiff
In [19]:

```
color_wheel={1: "#0392cf",2: "#7bc043"}
colors = df["Attrition"].map(lambda x: color_wheel.get(x+1))
print(df.Attrition.value_counts())
p=df.Attrition.value_counts().plot(kind="bar")
```

```
Attrition
1    31260
0    28338
Name: count, dtype: int64
```

30000
25000
20000
15000
10000
5000
0
1
0
Attrition

Encoding the columns to prepare for training.


https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
conda_python3
Run
Code
nbdiff

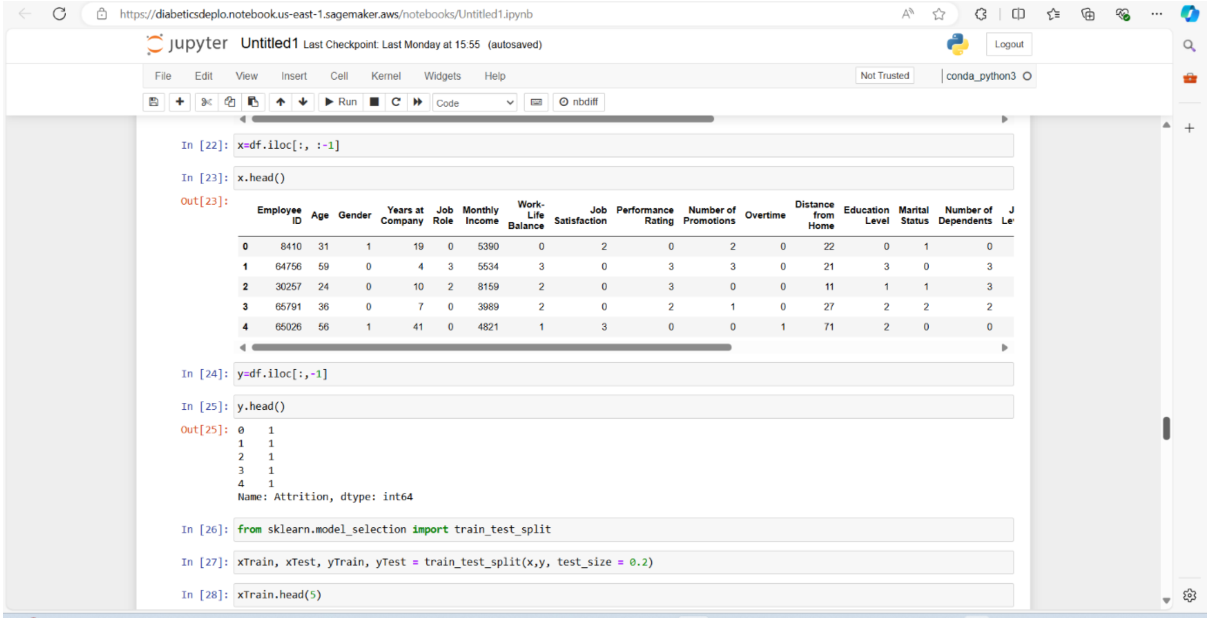
In [20]:
```
#Define the columns to be label encoded
labels_cols = ['Gender','Job Role','Overtime','Education Level','Marital Status','Company Size','Remote Work',
              'Leadership Opportunities', 'Innovation Opportunities','Work-Life Balance', 'Job Satisfaction','Performance Rating'
              'Company Reputation','Job Level', 'Employee Recognition']

#initialize label encoders
label_encoders = {col: LabelEncoder() for col in labels_cols}

#Apply the label Encoding
for col in labels_cols:
    df[col] = label_encoders[col].fit_transform(df[col])
```

In [21]:
```
df.head(10)
```

Out[21]:

	Employee ID	Age	Gender	Years at Company	Job Role	Monthly Income	Work-Life Balance	Job Satisfaction	Performance Rating	Number of Promotions	Overtime	Distance from Home	Education Level	Marital Status	Number of Dependents	J Le
0	8410	31	1	19	0	5390	0	2	0	2	0	22	0	1	0	
1	64756	59	0	4	3	5534	3	0	3	3	0	21	3	0	3	
2	30257	24	0	10	2	8159	2	0	3	0	0	11	1	1	3	
3	65791	36	0	7	0	3989	2	0	2	1	0	27	2	2	2	
4	65026	56	1	41	0	4821	1	3	0	0	1	71	2	0	0	
5	24368	38	0	3	4	9977	1	0	1	3	0	37	1	1	0	
6	64970	47	1	23	0	3681	1	0	2	1	1	75	2	0	3	
7	36999	48	1	16	1	11223	0	3	2	2	0	5	3	1	4	
8	32714	57	1	44	0	3773	2	2	2	1	1	39	2	1	4	
9	15944	24	0	1	2	7319	3	0	0	1	1	57	4	2	4	

Splitting data as X and Y, to prepare for training.


https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
conda_python3
Run
Code
nbdiff

In [22]:
```
x=df.iloc[:, :-1]
```

In [23]:
```
x.head()
```

Out[23]:

	Employee ID	Age	Gender	Years at Company	Job Role	Monthly Income	Work-Life Balance	Job Satisfaction	Performance Rating	Number of Promotions	Overtime	Distance from Home	Education Level	Marital Status	Number of Dependents	J Le
0	8410	31	1	19	0	5390	0	2	0	2	0	22	0	1	0	
1	64756	59	0	4	3	5534	3	0	3	3	0	21	3	0	3	
2	30257	24	0	10	2	8159	2	0	3	0	0	11	1	1	3	
3	65791	36	0	7	0	3989	2	0	2	1	0	27	2	2	2	
4	65026	56	1	41	0	4821	1	3	0	0	1	71	2	0	0	

In [24]:
```
y=df.iloc[:,-1]
```

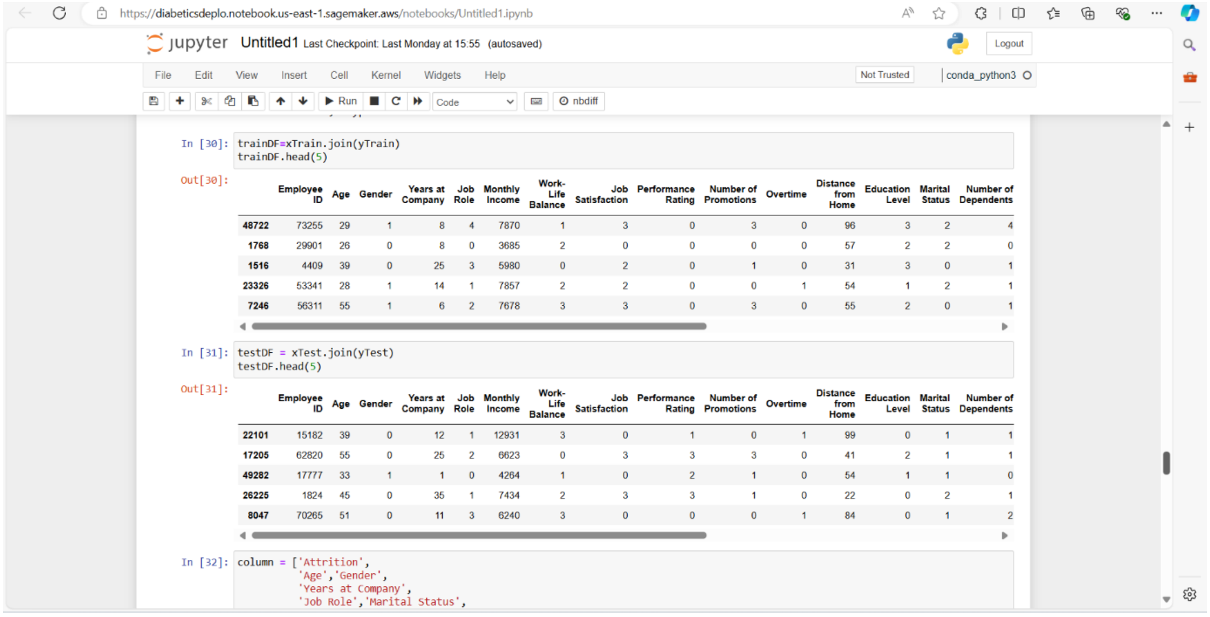
In [25]:
```
y.head()
```

Out[25]:
```
0    1
1    1
2    1
3    1
4    1
Name: Attrition, dtype: int64
```

In [26]:
```
from sklearn.model_selection import train_test_split
```

In [27]:
```
xTrain, xTest, yTrain, yTest = train_test_split(x,y, test_size = 0.2)
```

In [28]:
```
xTrain.head(5)
```



https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter
Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
conda_python3
Run
Code
nbdiff

```
In [30]: trainDF=xTrain.join(yTrain)
         trainDF.head(5)
```

Out[30]:

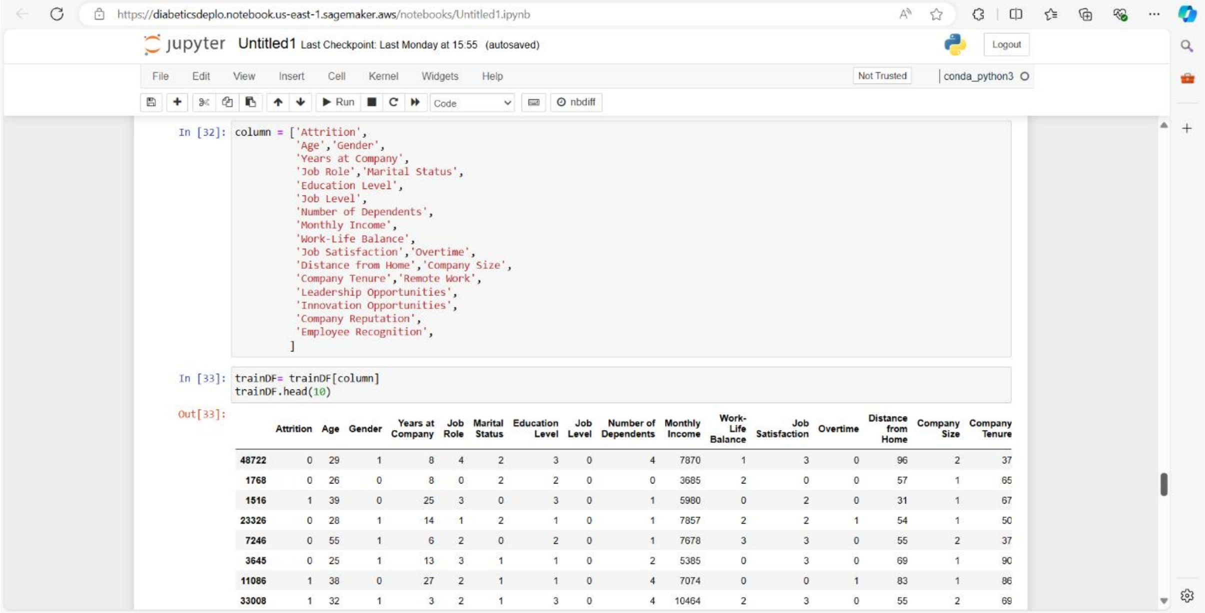
	Employee ID	Age	Gender	Years at Company	Job Role	Monthly Income	Work-Life Balance	Job Satisfaction	Performance Rating	Number of Promotions	Overtime	Distance from Home	Education Level	Marital Status	Number of Dependents
48722	73255	29	1	8	4	7870	1	3	0	3	0	96	3	2	4
1768	29901	26	0	8	0	3685	2	0	0	0	0	57	2	2	0
1516	4409	39	0	25	3	5980	0	2	0	1	0	31	3	0	1
23326	53341	28	1	14	1	7857	2	2	0	0	1	54	1	2	1
7246	56311	55	1	6	2	7678	3	3	0	3	0	55	2	0	1

```
In [31]: testDF = xTest.join(yTest)
         testDF.head(5)
```

