Pyspark Classification-SmallerYet-QUESTIONS

May 5, 2022

1 Logistic Regression with PySpark

[1]:
```
import os;


os.environ["SPARK_HOME"] = "/Users/guibs/AppData/Local/Packages/
 ↪PythonSoftwareFoundation.Python.3.10_qbz5n2kfra8p0/LocalCache/local-packages/
 ↪Python310/site-packages/pyspark"
#os.environ["JAVA_HOME"] ="/Library/Java/JavaVirtualMachines/adoptopenjdk-15.
 ↪jdk/Contents/Home"

#os.environ["SPARK_HOME"] = "/Users/pedro/servers/spark-3.1.1-bin-hadoop2.7"
#os.environ["JAVA_HOME"]="/Library/Java/JavaVirtualMachines/adoptopenjdk-8.jdk/
 ↪Contents/Home"
!java -version
```

```
java version "1.8.0_202"
Java(TM) SE Runtime Environment (build 1.8.0_202-b08)
Java HotSpot(TM) 64-Bit Server VM (build 25.202-b08, mixed mode)
```

[2]:
```
#import findspark
#findspark.init()
```

[3]:
```
from pyspark.sql import SparkSession
from pyspark.conf import SparkConf
from pyspark.sql.types import *
import pyspark.sql.functions as F
from pyspark.sql.functions import col, asc,desc
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
from pyspark.sql import SQLContext
from pyspark.mllib.stat import Statistics
import pandas as pd
from pyspark.sql.functions import udf
from pyspark.ml.feature import OneHotEncoder, StringIndexer,␣
 ↪VectorAssembler,StandardScaler
```

```
from pyspark.ml import Pipeline
from sklearn.metrics import confusion_matrix

spark = SparkSession.builder.appName("DataFrame").getOrCreate()
```

sc=spark.sparkContext sqlContext=SQLContext(sc)

1.1 Read File

[4]:
```
df=spark.read \
 .option("header","True")\
 .option("inferSchema","True")\
 .option("sep",",")\
 .csv("diabetes.csv")
print("There are",df.count(),"rows",len(df.columns),
      "columns" ,"in the data.")
```

```
There are 768 rows 9 columns in the data.
```

1.2 Show Sample Data

[5]:
```
df.show(4)
```

```
+-----------+-------+-------------+-------------+-------+----+------------------
------+---+-------+
|Pregnancies|Glucose|BloodPressure|SkinThickness|Insulin|
BMI|DiabetesPedigreeFunction|Age|Outcome|
+-----------+-------+-------------+-------------+-------+----+------------------
------+---+-------+
|          6|    148|           72|           35|      0|33.6|
0.627| 50|      1|
|          1|     85|           66|           29|      0|26.6|
0.351| 31|      0|
|          8|    183|           64|            0|      0|23.3|
0.672| 32|      1|
|          1|     89|           66|           23|     94|28.1|
0.167| 21|      0|
+-----------+-------+-------------+-------------+-------+----+------------------
------+---+-------+
only showing top 4 rows
```

1.3 Data Types of Columns

[6]:
```
df.printSchema()
```

```
root
 |-- Pregnancies: integer (nullable = true)
```

```
 |-- Glucose: integer (nullable = true)
 |-- BloodPressure: integer (nullable = true)
 |-- SkinThickness: integer (nullable = true)
 |-- Insulin: integer (nullable = true)
 |-- BMI: double (nullable = true)
 |-- DiabetesPedigreeFunction: double (nullable = true)
 |-- Age: integer (nullable = true)
 |-- Outcome: integer (nullable = true)
```

1.4 Statistics

[7]:
```
numeric_features = [t[0] for t in df.dtypes if t[1] == 'int']
df.select(numeric_features).describe().toPandas().transpose()
```

[7]:

	0	1	2	3	4
summary	count	mean	stddev	min	max
Pregnancies	768	3.8450520833333335	3.36957806269887	0	17
Glucose	768	120.89453125	31.97261819513622	0	199
BloodPressure	768	69.10546875	19.355807170644777	0	122
SkinThickness	768	20.536458333333332	15.952217567727642	0	99
Insulin	768	79.79947916666667	115.24400235133803	0	846
Age	768	33.240885416666664	11.760231540678689	21	81
Outcome	768	0.3489583333333333	0.476951377242799	0	1

[8]:
```
from pyspark.sql.functions import when
df=df.withColumn("Glucose",when(df.Glucose==0,np.nan).otherwise(df.Glucose))
df=df.withColumn("BloodPressure",when(df.BloodPressure==0,np.nan).otherwise(df.
 ↪BloodPressure))
df=df.withColumn("SkinThickness",when(df.SkinThickness==0,np.nan).otherwise(df.
 ↪SkinThickness))
df=df.withColumn("BMI",when(df.BMI==0,np.nan).otherwise(df.BMI))
df=df.withColumn("Insulin",when(df.Insulin==0,np.nan).otherwise(df.Insulin))
```

