

Step 1

```
In [1]: !pip install pyspark
```

```
Requirement already satisfied: pyspark in c:\users\ts\appdata\local\programs\python\python37\lib\site-packages (3.3.2)
```

```
Requirement already satisfied: py4j==0.10.9.5 in c:\users\ts\appdata\local\programs\python\python37\lib\site-packages (from pyspark) (0.10.9.5)
```

```
[notice] A new release of pip is available: 23.0.1 -> 23.1
```

```
[notice] To update, run: python.exe -m pip install --upgrade pip
```

```
In [1]: from pyspark.sql import SparkSession
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd
import numpy as np
```

```
In [36]: spark = SparkSession.builder.appName('ml-diabetes').getOrCreate()
df = spark.read.csv('ML_hw_dataset.csv', header = True, inferSchema = True)
df.printSchema()
```

```
root
|-- age: integer (nullable = true)
|-- job: string (nullable = true)
|-- marital: string (nullable = true)
|-- education: string (nullable = true)
|-- default: string (nullable = true)
|-- housing: string (nullable = true)
|-- loan: string (nullable = true)
|-- contact: string (nullable = true)
|-- month: string (nullable = true)
|-- day_of_week: string (nullable = true)
|-- duration: integer (nullable = true)
|-- campaign: integer (nullable = true)
|-- pdays: integer (nullable = true)
|-- previous: integer (nullable = true)
|-- poutcome: string (nullable = true)
|-- emp_var_rate: double (nullable = true)
|-- cons_price_idx: double (nullable = true)
|-- cons_conf_idx: double (nullable = true)
|-- euribor3m: double (nullable = true)
|-- nr_employed: double (nullable = true)
|-- y: integer (nullable = true)
```

```
In [37]: df.show(10)
```

```

+---+-----+-----+-----+-----+-----+-----+-----+-----+
-----+-----+-----+-----+-----+-----+-----+-----+-----+
--+-----+-----+-----+-----+-----+
|age|          job| marital|          education|default|housing|loan|  contact|month|d
ay_of_week|duration|campaign|pdays|previous|    poutcome|emp_var_rate|cons_price_i
dx|cons_conf_idx|euribor3m|nr_employed|  y|
+---+-----+-----+-----+-----+-----+-----+-----+-----+
-----+-----+-----+-----+-----+-----+-----+-----+-----+
--+-----+-----+-----+-----+-----+
| 44|blue-collar| married|          basic.4y|unknown|    yes|  no|cellular| aug|
thu|    210|          1| 999|          0|nonexistent|    1.4|          93.444|
-36.1|    4.963|          5228.1| 0|
| 53| technician| married|          unknown|    no|    no|  no|cellular| nov|
fri|    138|          1| 999|          0|nonexistent|   -0.1|          93.2|
-42.0|    4.021|          5195.8| 0|
| 28| management| single|university.degree|    no|    yes|  no|cellular| jun|
thu|    339|          3| 6|          2|    success|   -1.7|          94.055|
-39.8|    0.729|          4991.6| 1|
| 39|  services| married|          high.school|    no|    no|  no|cellular| apr|
fri|    185|          2| 999|          0|nonexistent|   -1.8|          93.075|
-47.1|    1.405|          5099.1| 0|
| 55|  retired| married|          basic.4y|    no|    yes|  no|cellular| aug|
fri|    137|          1| 3|          1|    success|   -2.9|          92.201|
-31.4|    0.869|          5076.2| 1|
| 30| management|divorced|          basic.4y|    no|    yes|  no|cellular| jul|
tue|     68|          8| 999|          0|nonexistent|    1.4|          93.918|
-42.7|    4.961|          5228.1| 0|
| 37|blue-collar| married|          basic.4y|    no|    yes|  no|cellular| may|
thu|    204|          1| 999|          0|nonexistent|   -1.8|          92.893|
-46.2|    1.327|          5099.1| 0|
| 39|blue-collar|divorced|          basic.9y|    no|    yes|  no|cellular| may|
fri|    191|          1| 999|          0|nonexistent|   -1.8|          92.893|
-46.2|    1.313|          5099.1| 0|
| 36|   admin.| married|university.degree|    no|    no|  no|cellular| jun|
mon|    174|          1| 3|          1|    success|   -2.9|          92.963|
-40.8|    1.266|          5076.2| 1|
| 27|blue-collar| single|          basic.4y|    no|    yes|  no|cellular| apr|
thu|    191|          2| 999|          1|    failure|   -1.8|          93.075|
-47.1|    1.41|          5099.1| 0|
+---+-----+-----+-----+-----+-----+-----+-----+-----+
-----+-----+-----+-----+-----+-----+-----+-----+-----+
--+-----+-----+-----+-----+-----+
only showing top 10 rows