Out[31]:

	Employee ID	Age	Gender	Years at Company	Job Role	Monthly Income	Work-Life Balance	Job Satisfaction	Performance Rating	Number of Promotions	Overtime	Distance from Home	Education Level	Marital Status	Number of Dependents
22101	15182	39	0	12	1	12931	3	0	1	0	1	99	0	1	1
17205	62820	55	0	25	2	6623	0	3	3	3	0	41	2	1	1
49282	17777	33	1	1	0	4264	1	0	2	1	0	54	1	1	0
26225	1824	45	0	35	1	7434	2	3	3	1	0	22	0	2	1
8047	70265	51	0	11	3	6240	3	0	0	0	1	84	0	1	2

```
In [32]: column = ['Attrition',
                   'Age','Gender',
                   'Years at Company',
                   'Job Role','Marital Status',
```



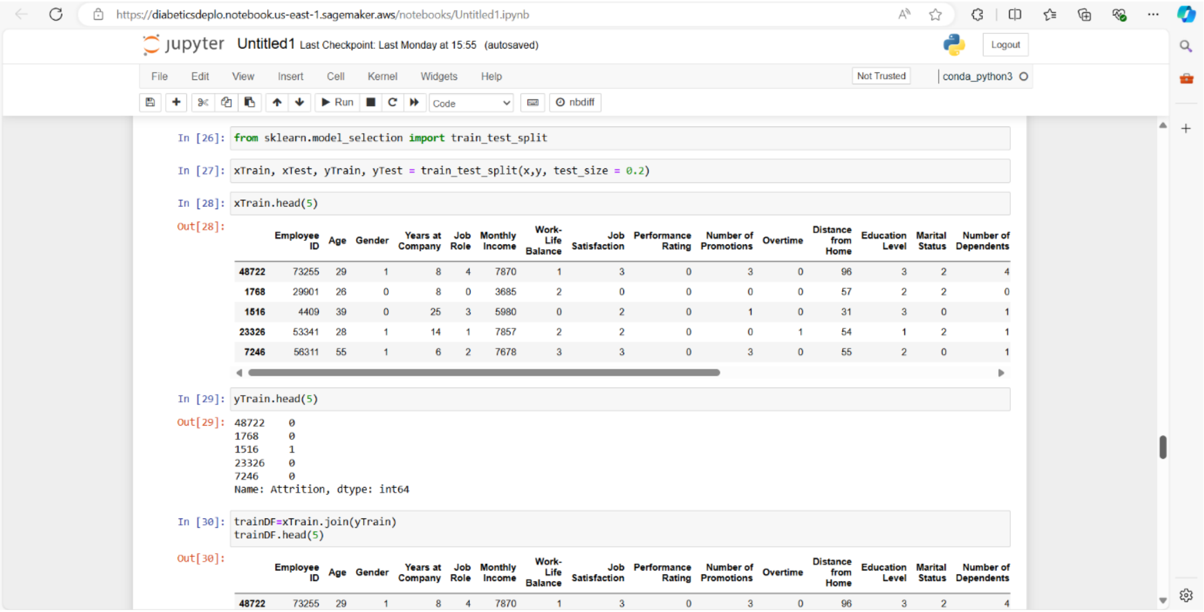
https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter
Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
conda_python3
Run
Code
nbdiff

```
In [32]: column = ['Attrition',
                   'Age','Gender',
                   'Years at Company',
                   'Job Role','Marital Status',
                   'Education Level',
                   'Job Level',
                   'Number of Dependents',
                   'Monthly Income',
                   'Work-Life Balance',
                   'Job Satisfaction','Overtime',
                   'Distance from Home','Company Size',
                   'Company Tenure','Remote Work',
                   'Leadership Opportunities',
                   'Innovation Opportunities',
                   'Company Reputation',
                   'Employee Recognition',
                  ]
```

```
In [33]: trainDF= trainDF[column]
         trainDF.head(10)
```

Out[33]:

	Attrition	Age	Gender	Years at Company	Job Role	Marital Status	Education Level	Job Level	Number of Dependents	Monthly Income	Work-Life Balance	Job Satisfaction	Overtime	Distance from Home	Company Size	Company Tenure
48722	0	29	1	8	4	2	3	0	4	7870	1	3	0	96	2	37
1768	0	26	0	8	0	2	2	0	0	3685	2	0	0	57	1	65
1516	1	39	0	25	3	0	3	0	1	5980	0	2	0	31	1	67
23326	0	28	1	14	1	2	1	0	1	7857	2	2	1	54	1	50
7246	0	55	1	6	2	0	2	0	1	7678	3	3	0	55	2	37
3645	0	25	1	13	3	1	1	0	2	5385	0	3	0	69	1	90
11086	1	38	0	27	2	1	1	0	4	7074	0	0	1	83	1	86
33008	1	32	1	3	2	1	3	0	4	10464	2	3	0	55	2	69


https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
conda_python3
Run
Code
nbdiff

```
In [26]: from sklearn.model_selection import train_test_split
```

```
In [27]: xTrain, xTest, yTrain, yTest = train_test_split(x,y, test_size = 0.2)
```

```
In [28]: xTrain.head(5)
```

Out[28]:

	Employee ID	Age	Gender	Years at Company	Job Role	Monthly Income	Work-Life Balance	Job Satisfaction	Performance Rating	Number of Promotions	Overtime	Distance from Home	Education Level	Marital Status	Number of Dependents
48722	73255	29	1	8	4	7870	1	3	0	3	0	96	3	2	4
1768	29901	26	0	8	0	3685	2	0	0	0	0	57	2	2	0
1516	4409	39	0	25	3	5980	0	2	0	1	0	31	3	0	1
23326	53341	28	1	14	1	7857	2	2	0	0	1	54	1	2	1
7246	56311	55	1	6	2	7678	3	3	0	3	0	55	2	0	1

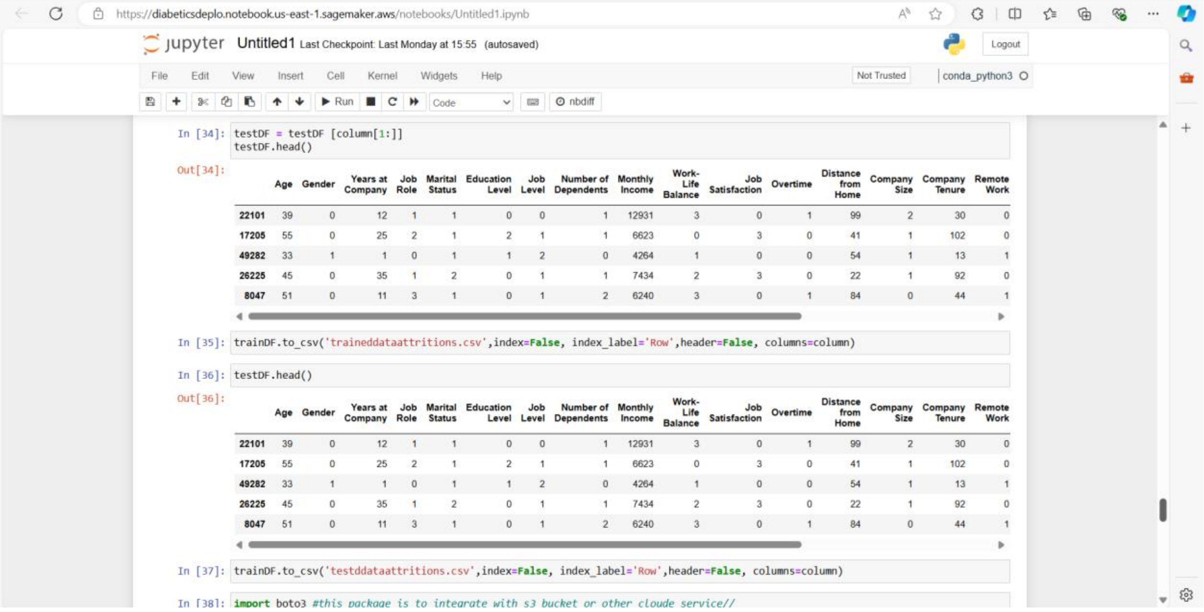
```
In [29]: yTrain.head(5)
```

```
Out[29]: 48722    0
         1768     0
         1516     1
         23326    0
         7246     0
         Name: Attrition, dtype: int64
```

```
In [30]: trainDF=xTrain.join(yTrain)
         trainDF.head(5)
```

Out[30]:

	Employee ID	Age	Gender	Years at Company	Job Role	Monthly Income	Work-Life Balance	Job Satisfaction	Performance Rating	Number of Promotions	Overtime	Distance from Home	Education Level	Marital Status	Number of Dependents
48722	73255	29	1	8	4	7870	1	3	0	3	0	96	3	2	4


https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
conda_python3
Run
Code
nbdiff

```
In [34]: testDF = testDF [column[1:]]
         testDF.head()
```

Out[34]:

	Age	Gender	Years at Company	Job Role	Marital Status	Education Level	Job Level	Number of Dependents	Monthly Income	Work-Life Balance	Job Satisfaction	Overtime	Distance from Home	Company Size	Company Tenure	Remote Work
22101	39	0	12	1	1	0	0	1	12931	3	0	1	99	2	30	0
17205	55	0	25	2	1	2	1	1	6623	0	3	0	41	1	102	0
49282	33	1	1	0	1	1	2	0	4264	1	0	0	54	1	13	1
26225	45	0	35	1	2	0	1	1	7434	2	3	0	22	1	92	0
8047	51	0	11	3	1	0	1	2	6240	3	0	1	84	0	44	1

```
In [35]: trainDF.to_csv('traineddataattritions.csv',index=False, index_label='Row',header=False, columns=column)
```

```
In [36]: testDF.head()
```