[9]:
```
from pyspark.ml.feature import Imputer
imputer=Imputer(inputCols=["Glucose","BloodPressure","SkinThickness","BMI","Insulin"],outputCo
model=imputer.fit(df)
raw_data=model.transform(df)
raw_data.show(5)
```

```
+-----------+-------+-------------+-----------------+-----------------+----+---
----------------------+---+-------+
|Pregnancies|Glucose|BloodPressure|     SkinThickness|          Insulin|
BMI|DiabetesPedigreeFunction|Age|Outcome|
+-----------+-------+-------------+-----------------+-----------------+----+---
----------------------+---+-------+
|          6|  148.0|         72.0|             35.0|155.5482233502538|33.6|
0.627| 50|      1|
```

```
|          1|   85.0|         66.0|               29.0|155.5482233502538|26.6|
0.351| 31|      0|
|          8|  183.0|         64.0|29.153419593345657|155.5482233502538|23.3|
0.672| 32|      1|
|          1|   89.0|         66.0|               23.0|             94.0|28.1|
0.167| 21|      0|
|          0|  137.0|         40.0|               35.0|            168.0|43.1|
2.288| 33|      1|
+-----------+-------+-------------+--------------------+-----------------+----+---
---------------------+---+-------+
only showing top 5 rows
```

1.5 Target Variable Distribution

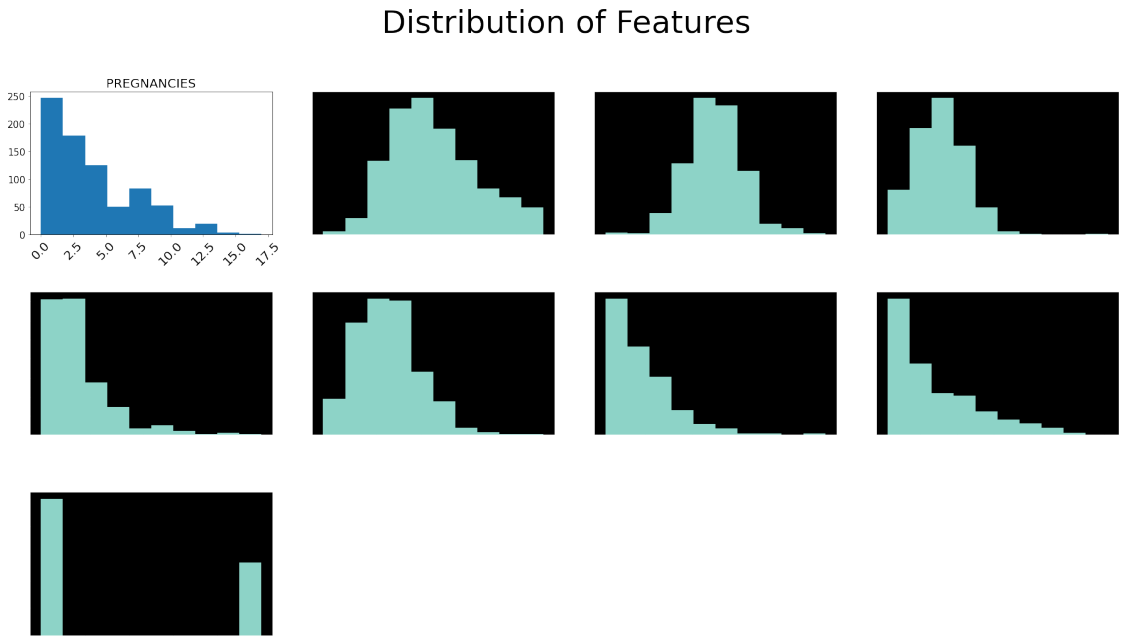
[10]:
```
df.groupby("Outcome").count().show()
```

```
+-------+-----+
|Outcome|count|
+-------+-----+
|      1|  268|
|      0|  500|
+-------+-----+
```

1.6 Distribution of Features

[11]:
```
from matplotlib import cm
fig = plt.figure(figsize=(25,15)) ## Plot Size
st = fig.suptitle("Distribution of Features", fontsize=50,
                  verticalalignment='center') # Plot Main Title

for col,num in zip(df.toPandas().describe().columns, range(1,11)):
    ax = fig.add_subplot(3,4,num)
    ax.hist(df.toPandas()[col])
    plt.style.use('dark_background')
    plt.grid(False)
    plt.xticks(rotation=45,fontsize=20)
    plt.yticks(fontsize=15)
    plt.title(col.upper(),fontsize=20)
plt.tight_layout()
st.set_y(0.95)
fig.subplots_adjust(top=0.85,hspace = 0.4)
plt.show()
```



1.7 Check For Null Values

[12]:
```
from pyspark.sql.functions import isnan, when, count, col
df.select([count(when(isnan(c), c)).alias(c) for c in df.columns]).toPandas().
  ↪head()
```

[12]:
```
   Pregnancies  Glucose  BloodPressure  SkinThickness  Insulin  BMI  \
0            0        5             35            227      374   11

   DiabetesPedigreeFunction  Age  Outcome
0                         0    0        0
```

1.8 WHAT IS THIS STEP DOING?

Changing diabeties from 1 and 0 to yes and no

[13]:
```
from pyspark.sql.functions import udf
y_udf = udf(lambda y: "No" if y==0 else "yes", StringType())

df=df.withColumn("HasDiabities", y_udf('OutCome')).drop("OutCome")
```