```

```
In [38]: df.groupby('y').count().show()
```

```

+---+-----+
|  y|count|
+---+-----+
|  1| 4640|
|  0|36548|
+---+-----+

```

```
In [39]: df.describe().show()
```

```

+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+
|summary|          age|    job| marital|education|default|housing| loan| co
ntact|month|day_of_week|    duration|    campaign|    pdays|
previous|poutcome|    emp_var_rate|    cons_price_idx|    cons_conf_idx|
euribor3m|    nr_employed|          y|
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+
| count|          41188| 41188| 41188| 41188| 41188| 41188|41188|
41188|41188|          41188|          41188|          41188|          41188|
41188| 41188|          41188|          41188|          41188|          41188|
41188|          41188|          41188|
| mean| 40.02406040594348| null| null| null| null| null| null|
null| null| null|258.2850101971448| 2.567592502670681| 962.4754540157328|0.
17296299893172767| null|0.08188550063125699| 93.57566436827325|-40.50260027191
949|3.6212908128582826|5167.035910942904|0.11265417111780131|
| stddev|10.421249980934071| null| null| null| null| null| null|
null| null| null|259.2792488364662|2.7700135429023405|186.91090734474142|0.
49490107983929005| null| 1.5709597405170228|0.5788400489541238| 4.628197856174
544| 1.734447404851268|72.25152766826125| 0.3161734269429653|
| min|          17| admin.|divorced| basic.4y|    no|    no|    no| cel
lular| apr|    fri|          0|          1|          0|
0| failure|          -3.4|          92.201|          -50.8|
0.634|          4963.6|          0|
| max|          98|unknown| unknown| unknown|    yes|    yes|    yes|tele
phone| sep|    wed|          4918|          56|          999|
7| success|          1.4|          94.767|          -26.9|
5.045|          5228.1|          1|
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+

```

```
In [40]: df.columns
```

```
Out[40]: ['age',
          'job',
          'marital',
          'education',
          'default',
          'housing',
          'loan',
          'contact',
          'month',
          'day_of_week',
          'duration',
          'campaign',
          'pdays',
          'previous',
          'poutcome',
          'emp_var_rate',
          'cons_price_idx',
          'cons_conf_idx',
          'euribor3m',
          'nr_employed',
          'y']
```

Fill Null values

```
In [41]: ## See if we have missing values
from pyspark.sql.functions import when, count
df.select([count(when(df[c]==None, c)).alias(c) for c in df.columns]).show()
```