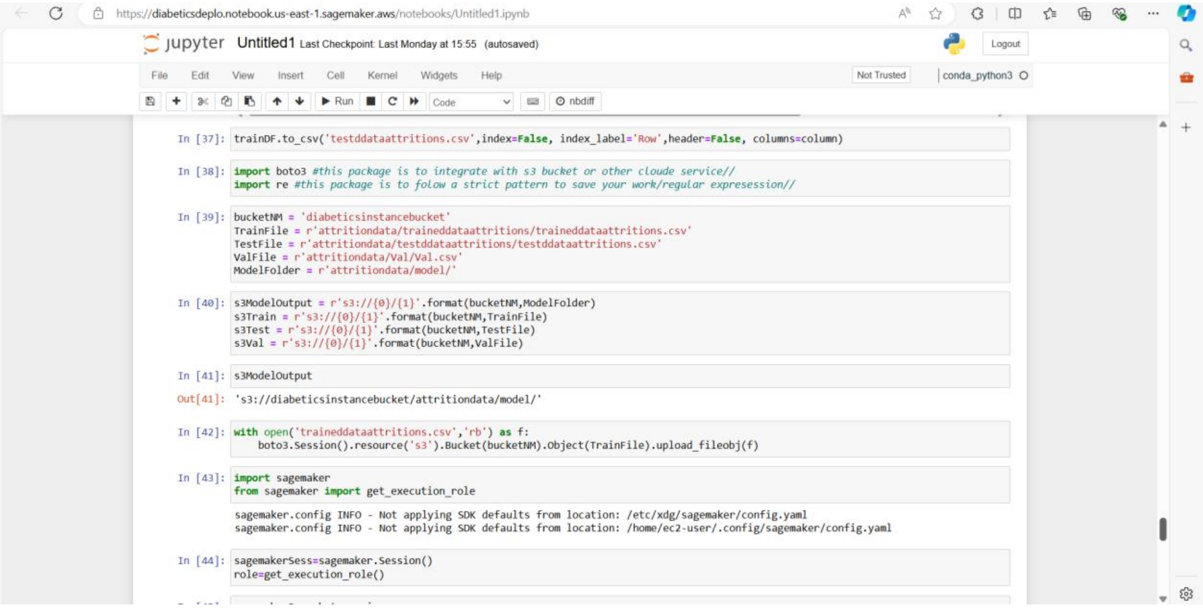
Out[36]:

	Age	Gender	Years at Company	Job Role	Marital Status	Education Level	Job Level	Number of Dependents	Monthly Income	Work-Life Balance	Job Satisfaction	Overtime	Distance from Home	Company Size	Company Tenure	Remote Work
22101	39	0	12	1	1	0	0	1	12931	3	0	1	99	2	30	0
17205	55	0	25	2	1	2	1	1	6623	0	3	0	41	1	102	0
49282	33	1	1	0	1	1	2	0	4264	1	0	0	54	1	13	1
26225	45	0	35	1	2	0	1	1	7434	2	3	0	22	1	92	0
8047	51	0	11	3	1	0	1	2	6240	3	0	1	84	0	44	1

```
In [37]: trainDF.to_csv('testddataattritions.csv',index=False, index_label='Row',header=False, columns=column)
```

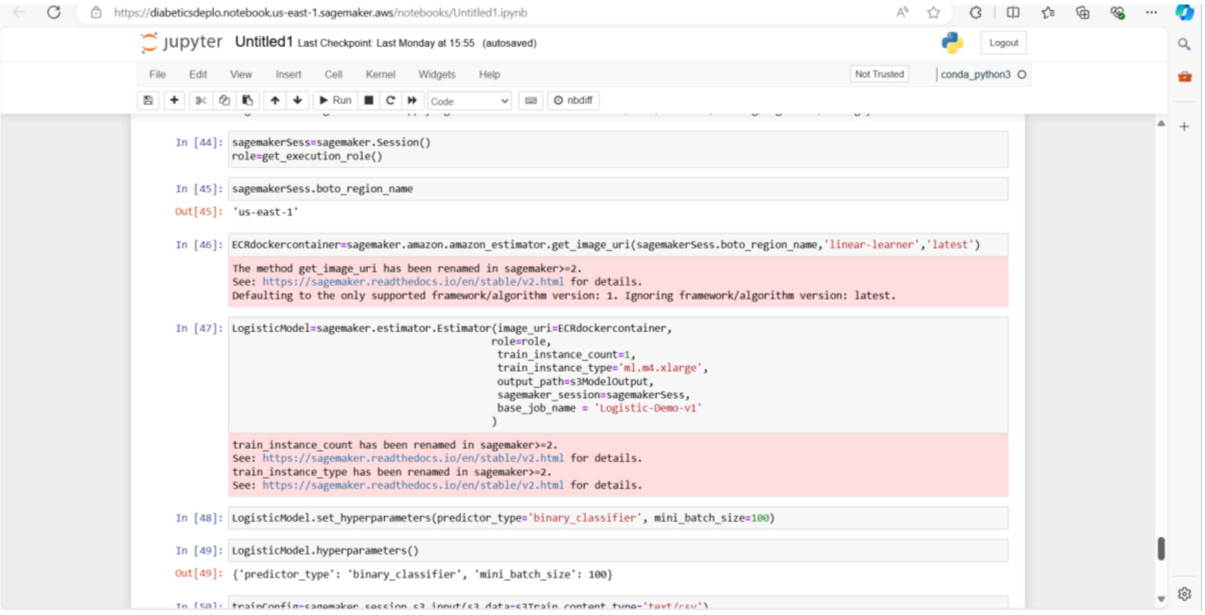
```
In [38]: import boto3 #this package is to integrate with s3 bucket or other cloude service//
```

Creating an s3 storage and moving the trained and test data to the s3 object storage


https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
conda_python3
Run
Code
nbdiff

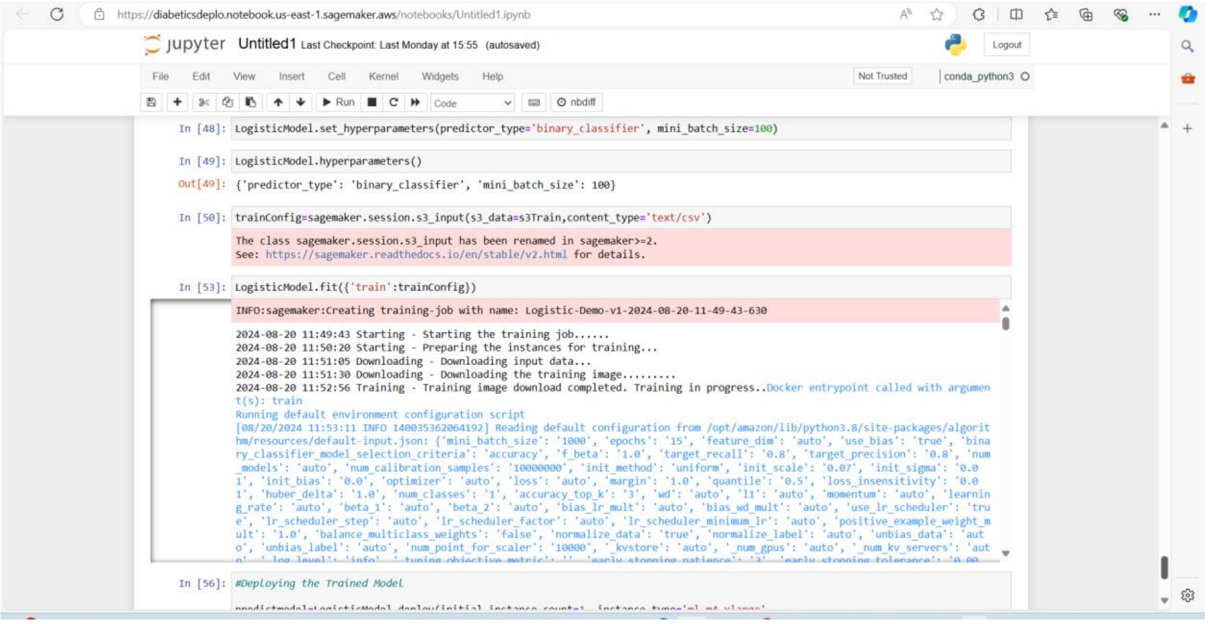
```
In [37]: trainDF.to_csv('testddataattritions.csv',index=False, index_label='Row',header=False, columns=column)

In [38]: import boto3 #this package is to integrate with s3 bucket or other cloude service//
         import re #this package is to folow a strict pattern to save your work/regular expresession//

In [39]: bucketNM = 'diabeticsinstancebucket'
         TrainFile = r'attritiondata/traineddataattritions/traineddataattritions.csv'
         TestFile = r'attritiondata/testddataattritions/testddataattritions.csv'
         ValFile = r'attritiondata/Val/Val.csv'
         ModelFolder = r'attritiondata/model/'

In [40]: s3ModelOutput = r's3://{0}/{1}'.format(bucketNM,ModelFolder)
         s3Train = r's3://{0}/{1}'.format(bucketNM,TrainFile)
         s3Test = r's3://{0}/{1}'.format(bucketNM,TestFile)
         s3Val = r's3://{0}/{1}'.format(bucketNM,ValFile)

In [41]: s3ModelOutput

Out[41]: 's3://diabeticsinstancebucket/attritiondata/model/'

In [42]: with open('traineddataattritions.csv','rb') as f:
             boto3.Session().resource('s3').Bucket(bucketNM).Object(TrainFile).upload_fileobj(f)

In [43]: import sagemaker
         from sagemaker import get_execution_role

         sagemaker.config INFO - Not applying SDK defaults from location: /etc/xdg/sagemaker/config.yaml
         sagemaker.config INFO - Not applying SDK defaults from location: /home/ec2-user/.config/sagemaker/config.yaml

In [44]: sagemakerSess=sagemaker.Session()
         role=get_execution_role()
```

Setting up the hyperparameter for tuning/optimization


https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
conda_python3
Run
Code
nbdiff

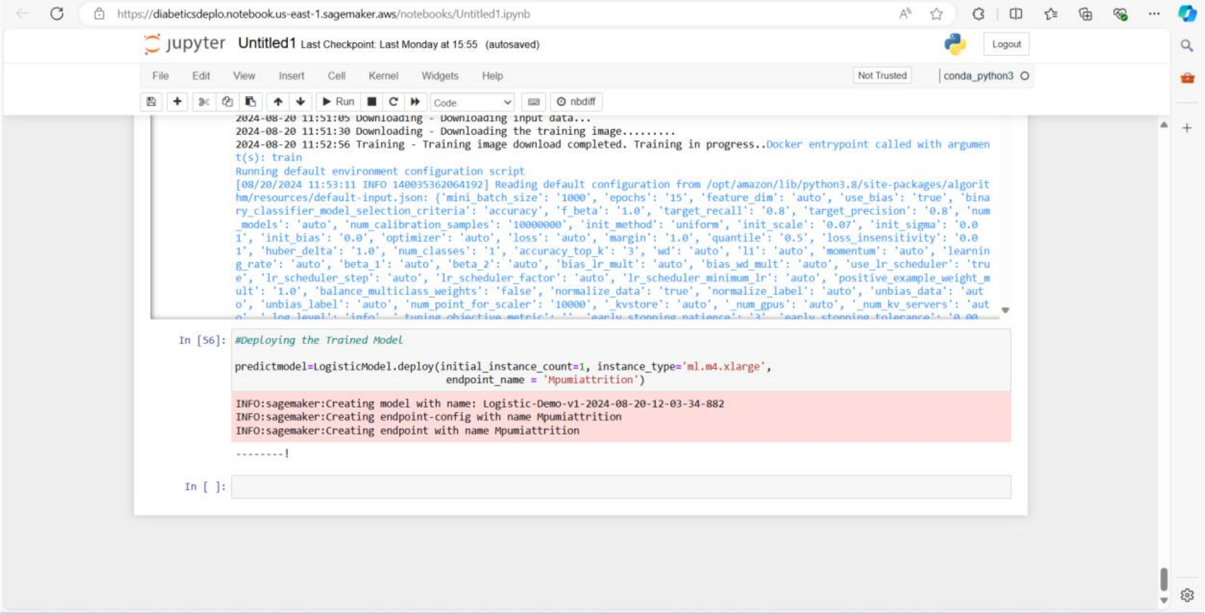
```
In [44]: sagemakerSess=sagemaker.Session()
         role=get_execution_role()

In [45]: sagemakerSess.boto_region_name

Out[45]: 'us-east-1'

In [46]: ECRdockercontainer=sagemaker.amazon.amazon_estimator.get_image_uri(sagemakerSess.boto_region_name,'linear-learner','latest')

         The method get_image_uri has been renamed in sagemaker>=2.
         See: https://sagemaker.readthedocs.io/en/stable/v2.html for details.
         Defaulting to the only supported framework/algorithm version: 1. Ignoring framework/algorithm version: latest.

In [47]: LogisticModel=sagemaker.estimator.Estimator(image_uri=ECRdockercontainer,
                                                    role=role,
                                                     train_instance_count=1,
                                                     train_instance_type='ml.m4.xlarge',
                                                     output_path=s3ModelOutput,
                                                     sagemaker_session=sagemakerSess,
                                                     base_job_name = 'Logistic-Demo-v1'
                                                    )

         train_instance_count has been renamed in sagemaker>=2.
         See: https://sagemaker.readthedocs.io/en/stable/v2.html for details.
         train_instance_type has been renamed in sagemaker>=2.
         See: https://sagemaker.readthedocs.io/en/stable/v2.html for details.

In [48]: LogisticModel.set_hyperparameters(predictor_type='binary_classifier', mini_batch_size=100)

In [49]: LogisticModel.hyperparameters()

Out[49]: {'predictor_type': 'binary_classifier', 'mini_batch_size': 100}

In [50]: trainConfig=sagemaker.session.s3_input(s3_data=s3Train,content_type='text/csv')
```