[14]:
```
df.show()
```

```
+-----------+-------+-------------+-------------+-------+----+-----------------
------+---+------------+
|Pregnancies|Glucose|BloodPressure|SkinThickness|Insulin|
```

```
BMI|DiabetesPedigreeFunction|Age|HasDiabities|
+-----------+-------+-------------+-------------+-------+----+-----------------
------+---+------------+
|          6|  148.0|         72.0|         35.0|    NaN|33.6|
0.627| 50|         yes|
|          1|   85.0|         66.0|         29.0|    NaN|26.6|
0.351| 31|          No|
|          8|  183.0|         64.0|          NaN|    NaN|23.3|
0.672| 32|         yes|
|          1|   89.0|         66.0|         23.0|   94.0|28.1|
0.167| 21|          No|
|          0|  137.0|         40.0|         35.0|  168.0|43.1|
2.288| 33|         yes|
|          5|  116.0|         74.0|          NaN|    NaN|25.6|
0.201| 30|          No|
|          3|   78.0|         50.0|         32.0|   88.0|31.0|
0.248| 26|         yes|
|         10|  115.0|          NaN|          NaN|    NaN|35.3|
0.134| 29|          No|
|          2|  197.0|         70.0|         45.0|  543.0|30.5|
0.158| 53|         yes|
|          8|  125.0|         96.0|          NaN|    NaN| NaN|
0.232| 54|         yes|
|          4|  110.0|         92.0|          NaN|    NaN|37.6|
0.191| 30|          No|
|         10|  168.0|         74.0|          NaN|    NaN|38.0|
0.537| 34|         yes|
|         10|  139.0|         80.0|          NaN|    NaN|27.1|
1.441| 57|          No|
|          1|  189.0|         60.0|         23.0|  846.0|30.1|
0.398| 59|         yes|
|          5|  166.0|         72.0|         19.0|  175.0|25.8|
0.587| 51|         yes|
|          7|  100.0|          NaN|          NaN|    NaN|30.0|
0.484| 32|         yes|
|          0|  118.0|         84.0|         47.0|  230.0|45.8|
0.551| 31|         yes|
|          7|  107.0|         74.0|          NaN|    NaN|29.6|
0.254| 31|         yes|
|          1|  103.0|         30.0|         38.0|   83.0|43.3|
0.183| 33|          No|
|          1|  115.0|         70.0|         30.0|   96.0|34.6|
0.529| 32|         yes|
+-----------+-------+-------------+-------------+-------+----+-----------------
------+---+------------+
only showing top 20 rows
```

1.8.1 WHAT IS THIS STEP DOING?

Categorizing by age groups instead of specific age

[15]:
```
def udf_multiple(age):
      if (age <= 25):
        return 'Under 25'
      elif (age >= 25 and age <= 35):
        return 'Between 25 and 35'
      elif (age > 35 and age < 50):
        return 'Between 36 and 49'
      elif (age >= 50):
        return 'Over 50'
      else: return 'N/A'

education_udf = udf(udf_multiple)
df=df.withColumn("Age_udf", education_udf('Age'))
```

[16]:
```
df.show()
```

```
+-----------+-------+-------------+-------------+-------+----+------------------
------+---+------------+-----------------+
|Pregnancies|Glucose|BloodPressure|SkinThickness|Insulin|
BMI|DiabetesPedigreeFunction|Age|HasDiabities|          Age_udf|
+-----------+-------+-------------+-------------+-------+----+------------------
------+---+------------+-----------------+
|          6|  148.0|         72.0|         35.0|    NaN|33.6|
0.627| 50|         yes|          Over 50|
|          1|   85.0|         66.0|         29.0|    NaN|26.6|
0.351| 31|          No|Between 25 and 35|
|          8|  183.0|         64.0|          NaN|    NaN|23.3|
0.672| 32|         yes|Between 25 and 35|
|          1|   89.0|         66.0|         23.0|   94.0|28.1|
0.167| 21|          No|         Under 25|
|          0|  137.0|         40.0|         35.0|  168.0|43.1|
2.288| 33|         yes|Between 25 and 35|
|          5|  116.0|         74.0|          NaN|    NaN|25.6|
0.201| 30|          No|Between 25 and 35|
|          3|   78.0|         50.0|         32.0|   88.0|31.0|
0.248| 26|         yes|Between 25 and 35|
|         10|  115.0|          NaN|          NaN|    NaN|35.3|
0.134| 29|          No|Between 25 and 35|
|          2|  197.0|         70.0|         45.0|  543.0|30.5|
0.158| 53|         yes|          Over 50|
|          8|  125.0|         96.0|          NaN|    NaN| NaN|
0.232| 54|         yes|          Over 50|
|          4|  110.0|         92.0|          NaN|    NaN|37.6|
0.191| 30|          No|Between 25 and 35|
```

```
|          10|  168.0|         74.0|          NaN|    NaN|38.0|
0.537| 34|         yes|Between 25 and 35|
|          10|  139.0|         80.0|          NaN|    NaN|27.1|
1.441| 57|          No|          Over 50|
|           1|  189.0|         60.0|         23.0|  846.0|30.1|
0.398| 59|         yes|          Over 50|
|           5|  166.0|         72.0|         19.0|  175.0|25.8|
0.587| 51|         yes|          Over 50|
|           7|  100.0|          NaN|          NaN|    NaN|30.0|
0.484| 32|         yes|Between 25 and 35|
|           0|  118.0|         84.0|         47.0|  230.0|45.8|
0.551| 31|         yes|Between 25 and 35|
|           7|  107.0|         74.0|          NaN|    NaN|29.6|
0.254| 31|         yes|Between 25 and 35|
|           1|  103.0|         30.0|         38.0|   83.0|43.3|
0.183| 33|          No|Between 25 and 35|
|           1|  115.0|         70.0|         30.0|   96.0|34.6|
0.529| 32|         yes|Between 25 and 35|
+------------+-------+-------------+-------------+-------+----+------------------
------+---+------------+-----------------+
only showing top 20 rows
```

1.8.2 WHAT IS THIS STEP DOING?

Agreggating age groups with glucose levels and well as showing max,mins averages and percentages