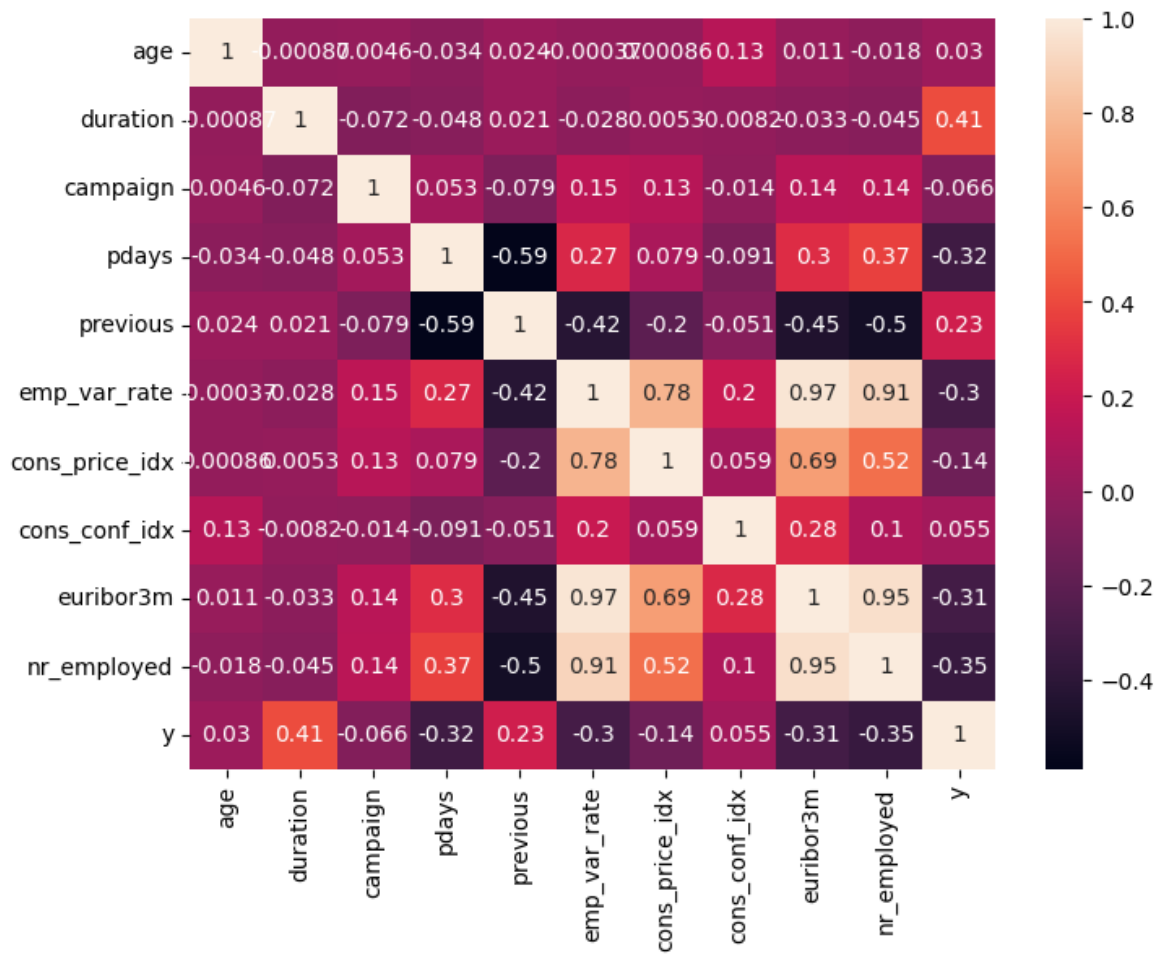
```
+---+---+-----+-----+-----+-----+---+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
|age|job|marital|education|default|housing|loan|contact|month|day_of_week|duration|campaign|pdays|previous|poutcome|emp_var_rate|cons_price_idx|cons_conf_idx|euribor3m|nr_employed| y|
+---+---+-----+-----+-----+-----+---+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
| 0| 0| 0| 0| 0| 0| 0| 0| 0| 0| 0| 0| 0|
0| 0| 0| 0| 0| 0| 0| 0| 0| 0| 0|
0| 0| 0|
+---+---+-----+-----+-----+-----+---+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+
+---+-----+-----+
```

Correlation Matrix

```
In [42]: Correlation_matrix = df.toPandas().corr()
fig, ax = plt.subplots(figsize=(8, 6))
sns.heatmap(Correlation_matrix, annot=True)
plt.show()
```

```
<ipython-input-42-738b4ffef13b>:1: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future version, it will default to False. Select only valid columns or specify the value of numeric_only to silence this warning.
```

```
Correlation_matrix = df.toPandas().corr()
```