Building and Training the model on sagemaker


https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter
Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
conda_python3
Run
Code
nbdiff

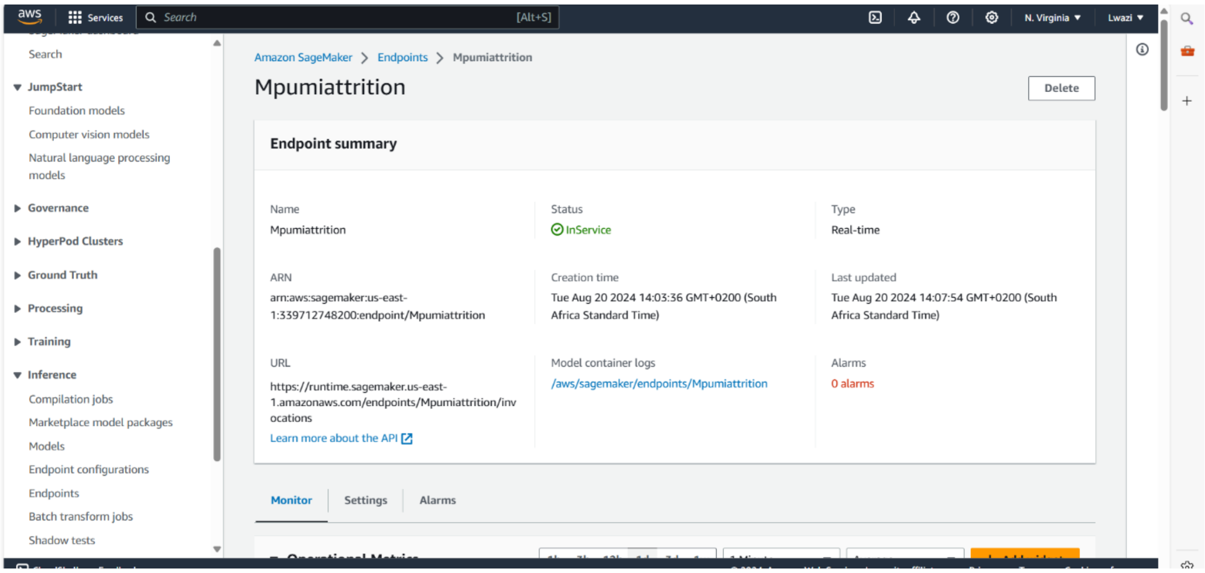
```
In [48]: LogisticModel.set_hyperparameters(predictor_type='binary_classifier', mini_batch_size=100)

In [49]: LogisticModel.hyperparameters()
Out[49]: {'predictor_type': 'binary_classifier', 'mini_batch_size': 100}

In [50]: trainConfig=sagemaker.session.s3_input(s3_data=s3Train,content_type='text/csv')

         The class sagemaker.session.s3_input has been renamed in sagemaker>=2.
         See: https://sagemaker.readthedocs.io/en/stable/v2.html for details.

In [53]: LogisticModel.fit({'train':trainConfig})

         INFO:sagemaker:Creating training-job with name: Logistic-Demo-v1-2024-08-20-11-49-43-630

         2024-08-20 11:49:43 Starting - Starting the training job......
         2024-08-20 11:50:20 Starting - Preparing the instances for training...
         2024-08-20 11:51:05 Downloading - Downloading input data...
         2024-08-20 11:51:30 Downloading - Downloading the training image.........
         2024-08-20 11:52:56 Training - Training image download completed. Training in progress..Docker entrypoint called with argumen
         t(s): train
         Running default environment configuration script
         [08/20/2024 11:53:11 INFO 140035362064192] Reading default configuration from /opt/amazon/lib/python3.8/site-packages/algorit
         hm/resources/default-input.json: {'mini_batch_size': '1000', 'epochs': '15', 'feature_dim': 'auto', 'use_bias': 'true', 'bina
         ry_classifier_model_selection_criteria': 'accuracy', 'f_beta': '1.0', 'target_recall': '0.8', 'target_precision': '0.8', 'num
         _models': 'auto', 'num_calibration_samples': '10000000', 'init_method': 'uniform', 'init_scale': '0.07', 'init_sigma': '0.0
         1', 'init_bias': '0.0', 'optimizer': 'auto', 'loss': 'auto', 'margin': '1.0', 'quantile': '0.5', 'loss_insensitivity': '0.0
         1', 'huber_delta': '1.0', 'num_classes': '1', 'accuracy_top_k': '3', 'wd': 'auto', 'l1': 'auto', 'momentum': 'auto', 'learnin
         g_rate': 'auto', 'beta_1': 'auto', 'beta_2': 'auto', 'bias_lr_mult': 'auto', 'bias_wd_mult': 'auto', 'use_lr_scheduler': 'tru
         e', 'lr_scheduler_step': 'auto', 'lr_scheduler_factor': 'auto', 'lr_scheduler_minimum_lr': 'auto', 'positive_example_weight_m
         ult': '1.0', 'balance_multiclass_weights': 'false', 'normalize_data': 'true', 'normalize_label': 'auto', 'unbias_data': 'aut
         o', 'unbias_label': 'auto', 'num_point_for_scaler': '10000', '_kvstore': 'auto', '_num_gpus': 'auto', '_num_kv_servers': 'aut
         o', '_log_level': 'info', '_tuning_objective_metric': '', 'early_stopping_patience': '3', 'early_stopping_tolerance': '0.00

In [56]: #Deploying the Trained Model

         predictmodel=LogisticModel.deploy(initial_instance_count=1, instance_type='ml.m4.xlarge',
```

Deploying the model after training


https://diabeticsdeplo.notebook.us-east-1.sagemaker.aws/notebooks/Untitled1.ipynb
jupyter
Untitled1 Last Checkpoint: Last Monday at 15:55 (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
conda_python3
Run
Code
nbdiff

```
         2024-08-20 11:51:05 Downloading - Downloading input data...
         2024-08-20 11:51:30 Downloading - Downloading the training image.........
         2024-08-20 11:52:56 Training - Training image download completed. Training in progress..Docker entrypoint called with argumen
         t(s): train
         Running default environment configuration script
         [08/20/2024 11:53:11 INFO 140035362064192] Reading default configuration from /opt/amazon/lib/python3.8/site-packages/algorit
         hm/resources/default-input.json: {'mini_batch_size': '1000', 'epochs': '15', 'feature_dim': 'auto', 'use_bias': 'true', 'bina
         ry_classifier_model_selection_criteria': 'accuracy', 'f_beta': '1.0', 'target_recall': '0.8', 'target_precision': '0.8', 'num
         _models': 'auto', 'num_calibration_samples': '10000000', 'init_method': 'uniform', 'init_scale': '0.07', 'init_sigma': '0.0
         1', 'init_bias': '0.0', 'optimizer': 'auto', 'loss': 'auto', 'margin': '1.0', 'quantile': '0.5', 'loss_insensitivity': '0.0
         1', 'huber_delta': '1.0', 'num_classes': '1', 'accuracy_top_k': '3', 'wd': 'auto', 'l1': 'auto', 'momentum': 'auto', 'learnin
         g_rate': 'auto', 'beta_1': 'auto', 'beta_2': 'auto', 'bias_lr_mult': 'auto', 'bias_wd_mult': 'auto', 'use_lr_scheduler': 'tru
         e', 'lr_scheduler_step': 'auto', 'lr_scheduler_factor': 'auto', 'lr_scheduler_minimum_lr': 'auto', 'positive_example_weight_m
         ult': '1.0', 'balance_multiclass_weights': 'false', 'normalize_data': 'true', 'normalize_label': 'auto', 'unbias_data': 'aut
         o', 'unbias_label': 'auto', 'num_point_for_scaler': '10000', '_kvstore': 'auto', '_num_gpus': 'auto', '_num_kv_servers': 'aut
         o', '_log_level': 'info', '_tuning_objective_metric': '', 'early_stopping_patience': '3', 'early_stopping_tolerance': '0.00

In [56]: #Deploying the Trained Model

         predictmodel=LogisticModel.deploy(initial_instance_count=1, instance_type='ml.m4.xlarge',
                                           endpoint_name = 'Mpumiattrition')

         INFO:sagemaker:Creating model with name: Logistic-Demo-v1-2024-08-20-12-03-34-882
         INFO:sagemaker:Creating endpoint-config with name Mpumiattrition
         INFO:sagemaker:Creating endpoint with name Mpumiattrition

         --------!

In [ ]:
```