[17]:
```
from pyspark.sql import functions as F
from pyspark.sql.functions import rank,sum,col
from pyspark.sql import Window

window = Window.rowsBetween(Window.unboundedPreceding,Window.unboundedFollowing)
tab = df.select(['Age_udf','Glucose']).\
   groupBy('Age_udf').\
       agg(F.count('Glucose').alias('UserCount'),
           F.mean('Glucose').alias('Glucose_AVG'),
           F.min('Glucose').alias('Glucose_MIN'),
           F.max('Glucose').alias('Glucose_MAX')).\
       withColumn('total',sum(col('UserCount')).over(window)).\
       withColumn('Percent',col('UserCount')*100/col('total')).\
       drop(col('total')).sort(desc("Percent"))
```

[18]:
```
tab.show()
```

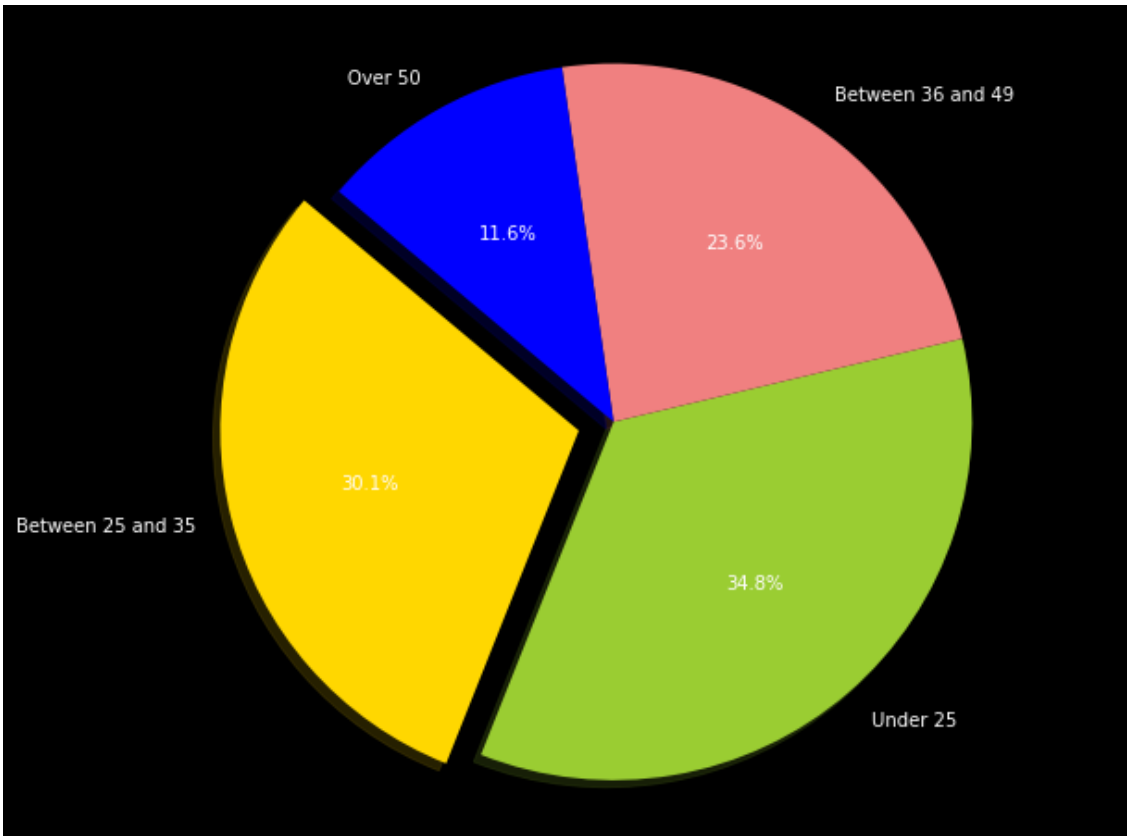
```
+-----------------+---------+-----------------+-----------+-----------+--------
----------+
|          Age_udf|UserCount|       Glucose_AVG|Glucose_MIN|Glucose_MAX|
Percent|
```

```
+----------------+---------+-----------------+-----------+-----------+--------
----------+
|         Under 25|      267|              NaN|       56.0|        NaN|
34.765625|
|Between 25 and 35|      231|121.67099567099567|       71.0|      198.0|
30.078125|
|Between 36 and 49|      181|              NaN|       44.0|
NaN|23.567708333333332|
|          Over 50|       89| 139.5505617977528|       57.0|
197.0|11.588541666666666|
+----------------+---------+-----------------+-----------+-----------+--------
----------+
```

[19]:
```
# Data to plot
labels = list(tab.select('Age_udf').distinct().toPandas()['Age_udf'])
sizes =  list(tab.select('Percent').distinct().toPandas()['Percent'])
colors = ['gold', 'yellowgreen', 'lightcoral','blue',␣
 ↪'lightskyblue','green','red']
explode = (0.1, 0.0, 0 ,0.0 )  # explode 1st slice

# Plot
plt.figure(figsize=(10,8))
plt.pie(sizes, explode=explode, labels=labels, colors=colors,
        autopct='%1.1f%%', shadow=True, startangle=140)

plt.axis('equal')
plt.show()
```



2 WHAT IS THIS STEP DOING? How do we analyze the results and what are the conclusions from them?

Showing how age groups are represented in the dataset by percentages.