Step 2

Indexing and One-Hot-Encoding

```
In [43]: from pyspark.ml.feature import StringIndexer
from pyspark.ml.feature import VectorAssembler
from pyspark.ml.functions import vector_to_array
import pyspark.sql.functions as F
from pyspark.ml import Pipeline
from pyspark.pandas import concat, Series
```

```
In [44]: df = StringIndexer(inputCol='job', outputCol='si_job').fit(df).transform(df)
df = StringIndexer(inputCol='marital', outputCol='si_marital').fit(df).transform(df)
df = StringIndexer(inputCol='education', outputCol='si_education').fit(df).transform(df)
df = StringIndexer(inputCol='default', outputCol='si_default').fit(df).transform(df)
df = StringIndexer(inputCol='housing', outputCol='si_housing').fit(df).transform(df)
df = StringIndexer(inputCol='loan', outputCol='si_loan').fit(df).transform(df)
df = StringIndexer(inputCol='contact', outputCol='si_contact').fit(df).transform(df)
df = StringIndexer(inputCol='poutcome', outputCol='si_poutcome').fit(df).transform(df)
```

```
df = StringIndexer(inputCol='month', outputCol='si_month').fit(df).transform(df)
df = StringIndexer(inputCol='day_of_week', outputCol='si_day').fit(df).transform
```

Select Features for Classification

In [47]: `df.show()`

```

+---+-----+-----+-----+-----+-----+-----+-----+-----+
-----+-----+-----+-----+-----+-----+-----+-----+-----+
---+-----+-----+-----+-----+-----+-----+-----+-----+
-----+-----+-----+-----+-----+-----+-----+-----+
|age|          job| marital|          education|default|housing|loan|  contact|month|
day_of_week|duration|campaign|pdays|previous|  poutcome|emp_var_rate|cons_price_
idx|cons_conf_idx|euribor3m|nr_employed|  y|si_job|si_marital|si_education|si_def
ault|si_housing|si_loan|si_contact|si_poutcome|si_month|si_day|
+---+-----+-----+-----+-----+-----+-----+-----+-----+
-----+-----+-----+-----+-----+-----+-----+-----+-----+
---+-----+-----+-----+-----+-----+-----+-----+-----+
-----+-----+-----+-----+-----+-----+-----+-----+
| 44|blue-collar| married|          basic.4y|unknown|  yes|  no| cellular| aug|
thu|    210|          1| 999|          0|nonexistent|    1.4|    93.444|
-36.1|    4.963|    5228.1| 0|    1.0|          0.0|    4.0|    1.0|
0.0|    0.0|          0.0|    0.0|    2.0|    0.0|
| 53|technician| married|          unknown|  no|  no| no| cellular| nov|
fri|    138|          1| 999|          0|nonexistent|   -0.1|    93.2|
-42.0|    4.021|    5195.8| 0|    2.0|          0.0|    6.0|    0.0|
1.0|    0.0|          0.0|    0.0|    4.0|    4.0|
| 28|management| single|university.degree|  no| yes|  no| cellular| jun|
thu|    339|          3| 6|          2| success|   -1.7|    94.055|
-39.8|    0.729|    4991.6| 1|    4.0|          1.0|    0.0|    0.0|
0.0|    0.0|          0.0|    2.0|    3.0|    0.0|
| 39|services| married|          high.school|  no|  no| no| cellular| apr|
fri|    185|          2| 999|          0|nonexistent|   -1.8|    93.075|
-47.1|    1.405|    5099.1| 0|    3.0|          0.0|    1.0|    0.0|
1.0|    0.0|          0.0|    0.0|    5.0|    4.0|
| 55|retired| married|          basic.4y|  no| yes|  no| cellular| aug|
fri|    137|          1| 3|          1| success|   -2.9|    92.201|
-31.4|    0.869|    5076.2| 1|    5.0|          0.0|    4.0|    0.0|
0.0|    0.0|          0.0|    2.0|    2.0|    4.0|
| 30|management|divorced|          basic.4y|  no| yes|  no| cellular| jul|
tue|     68|          8| 999|          0|nonexistent|    1.4|    93.918|
-42.7|    4.961|    5228.1| 0|    4.0|          2.0|    4.0|    0.0|
0.0|    0.0|          0.0|    0.0|    1.0|    3.0|
| 37|blue-collar| married|          basic.4y|  no| yes|  no| cellular| may|
thu|    204|          1| 999|          0|nonexistent|   -1.8|    92.893|
-46.2|    1.327|    5099.1| 0|    1.0|          0.0|    4.0|    0.0|
0.0|    0.0|          0.0|    0.0|    0.0|    0.0|
| 39|blue-collar|divorced|          basic.9y|  no| yes|  no| cellular| may|
fri|    191|          1| 999|          0|nonexistent|   -1.8|    92.893|
-46.2|    1.313|    5099.1| 0|    1.0|          2.0|    2.0|    0.0|
0.0|    0.0|          0.0|    0.0|    0.0|    4.0|
| 36|admin.| married|university.degree|  no|  no| no| cellular| jun|
mon|    174|          1| 3|          1| success|   -2.9|    92.963|
-40.8|    1.266|    5076.2| 1|    0.0|          0.0|    0.0|    0.0|
1.0|    0.0|          0.0|    2.0|    3.0|    1.0|
| 27|blue-collar| single|          basic.4y|  no| yes|  no| cellular| apr|
thu|    191|          2| 999|          1| failure|   -1.8|    93.075|
-47.1|    1.41|    5099.1| 0|    1.0|          1.0|    4.0|    0.0|
0.0|    0.0|          0.0|    1.0|    5.0|    0.0|
| 34|housemaid| single|university.degree|  no|  no| no| telephone| may|
fri|     62|          2| 999|          0|nonexistent|    1.1|    93.994|
-36.4|    4.864|    5191.0| 0|    8.0|          1.0|    0.0|    0.0|
1.0|    0.0|          1.0|    0.0|    0.0|    4.0|
| 41|management| married|university.degree|  no| yes|  no| cellular| aug|
thu|    789|          1| 999|          0|nonexistent|    1.4|    93.444|
-36.1|    4.964|    5228.1| 0|    4.0|          0.0|    0.0|    0.0|
0.0|    0.0|          0.0|    0.0|    2.0|    0.0|