The deployed model on AWS sagemaker



Mpumiattrition

Endpoint summary

Name
Mpumiattrition

Status
InService

Type
Real-time

ARN
arn:aws:sagemaker:us-east-1:339712748200:endpoint/Mpumiattrition

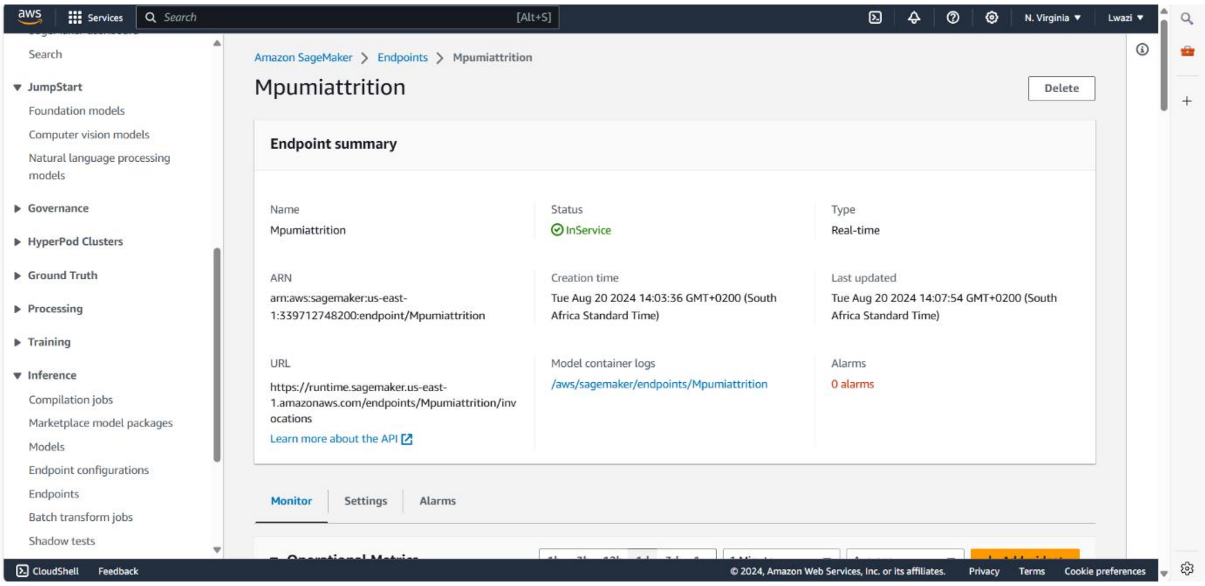
Creation time
Tue Aug 20 2024 14:03:36 GMT+0200 (South Africa Standard Time)

Last updated
Tue Aug 20 2024 14:07:54 GMT+0200 (South Africa Standard Time)

URL
https://runtime.sagemaker.us-east-1.amazonaws.com/endpoints/Mpumiattrition/invocations
Learn more about the API

Model container logs
/aws/sagemaker/endpoints/Mpumiattrition

Alarms
0 alarms



Mpumiattrition

Endpoint summary

Name
Mpumiattrition

Status
InService

Type
Real-time

ARN
arn:aws:sagemaker:us-east-1:339712748200:endpoint/Mpumiattrition

Creation time
Tue Aug 20 2024 14:03:36 GMT+0200 (South Africa Standard Time)

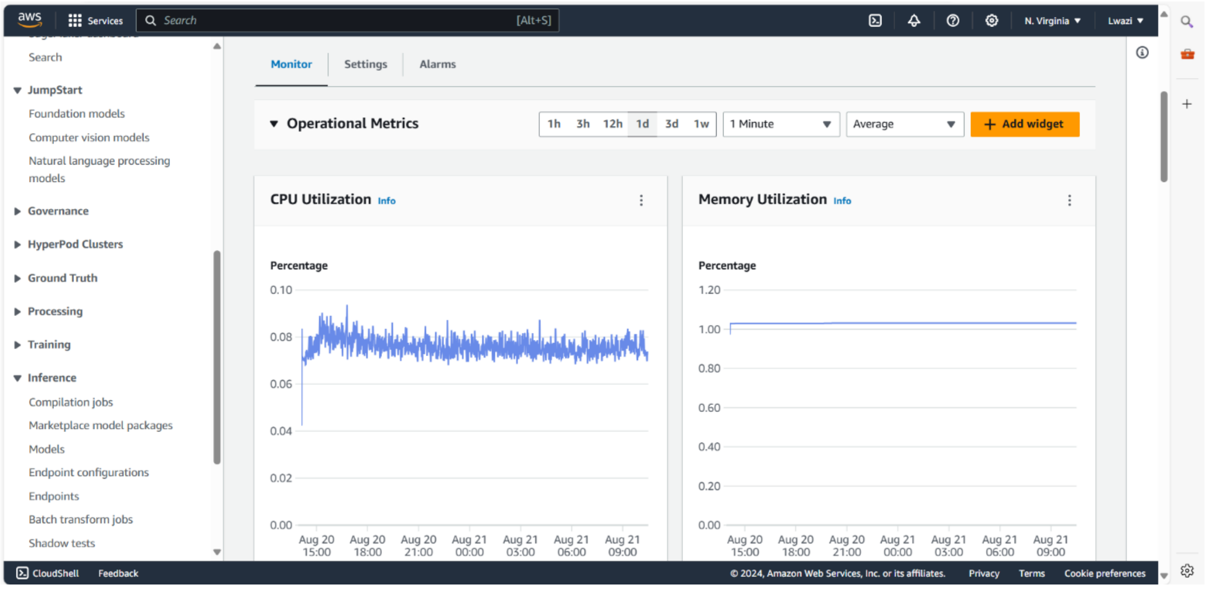
Last updated
Tue Aug 20 2024 14:07:54 GMT+0200 (South Africa Standard Time)

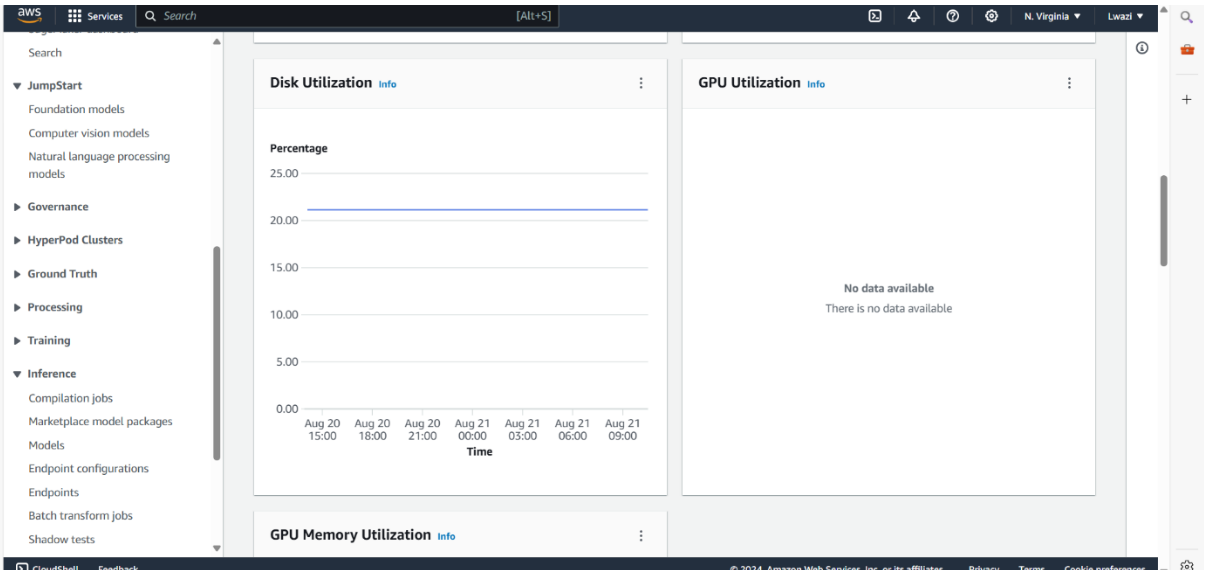
URL
https://runtime.sagemaker.us-east-1.amazonaws.com/endpoints/Mpumiattrition/invocations
Learn more about the API

Model container logs
/aws/sagemaker/endpoints/Mpumiattrition

Alarms
0 alarms

The metrics and monitoring of resources

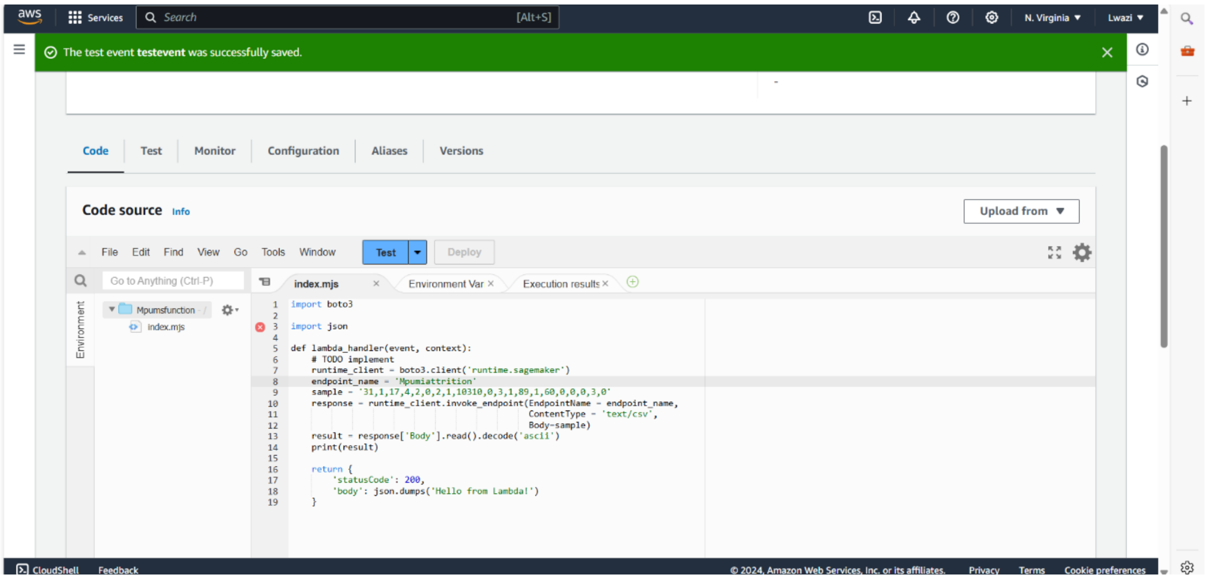




The end point link:

https://runtime.sagemaker.us-east-1.amazonaws.com/endpoints/Mpumiattrition/invocations

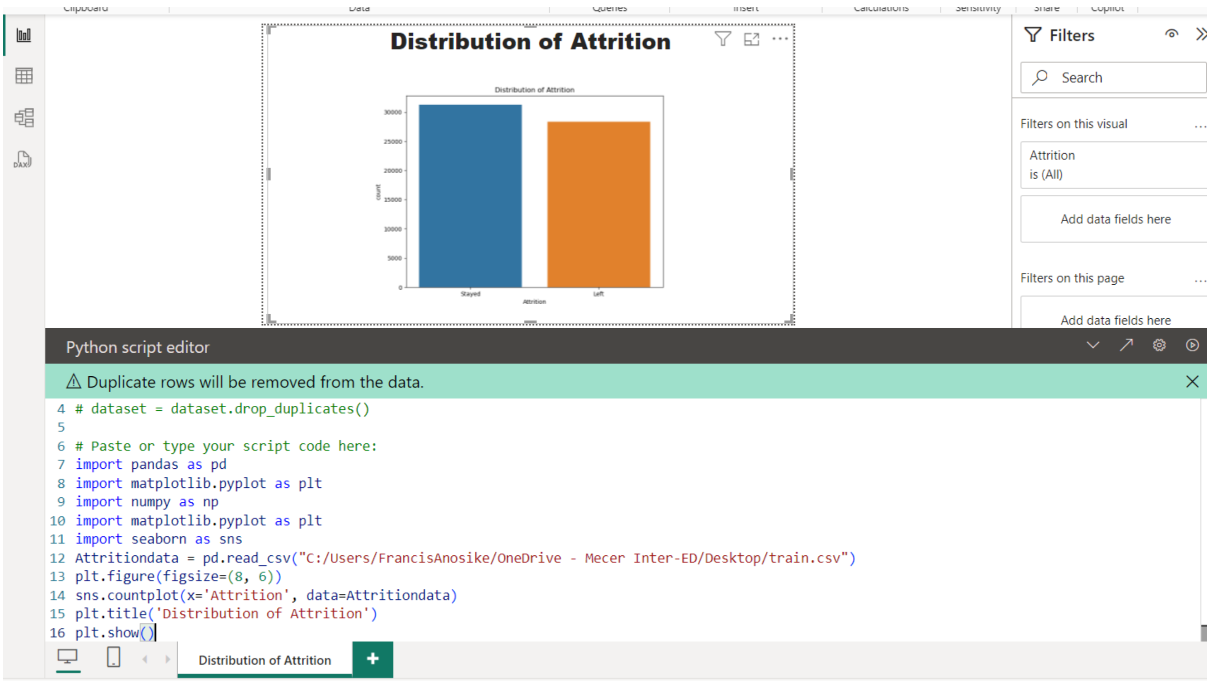
Testing the model on the lambda service on AWS:



```python
import boto3

import json

def lambda_handler(event, context):
    # TODO implement
    runtime_client = boto3.client('runtime.sagemaker')
    endpoint_name = 'Mpumiattrition'
    sample = '31,1,17,4,2,0,2,1,10310,0,3,1,89,1,60,0,0,0,3,0'
    response = runtime_client.invoke_endpoint(EndpointName = endpoint_name,
                                              ContentType = 'text/csv',
                                              Body=sample)
    result = response['Body'].read().decode('ascii')
    print(result)

    return {
        'statusCode': 200,
        'body': json.dumps('Hello from Lambda!')
    }
```