[20]:
```
numeric_features = [t[0] for t in df.dtypes if t[1] != 'string']
numeric_features_df=df.select(numeric_features)
numeric_features_df.toPandas().head()
```

[20]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	\
0	6	148.0	72.0	35.0	NaN	33.6	
1	1	85.0	66.0	29.0	NaN	26.6	
2	8	183.0	64.0	NaN	NaN	23.3	
3	1	89.0	66.0	23.0	94.0	28.1	
4	0	137.0	40.0	35.0	168.0	43.1	

	DiabetesPedigreeFunction	Age
0	0.627	50
1	0.351	31
2	0.672	32

```
3                    0.167   21
4                    2.288   33
```

[21]:
```
col_names =numeric_features_df.columns
features = numeric_features_df.rdd.map(lambda row: row[0:])
corr_mat=Statistics.corr(features, method="pearson")
corr_df = pd.DataFrame(corr_mat)
corr_df.index, corr_df.columns = col_names, col_names

corr_df
```

[21]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age
Pregnancies	1.000000	NaN	NaN	NaN	NaN	NaN	-0.033523	0.544341
Glucose	NaN	1.0	NaN	NaN	NaN	NaN	NaN	NaN
BloodPressure	NaN	NaN	1.0	NaN	NaN	NaN	NaN	NaN
SkinThickness	NaN	NaN	NaN	1.0	NaN	NaN	NaN	NaN
Insulin	NaN	NaN	NaN	NaN	1.0	NaN	NaN	NaN
BMI	NaN	NaN	NaN	NaN	NaN	1.0	NaN	NaN
DiabetesPedigreeFunction	-0.033523	NaN	NaN	NaN	NaN	NaN	1.000000	0.033561
Age	0.544341	NaN	NaN	NaN	NaN	NaN	0.033561	1.000000

3 Drop Age

[22]:
```
df=df.drop("Age")
```

[23]:
```
df.show(4)
```

```
+-----------+-------+-------------+-------------+-------+----+------------------
------+------------+-----------------+
|Pregnancies|Glucose|BloodPressure|SkinThickness|Insulin|
BMI|DiabetesPedigreeFunction|HasDiabities|          Age_udf|
+-----------+-------+-------------+-------------+-------+----+------------------
------+------------+-----------------+
|          6|  148.0|         72.0|         35.0|    NaN|33.6|
0.627|         yes|          Over 50|
|          1|   85.0|         66.0|         29.0|    NaN|26.6|
```

```
0.351|          No|Between 25 and 35|
|          8|  183.0|         64.0|          NaN|    NaN|23.3|
0.672|         yes|Between 25 and 35|
|          1|   89.0|         66.0|         23.0|   94.0|28.1|
0.167|          No|         Under 25|
+-----------+-------+-------------+-------------+-------+----+-----------------
------+------------+-----------------+
only showing top 4 rows
```

4 Prepare Data for Machine Learning

4.1 a) WHAT IS THIS STEP DOING?

creating clones of df and assigning indexes to age groups

[24]:
```
df2=df
df3=df
```

[25]:
```
stringIndexer = StringIndexer()\
                  .setInputCol ("Age_udf")\
                  .setOutputCol ("Age_udfIndex")

Age_udfIndex_model=stringIndexer.fit(df2)
Age_udfIndex_df=Age_udfIndex_model.transform(df2)
Age_udfIndex_df.toPandas()
```

[25]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	\
0	6	148.0	72.0	35.0	NaN	33.6	
1	1	85.0	66.0	29.0	NaN	26.6	
2	8	183.0	64.0	NaN	NaN	23.3	
3	1	89.0	66.0	23.0	94.0	28.1	
4	0	137.0	40.0	35.0	168.0	43.1	
..	…	…	…	…	…	…	
763	10	101.0	76.0	48.0	180.0	32.9	
764	2	122.0	70.0	27.0	NaN	36.8	
765	5	121.0	72.0	23.0	112.0	26.2	
766	1	126.0	60.0	NaN	NaN	30.1	
767	1	93.0	70.0	31.0	NaN	30.4	

	DiabetesPedigreeFunction	HasDiabities	Age_udf	Age_udfIndex
0	0.627	yes	Over 50	3.0
1	0.351	No	Between 25 and 35	1.0
2	0.672	yes	Between 25 and 35	1.0
3	0.167	No	Under 25	0.0
4	2.288	yes	Between 25 and 35	1.0
..	…	…	…	…
763	0.171	No	Over 50	3.0

```
764                    0.340           No  Between 25 and 35           1.0
765                    0.245           No  Between 25 and 35           1.0
766                    0.349          yes  Between 36 and 49           2.0
767                    0.315           No           Under 25           0.0

[768 rows x 10 columns]
```

4.2 b) WHAT IS THIS STEP DOING? What is a OneHotEncoder

I dont know

[26]:
```
encoder = OneHotEncoder()\
          .setInputCols (["Age_udfIndex"])\
          .setOutputCols (["Age_encoded"])

encoder_model=encoder.fit(Age_udfIndex_df)
encoder_df=encoder_model.transform(Age_udfIndex_df)

encoder_df.toPandas().head()
```

[26]:
```
   Pregnancies  Glucose  BloodPressure  SkinThickness  Insulin   BMI  \
0            6    148.0           72.0           35.0      NaN  33.6
1            1     85.0           66.0           29.0      NaN  26.6
2            8    183.0           64.0            NaN      NaN  23.3
3            1     89.0           66.0           23.0     94.0  28.1
4            0    137.0           40.0           35.0    168.0  43.1

   DiabetesPedigreeFunction HasDiabities            Age_udf  Age_udfIndex  \
0                     0.627          yes            Over 50           3.0
1                     0.351           No  Between 25 and 35           1.0
2                     0.672          yes  Between 25 and 35           1.0
3                     0.167           No           Under 25           0.0
4                     2.288          yes  Between 25 and 35           1.0

       Age_encoded
0  (0.0, 0.0, 0.0)
1  (0.0, 1.0, 0.0)
2  (0.0, 1.0, 0.0)
3  (1.0, 0.0, 0.0)
4  (0.0, 1.0, 0.0)
```