```

55	management	married	university.degree	no	no	no	cellular	aug
mon	372	3	999	0 nonexistent	1.4		93.444	
-36.1	4.965	5228.1	1	4.0	0.0	0.0	0.0	
1.0	0.0	0.0	0.0	2.0	1.0			
33	services	divorced	high.school	no	yes	no	cellular	may
tue	75	5	999	0 nonexistent	-1.8		92.893	
-46.2	1.291	5099.1	0	3.0	2.0	1.0	0.0	
0.0	0.0	0.0	0.0	0.0	3.0			
26	admin.	married	high.school	no	no	yes	telephone	jun
mon	1021	1	999	0 nonexistent	1.4		94.465	
-41.8	4.96	5228.1	0	0.0	0.0	1.0	0.0	
1.0	1.0	1.0	0.0	3.0	1.0			
52	services	married	high.school	unknown	yes	no	cellular	jul
thu	117	2	999	0 nonexistent	1.4		93.918	
-42.7	4.962	5228.1	0	3.0	0.0	1.0	1.0	
0.0	0.0	0.0	0.0	1.0	0.0			
35	services	married	high.school	no	no	no	cellular	apr
thu	1034	2	999	0 nonexistent	-1.8		93.075	
-47.1	1.365	5099.1	1	3.0	0.0	1.0	0.0	
1.0	0.0	0.0	0.0	5.0	0.0			
27	admin.	single	university.degree	no	no	no	telephone	oct
tue	540	1	999	0 nonexistent	-0.1		93.798	
-40.4	4.86	5195.8	1	0.0	1.0	0.0	0.0	
1.0	0.0	1.0	0.0	6.0	3.0			
28	blue-collar	married	basic.9y	unknown	no	no	telephone	may
thu	140	1	999	0 nonexistent	1.1		93.994	
-36.4	4.86	5191.0	0	1.0	0.0	2.0	1.0	
1.0	0.0	1.0	0.0	0.0	0.0			
26	unemployed	single	basic.9y	no	yes	yes	cellular	jul
mon	104	4	999	0 nonexistent	1.4		93.918	
-42.7	4.96	5228.1	0	9.0	1.0	2.0	0.0	
0.0	1.0	0.0	0.0	1.0	1.0			