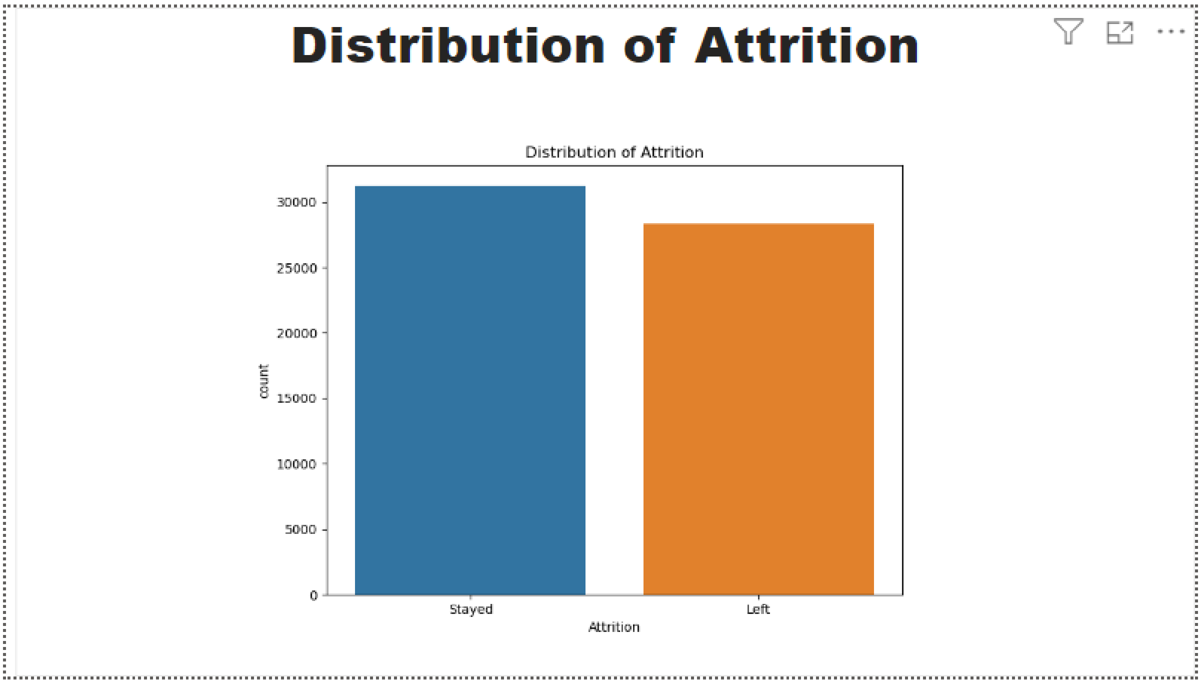
Visualizing the dataset using python scripts embedded in PowerBI

The employee attrition visual

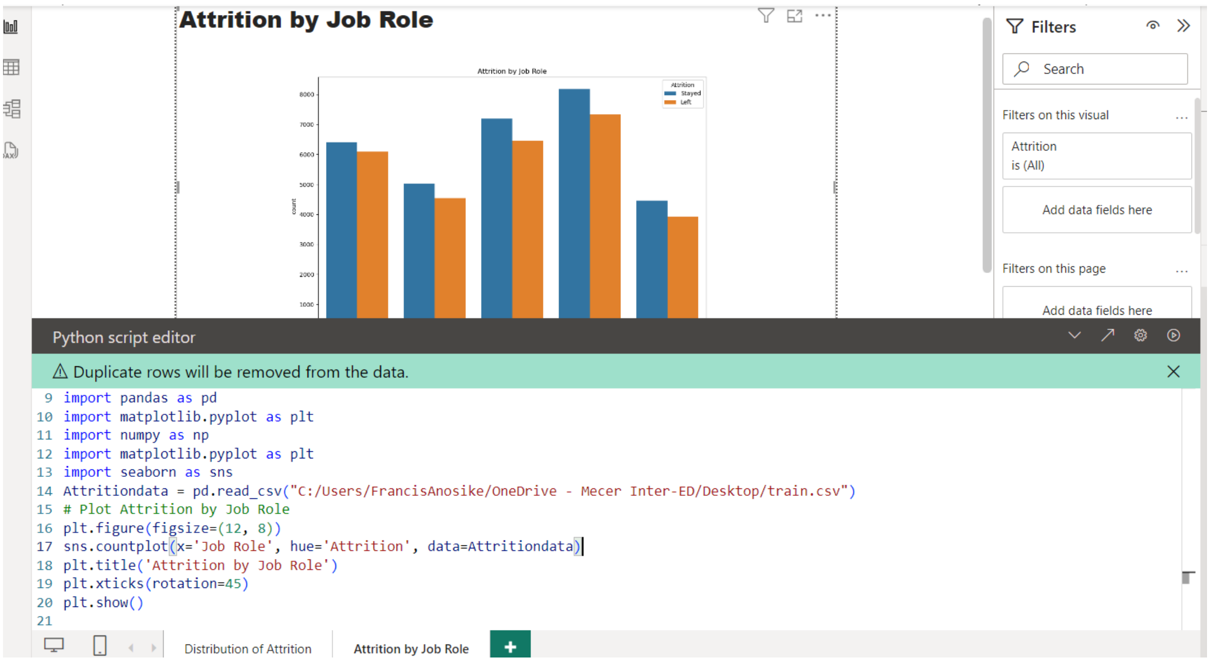


⚠ Duplicate rows will be removed from the data.

```python
# dataset = dataset.drop_duplicates()

# Paste or type your script code here:
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
Attritiondata = pd.read_csv("C:/Users/FrancisAnosike/OneDrive - Mecer Inter-ED/Desktop/train.csv")
plt.figure(figsize=(8, 6))
sns.countplot(x='Attrition', data=Attritiondata)
plt.title('Distribution of Attrition')
plt.show()
```