4.3 c) WHAT IS THIS STEP DOING? What is a VectorAssembler

i dont know its not working

[27]:
```
import pandas as pd
pd.set_option('display.max_colwidth', 80)
pd.set_option('max_columns', 12)
```

```
---------------------------------------------------------------------------
OptionError                               Traceback (most recent call last)
Input In [28], in <cell line: 3>()
      1 import pandas as pd
      2 pd.set_option('display.max_colwidth', 80)
----> 3 pd.set_option('max_columns', 13)

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pandas\_co fig\config.
 ↪py:256, in CallableDynamicDoc.__call__(self, *args, **kwds)
    255 def __call__(self, *args, **kwds):
--> 256     return self.__func__(*args, **kwds)

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pandas\_co fig\config.
 ↪py:149, in _set_option(*args, **kwargs)
    146     raise TypeError(f'_set_option() got an unexpected keyword argument␣
 ↪"{kwarg}"')
    148 for k, v in zip(args[::2], args[1::2]):
--> 149     key = _get_single_key(k, silent)
    151     o = _get_registered_option(key)
    152     if o and o.validator:

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pandas\_co fig\config.
 ↪py:116, in _get_single_key(pat, silent)
    114     raise OptionError(f"No such keys(s): {repr(pat)}")
    115 if len(keys) > 1:
--> 116     raise OptionError("Pattern matched multiple keys")
    117 key = keys[0]
    119 if not silent:

OptionError: Pattern matched multiple keys
```