+-----+-----+-----+-----+-----+-----+-----+-----+-----+

-----+-----+-----+-----+-----+-----+-----+-----+-----+

---+-----+-----+-----+-----+-----+-----+-----+-----+-----+

---+-----+-----+-----+-----+-----+-----+-----+-----+-----+

only showing top 20 rows

```
In [48]: assembler = VectorAssembler(inputCols = ['age', 'duration', 'campaign', 'pdays',
                                                'emp_var_rate', 'cons_price_idx', 'cons
                                                'si_job', 'si_marital', 'si_education',
                                                'si_contact', 'si_poutcome', 'si_month'
                                                outputCol = 'features')
output = assembler.transform(df)
final_data = output.select('features', 'y')
```

```
In [49]: final_data.show()
```



```

+-----+-----+
|          features|  y|
+-----+-----+
|[44.0,210.0,1.0,9...|  0|
|[53.0,138.0,1.0,9...|  0|
|[28.0,339.0,3.0,6...|  1|
|[39.0,185.0,2.0,9...|  0|
|[55.0,137.0,1.0,3...|  1|
|[30.0,68.0,8.0,99...|  0|
|(20,[0,1,2,3,5,6,...|  0|
|[39.0,191.0,1.0,9...|  0|
|[36.0,174.0,1.0,3...|  1|
|[27.0,191.0,2.0,9...|  0|
|[34.0,62.0,2.0,99...|  0|
|(20,[0,1,2,3,5,6,...|  0|
|[55.0,372.0,3.0,9...|  1|
|[33.0,75.0,5.0,99...|  0|
|[26.0,1021.0,1.0,...|  0|
|[52.0,117.0,2.0,9...|  0|
|[35.0,1034.0,2.0,...|  1|
|[27.0,540.0,1.0,9...|  1|
|[28.0,140.0,1.0,9...|  0|
|[26.0,104.0,4.0,9...|  0|
+-----+-----+
only showing top 20 rows

```

```
In [50]: (trainingData, testData) = final_data.randomSplit([0.8, 0.2])
```

Modeling

```
In [51]: from pyspark.ml.evaluation import BinaryClassificationEvaluator
binary_evaluator = BinaryClassificationEvaluator(rawPredictionCol='prediction',
```

Step 3

Logestic Regression

```
In [52]: from pyspark.ml.classification import LogisticRegression

lr = LogisticRegression(labelCol="y", featuresCol="features", maxIter=10)
model = lr.fit(trainingData)

predictions = model.transform(testData)
print('Logistic Regression Accuracy:', binary_evaluator.evaluate(predictions))
```

Logistic Regression Accuracy: 0.6724903607821537

Step 4

Decision Tree

```
In [54]: from pyspark.ml.classification import DecisionTreeClassifier

dt = DecisionTreeClassifier(labelCol="y", featuresCol="features", maxDepth=5)
model = dt.fit(trainingData)

predictions = model.transform(testData)
print('Decision Tree classifier Accuracy:', binary_evaluator.evaluate(predictions))
```

Decision Tree classifier Accuracy: 0.7521323208197608

Random Forest

```
In [55]: from pyspark.ml.classification import RandomForestClassifier

rf = RandomForestClassifier(labelCol="y", featuresCol="features", maxDepth=5)
model = rf.fit(trainingData)

predictions = model.transform(testData)
print('Random Forest classifier Accuracy:', binary_evaluator.evaluate(predictions))
```

Random Forest classifier Accuracy: 0.6136908351456329

classifier Accuracy: Random Forest(0.61369) < Decision Tree(0.7521)