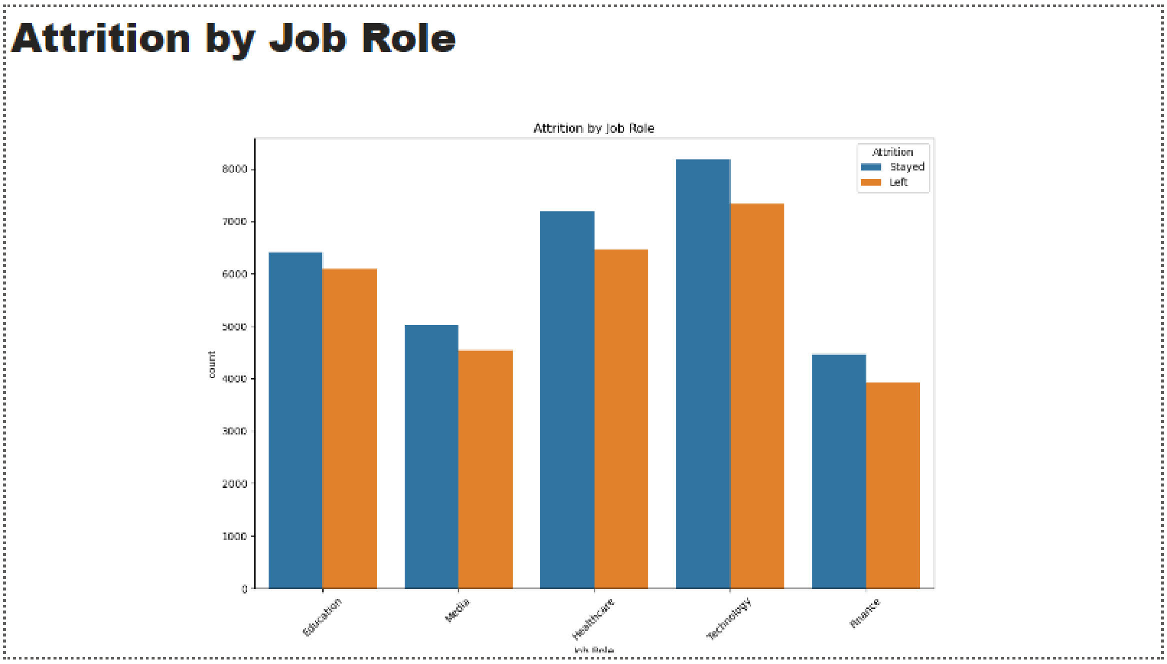


Employee Attrition by job role

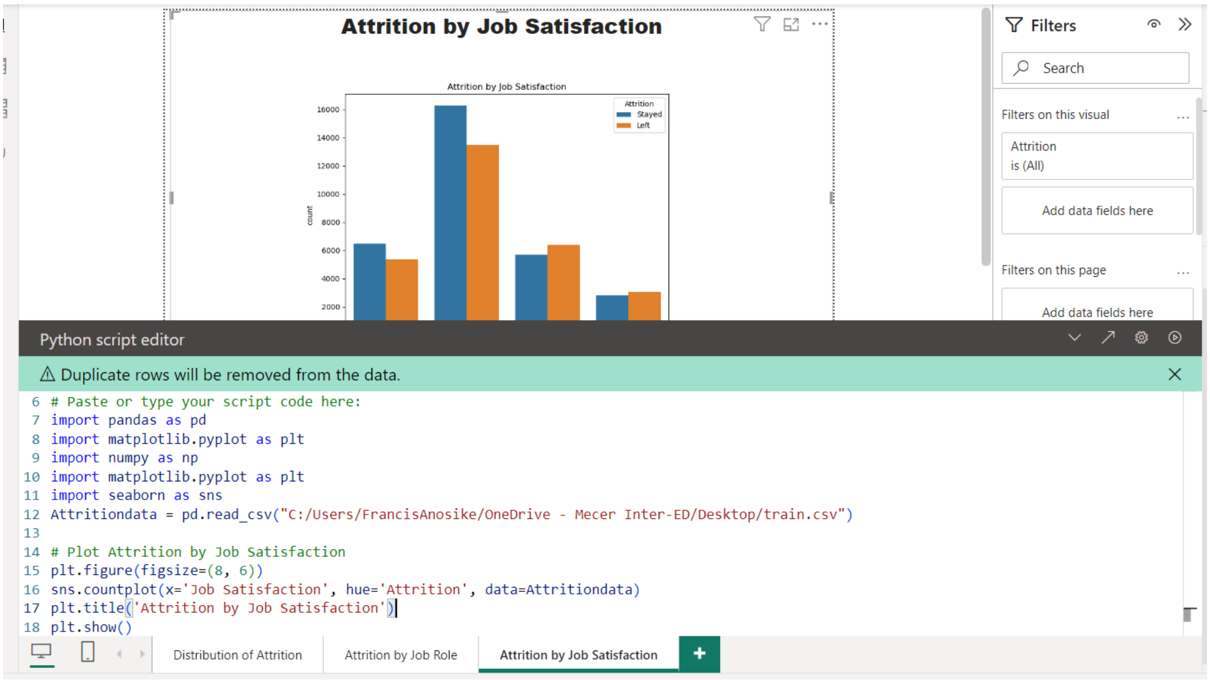


⚠ Duplicate rows will be removed from the data.

```python
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
Attritiondata = pd.read_csv("C:/Users/FrancisAnosike/OneDrive - Mecer Inter-ED/Desktop/train.csv")
# Plot Attrition by Job Role
plt.figure(figsize=(12, 8))
sns.countplot(x='Job Role', hue='Attrition', data=Attritiondata)
plt.title('Attrition by Job Role')
plt.xticks(rotation=45)
plt.show()
```

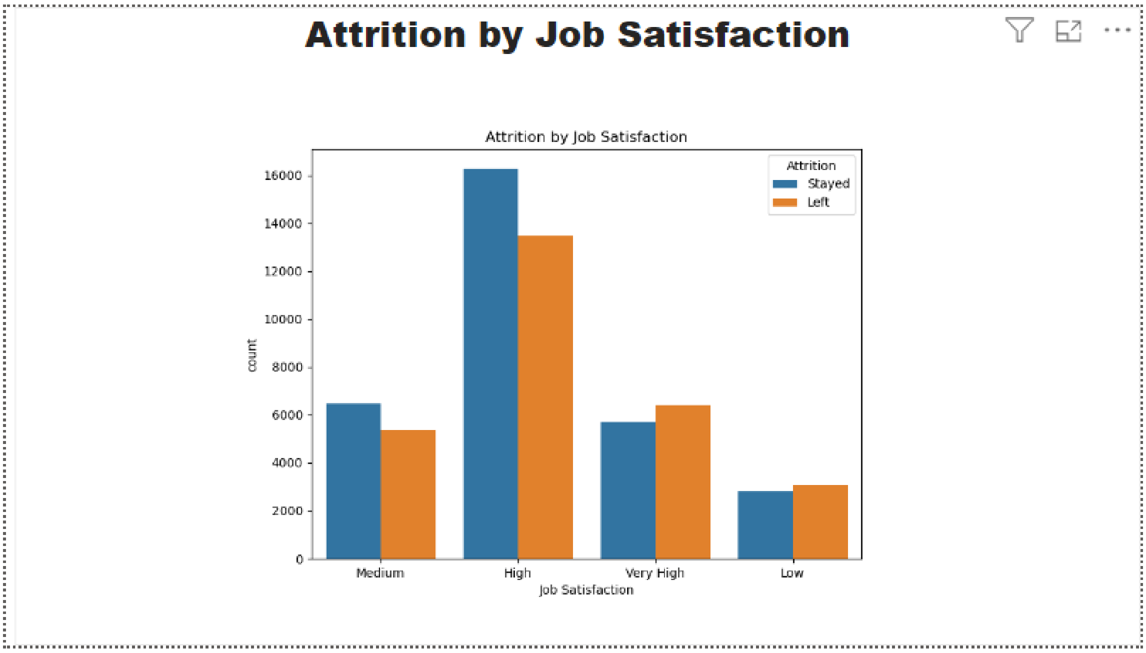


Employee attrition by job satisfaction.

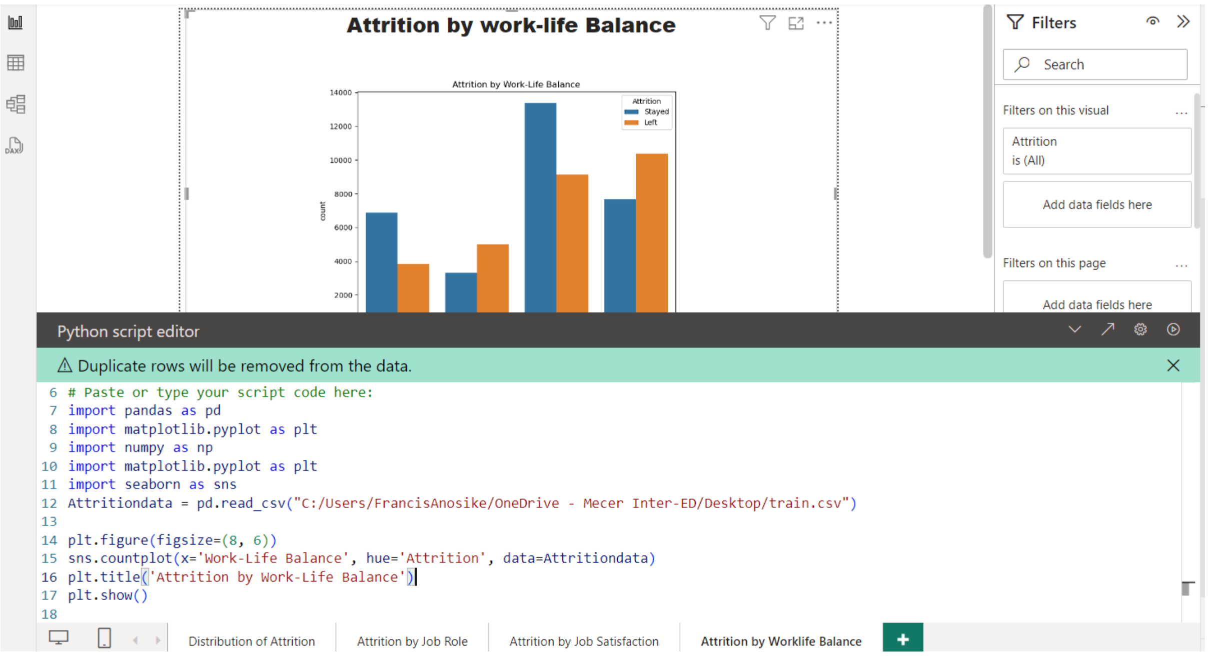


⚠ Duplicate rows will be removed from the data.

```
# Paste or type your script code here:
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
Attritiondata = pd.read_csv("C:/Users/FrancisAnosike/OneDrive - Mecer Inter-ED/Desktop/train.csv")

# Plot Attrition by Job Satisfaction
plt.figure(figsize=(8, 6))
sns.countplot(x='Job Satisfaction', hue='Attrition', data=Attritiondata)
plt.title('Attrition by Job Satisfaction')
plt.show()
```

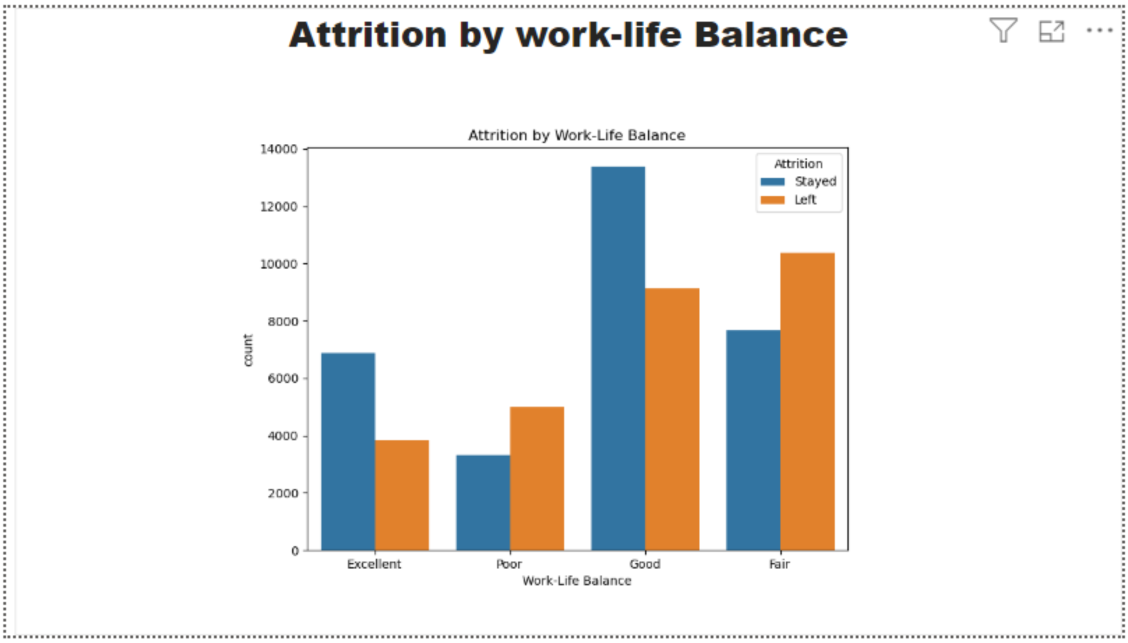


Employee attrition by work-life balance.