[29]:
```
assembler = VectorAssembler()\
        .setInputCols (["Age_encoded","Pregnancies","Glucose",
                        "BloodPressure","SkinThickness",\
                        "Insulin","BMI","DiabetesPedigreeFunction"])\
        .setOutputCol ("vectorized_features")


assembler_df=assembler.transform(encoder_df)
assembler_df.toPandas().head()
```

```
---------------------------------------------------------------------------
Py4JJavaError                             Traceback (most recent call last)
Input In [29], in <cell line: 9>()
```

```
      1 assembler = VectorAssembler()\
      2           .setInputCols (["Age_encoded","Pregnancies","Glucose",
      3                           "BloodPressure","SkinThickness",\
      4                           "Insulin","BMI","DiabetesPedigreeFunction"])\
      5           .setOutputCol ("vectorized_features")
      8 assembler_df=assembler.transform(encoder_df)
----> 9 assembler_df.toPandas().head()

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pyspark\sq.\pandas\conve
 ↪py:141, in PandasConversionMixin.toPandas(self)
    138             raise
    140 # Below is toPandas without Arrow optimization.
--> 141 pdf = pd.DataFrame.from_records(self.collect(), columns=self.columns)
    142 column_counter = Counter(self.columns)
    144 dtype = [None] * len(self.schema)

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pyspark\sq.\dataframe.
 ↪py:677, in DataFrame.collect(self)
    667 """Returns all the records as a list of :class:`Row`.
    668
    669 .. versionadded:: 1.3.0
   (…)
    674 [Row(age=2, name='Alice'), Row(age=5, name='Bob')]
    675 """
    676 with SCCallSiteSync(self._sc) as css:
--> 677     sock_info = self._jdf.collectToPython()
    678 return list(_load_from_socket(sock_info,␣
 ↪BatchedSerializer(PickleSerializer())))

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\py4j\java_gateway.
 ↪py:1304, in JavaMember.__call__(self, *args)
   1298 command = proto.CALL_COMMAND_NAME +\
   1299     self.command_header +\
   1300     args_command +\
   1301     proto.END_COMMAND_PART
   1303 answer = self.gateway_client.send_command(command)
-> 1304 return_value = get_return_value(
   1305     answer, self.gateway_client, self.target_id, self.name)
   1307 for temp_arg in temp_args:
   1308     temp_arg._detach()

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pyspark\sq.\utils.
 ↪py:111, in capture_sql_exception.<locals>.deco(*a, **kw)
    109 def deco(*a, **kw):
    110     try:
```

```
--> 111         return f(*a, **kw)
    112     except py4j.protocol.Py4JJavaError as e:
    113         converted = convert_exception(e.java_exception)

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\py4j\proto ol.
 ↪py:326, in get_return_value(answer, gateway_client, target_id, name)
    324 value = OUTPUT_CONVERTER[type](answer[2:], gateway_client)
    325 if answer[1] == REFERENCE_TYPE:
--> 326     raise Py4JJavaError(
    327         "An error occurred while calling {0}{1}{2}.\n".
    328         format(target_id, ".", name), value)
    329 else:
    330     raise Py4JError(
    331         "An error occurred while calling {0}{1}{2}. Trace:\n{3}\n".
    332         format(target_id, ".", name, value))

Py4JJavaError: An error occurred while calling o479.collectToPython.
: org.apache.spark.SparkException: Job aborted due to stage failure: Task 0 in␣
 ↪stage 55.0 failed 1 times, most recent failure: Lost task 0.0 in stage 55.0␣
 ↪(TID 856) (DESKTOP-15DALSG.mshome.net executor driver): org.apache.spark.
 ↪SparkException: Failed to execute user defined␣
 ↪function(VectorAssembler$$Lambda$3659/168973317: (struct<Age_encoded:
 ↪struct<type:tinyint,size:int,indices:array<int>,values:
 ↪array<double>>,Pregnancies_double_VectorAssembler_53ccd1a89127:double,Glucose
 ↪double,BloodPressure:double,SkinThickness:double,Insulin:double,BMI:
 ↪double,DiabetesPedigreeFunction:double>) => struct<type:tinyint,size:
 ↪int,indices:array<int>,values:array<double>>)

        at org.apache.spark.sql.catalyst.expressions.
 ↪GeneratedClass$GeneratedIteratorForCodegenStage1.processNext(Unknown Source)

        at org.apache.spark.sql.execution.BufferedRowIterator.
 ↪hasNext(BufferedRowIterator.java:43)

        at org.apache.spark.sql.execution.WholeStageCodegenExec$$anon$1.
 ↪hasNext(WholeStageCodegenExec.scala:755)

        at org.apache.spark.sql.execution.SparkPlan.
 ↪$anonfun$getByteArrayRdd$1(SparkPlan.scala:345)

        at org.apache.spark.rdd.RDD.$anonfun$mapPartitionsInternal$2(RDD.scala:
 ↪898)

        at org.apache.spark.rdd.RDD.$anonfun$mapPartitionsInternal$2$adapted(RDD.
 ↪scala:898)

        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitionsRDD.scala
 ↪52)
```

```
        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scala:373)

        at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)

        at org.apache.spark.scheduler.ResultTask.runTask(ResultTask.scala:90)

        at org.apache.spark.scheduler.Task.run(Task.scala:131)

        at org.apache.spark.executor.Executor$TaskRunner.$anonfun$run$3(Executo
↪scala:498)

        at org.apache.spark.util.Utils$.tryWithSafeFinally(Utils.scala:1439)

        at org.apache.spark.executor.Executor$TaskRunner.run(Executor.scala:501

        at java.util.concurrent.ThreadPoolExecutor.runWorker(Unknown Source)

        at java.util.concurrent.ThreadPoolExecutor$Worker.run(Unknown Source)

        at java.lang.Thread.run(Unknown Source)

Caused by: org.apache.spark.SparkException: Encountered NaN while assembling a␣
↪row with handleInvalid = "error". Consider
removing NaNs from dataset or using handleInvalid = "keep" or "skip".

        at org.apache.spark.ml.feature.VectorAssembler$.
↪$anonfun$assemble$1(VectorAssembler.scala:264)

        at org.apache.spark.ml.feature.VectorAssembler$.
↪$anonfun$assemble$1$adapted(VectorAssembler.scala:260)

        at scala.collection.IndexedSeqOptimized.foreach(IndexedSeqOptimized.
↪scala:36)

        at scala.collection.IndexedSeqOptimized.foreach$(IndexedSeqOptimized.
↪scala:33)

        at scala.collection.mutable.WrappedArray.foreach(WrappedArray.scala:38)

        at org.apache.spark.ml.feature.VectorAssembler$.assemble(VectorAssembler
↪scala:260)

        at org.apache.spark.ml.feature.VectorAssembler.
↪$anonfun$transform$6(VectorAssembler.scala:143)

        … 17 more
```

```
Driver stacktrace:

        at org.apache.spark.scheduler.DAGScheduler.
↪failJobAndIndependentStages(DAGScheduler.scala:2303)

        at org.apache.spark.scheduler.DAGScheduler.
↪$anonfun$abortStage$2(DAGScheduler.scala:2252)

        at org.apache.spark.scheduler.DAGScheduler.
↪$anonfun$abortStage$2$adapted(DAGScheduler.scala:2251)