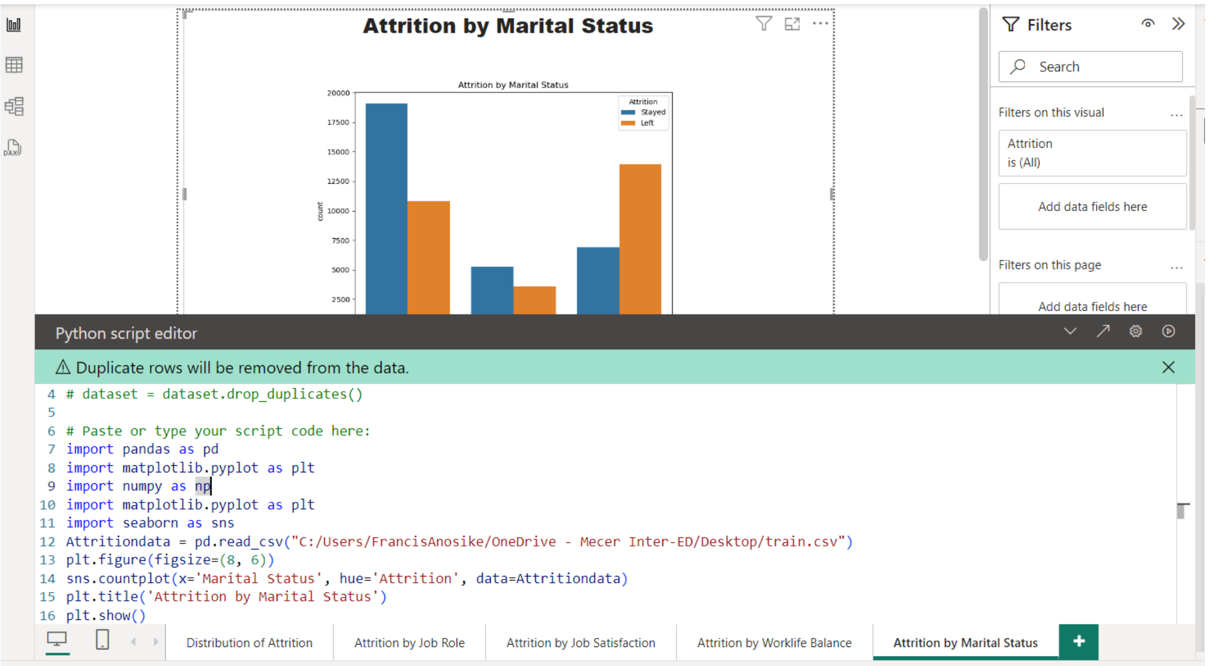


⚠ Duplicate rows will be removed from the data.

```
# Paste or type your script code here:
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
Attritiondata = pd.read_csv("C:/Users/FrancisAnosike/OneDrive - Mecer Inter-ED/Desktop/train.csv")

plt.figure(figsize=(8, 6))
sns.countplot(x='Work-Life Balance', hue='Attrition', data=Attritiondata)
plt.title('Attrition by Work-Life Balance')
plt.show()

```

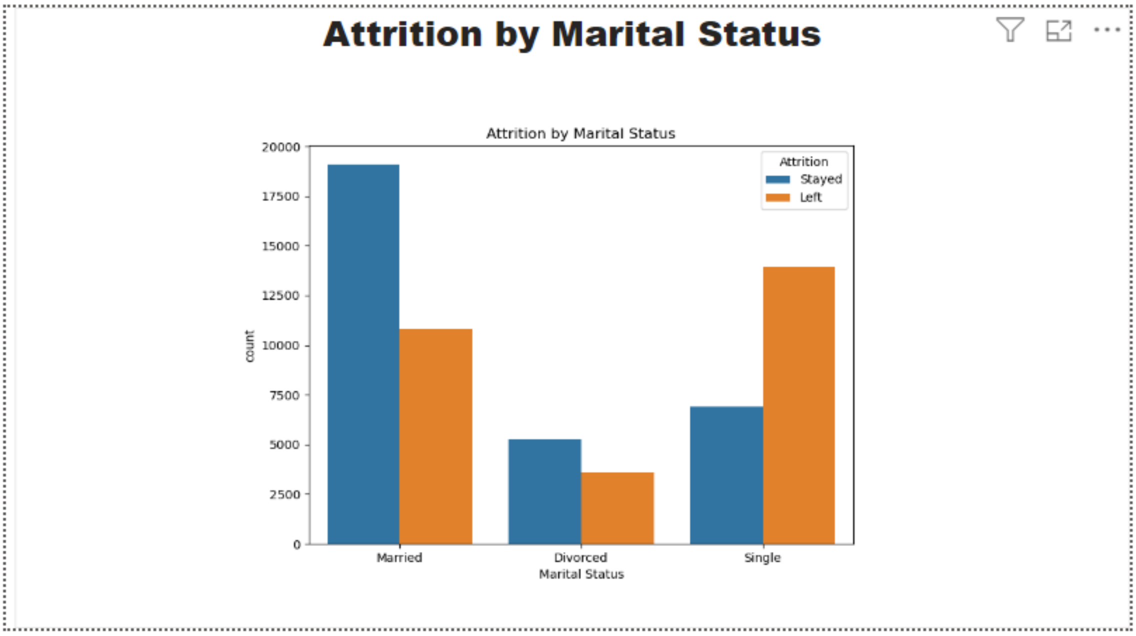


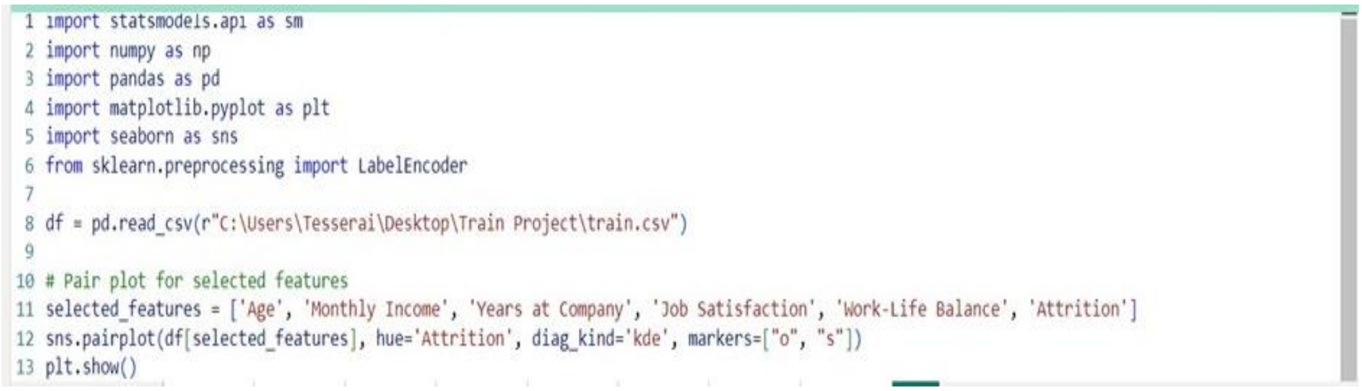
Employee attrition by marital status.



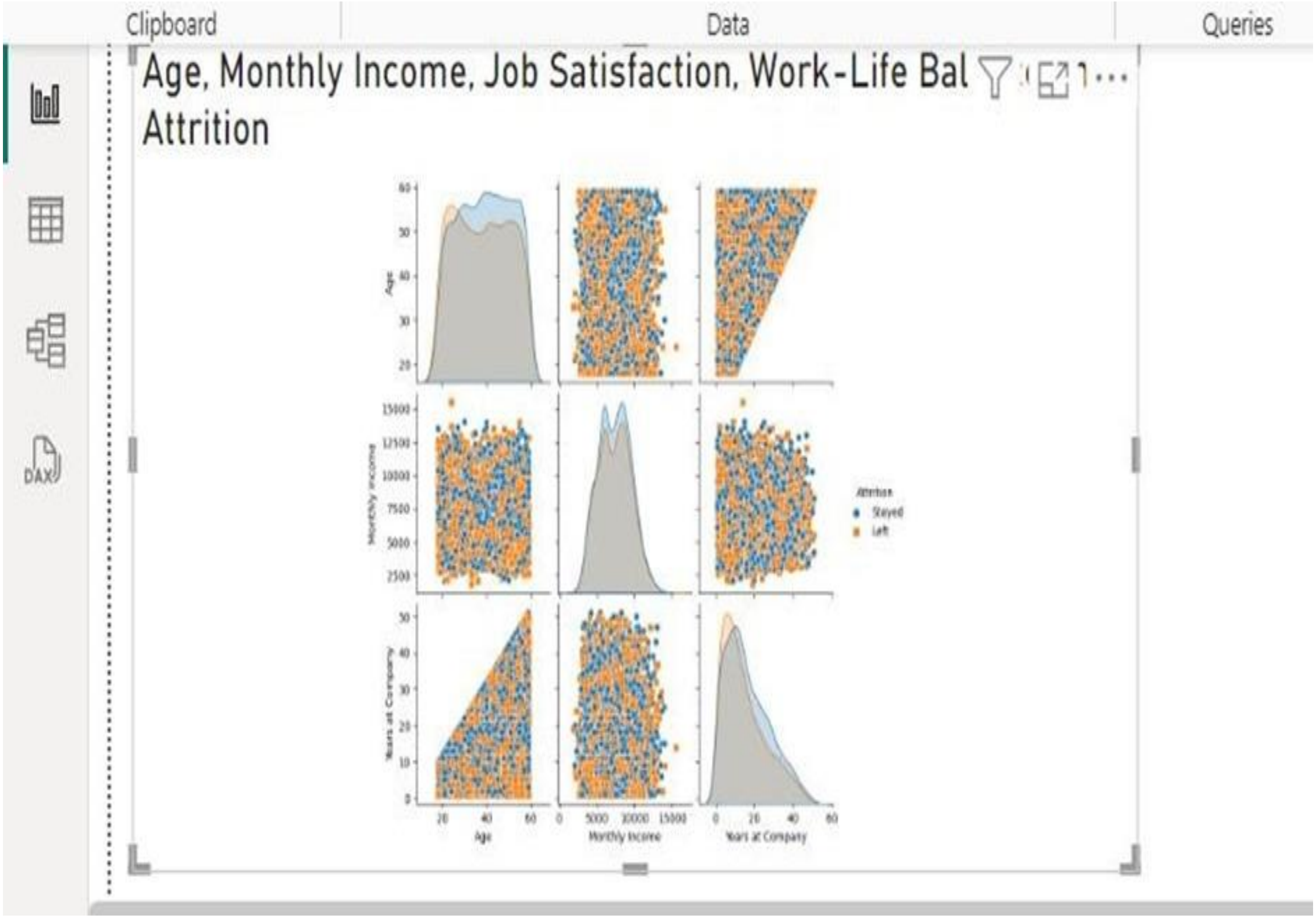
⚠ Duplicate rows will be removed from the data.

```
# dataset = dataset.drop_duplicates()

# Paste or type your script code here:
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
Attritiondata = pd.read_csv("C:/Users/FrancisAnosike/OneDrive - Mecer Inter-ED/Desktop/train.csv")
plt.figure(figsize=(8, 6))
sns.countplot(x='Marital Status', hue='Attrition', data=Attritiondata)
plt.title('Attrition by Marital Status')
plt.show()
```





```python
import statsmodels.api as sm
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import LabelEncoder

df = pd.read_csv(r"C:\Users\Tesserai\Desktop\Train Project\train.csv")

# Pair plot for selected features
selected_features = ['Age', 'Monthly Income', 'Years at Company', 'Job Satisfaction', 'Work-Life Balance', 'Attrition']
sns.pairplot(df[selected_features], hue='Attrition', diag_kind='kde', markers=["o", "s"])
plt.show()
```



Monitoring and Maintenance:

Once deployed, the model's performance needs to be monitored regularly to ensure that it continues to provide accurate predictions over time. This may involve updating the model with new data or retraining it periodically to maintain its accuracy.

Deployment Environment used for the deployment of the model:

Frameworks and Libraries:

NumPy: For numerical computations and array manipulations.

Pandas: For data manipulation and analysis, particularly useful for handling datasets like the Iris dataset.

Scikit-learn: For machine learning algorithms and model training. It includes logistic regression and utilities for model evaluation.

Development Tools:

IDEs: PowerBI, VS Code, or Jupyter notebook for coding and testing.

Version Control: Git for managing code versions.

Package Management: pip or conda for managing Python packages and dependencies.

Security Considerations:

Access Control: Restrict access to the deployed model and its endpoints. Implement role-based access control (RBAC) to ensure only authorized personnel can interact with the model.

Encryption: Use encryption mechanisms (e.g., HTTPS/TLS) to secure data transmission between clients and the deployed model, preventing eavesdropping and data tampering.

Input Validation: Validate input data to prevent injection attacks and ensure that only expected and sanitized data is processed by the model.