        at scala.collection.mutable.ResizableArray.foreach(ResizableArray.scala
↪62)

        at scala.collection.mutable.ResizableArray.foreach$(ResizableArray.scal :
↪55)

        at scala.collection.mutable.ArrayBuffer.foreach(ArrayBuffer.scala:49)

        at org.apache.spark.scheduler.DAGScheduler.abortStage(DAGScheduler.scal :
↪2251)

        at org.apache.spark.scheduler.DAGScheduler.
↪$anonfun$handleTaskSetFailed$1(DAGScheduler.scala:1124)

        at org.apache.spark.scheduler.DAGScheduler.
↪$anonfun$handleTaskSetFailed$1$adapted(DAGScheduler.scala:1124)

        at scala.Option.foreach(Option.scala:407)

        at org.apache.spark.scheduler.DAGScheduler.
↪handleTaskSetFailed(DAGScheduler.scala:1124)

        at org.apache.spark.scheduler.DAGSchedulerEventProcessLoop.
↪doOnReceive(DAGScheduler.scala:2490)

        at org.apache.spark.scheduler.DAGSchedulerEventProcessLoop.
↪onReceive(DAGScheduler.scala:2432)

        at org.apache.spark.scheduler.DAGSchedulerEventProcessLoop.
↪onReceive(DAGScheduler.scala:2421)

        at org.apache.spark.util.EventLoop$$anon$1.run(EventLoop.scala:49)

        at org.apache.spark.scheduler.DAGScheduler.runJob(DAGScheduler.scala:90 )

        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2196)
```

```
        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2217)

        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2236)

        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2261)

        at org.apache.spark.rdd.RDD.$anonfun$collect$1(RDD.scala:1030)

        at org.apache.spark.rdd.RDDOperationScope$.withScope(RDDOperationScope.
↪scala:151)

        at org.apache.spark.rdd.RDDOperationScope$.withScope(RDDOperationScope.
↪scala:112)

        at org.apache.spark.rdd.RDD.withScope(RDD.scala:414)

        at org.apache.spark.rdd.RDD.collect(RDD.scala:1029)

        at org.apache.spark.sql.execution.SparkPlan.executeCollect(SparkPlan.
↪scala:390)

        at org.apache.spark.sql.Dataset.$anonfun$collectToPython$1(Dataset.scala:
↪3532)

        at org.apache.spark.sql.Dataset.$anonfun$withAction$1(Dataset.scala:370)

        at org.apache.spark.sql.execution.SQLExecution$.
↪$anonfun$withNewExecutionId$5(SQLExecution.scala:103)

        at org.apache.spark.sql.execution.SQLExecution$.
↪withSQLConfPropagated(SQLExecution.scala:163)

        at org.apache.spark.sql.execution.SQLExecution$.
↪$anonfun$withNewExecutionId$1(SQLExecution.scala:90)

        at org.apache.spark.sql.SparkSession.withActive(SparkSession.scala:775)

        at org.apache.spark.sql.execution.SQLExecution$.
↪withNewExecutionId(SQLExecution.scala:64)

        at org.apache.spark.sql.Dataset.withAction(Dataset.scala:3698)

        at org.apache.spark.sql.Dataset.collectToPython(Dataset.scala:3529)

        at sun.reflect.GeneratedMethodAccessor125.invoke(Unknown Source)

        at sun.reflect.DelegatingMethodAccessorImpl.invoke(Unknown Source)
```

```
        at java.lang.reflect.Method.invoke(Unknown Source)

        at py4j.reflection.MethodInvoker.invoke(MethodInvoker.java:244)

        at py4j.reflection.ReflectionEngine.invoke(ReflectionEngine.java:357)

        at py4j.Gateway.invoke(Gateway.java:282)

        at py4j.commands.AbstractCommand.invokeMethod(AbstractCommand.java:132)

        at py4j.commands.CallCommand.execute(CallCommand.java:79)

        at py4j.GatewayConnection.run(GatewayConnection.java:238)

        at java.lang.Thread.run(Unknown Source)

Caused by: org.apache.spark.SparkException: Failed to execute user defined␣
 ↪function(VectorAssembler$$Lambda$3659/168973317: (struct<Age_encoded:
 ↪struct<type:tinyint,size:int,indices:array<int>,values:
 ↪array<double>>,Pregnancies_double_VectorAssembler_53ccd1a89127:double,Glucose
 ↪double,BloodPressure:double,SkinThickness:double,Insulin:double,BMI:
 ↪double,DiabetesPedigreeFunction:double>) => struct<type:tinyint,size:
 ↪int,indices:array<int>,values:array<double>>)

        at org.apache.spark.sql.catalyst.expressions.
 ↪GeneratedClass$GeneratedIteratorForCodegenStage1.processNext(Unknown Source)

        at org.apache.spark.sql.execution.BufferedRowIterator.
 ↪hasNext(BufferedRowIterator.java:43)

        at org.apache.spark.sql.execution.WholeStageCodegenExec$$anon$1.
 ↪hasNext(WholeStageCodegenExec.scala:755)

        at org.apache.spark.sql.execution.SparkPlan.
 ↪$anonfun$getByteArrayRdd$1(SparkPlan.scala:345)

        at org.apache.spark.rdd.RDD.$anonfun$mapPartitionsInternal$2(RDD.scala:
 ↪898)

        at org.apache.spark.rdd.RDD.$anonfun$mapPartitionsInternal$2$adapted(RDD.
 ↪scala:898)

        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitionsRDD.scala
 ↪52)

        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scala:373)

        at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)
```

```
        at org.apache.spark.scheduler.ResultTask.runTask(ResultTask.scala:90)

        at org.apache.spark.scheduler.Task.run(Task.scala:131)

        at org.apache.spark.executor.Executor$TaskRunner.$anonfun$run$3(Executo
↪scala:498)

        at org.apache.spark.util.Utils$.tryWithSafeFinally(Utils.scala:1439)

        at org.apache.spark.executor.Executor$TaskRunner.run(Executor.scala:501

        at java.util.concurrent.ThreadPoolExecutor.runWorker(Unknown Source)

        at java.util.concurrent.ThreadPoolExecutor$Worker.run(Unknown Source)

        … 1 more

Caused by: org.apache.spark.SparkException: Encountered NaN while assembling a␣
↪row with handleInvalid = "error". Consider
removing NaNs from dataset or using handleInvalid = "keep" or "skip".

        at org.apache.spark.ml.feature.VectorAssembler$.
↪$anonfun$assemble$1(VectorAssembler.scala:264)

        at org.apache.spark.ml.feature.VectorAssembler$.
↪$anonfun$assemble$1$adapted(VectorAssembler.scala:260)

        at scala.collection.IndexedSeqOptimized.foreach(IndexedSeqOptimized.
↪scala:36)

        at scala.collection.IndexedSeqOptimized.foreach$(IndexedSeqOptimized.
↪scala:33)

        at scala.collection.mutable.WrappedArray.foreach(WrappedArray.scala:38)

        at org.apache.spark.ml.feature.VectorAssembler$.assemble(VectorAssembler
↪scala:260)

        at org.apache.spark.ml.feature.VectorAssembler.
↪$anonfun$transform$6(VectorAssembler.scala:143)

        … 17 more
```

4.4 d) WHAT IS THIS STEP DOING? What is a LabelIndexer

assigns indexes to labels

[30]:
```
label_indexer = StringIndexer()\
         .setInputCol ("HasDiabities")\
         .setOutputCol ("label")

label_indexer_model=label_indexer.fit(assembler_df)
label_indexer_df=label_indexer_model.transform(assembler_df)

label_indexer_df.select("HasDiabities","label").toPandas().head()
```

[30]:

	HasDiabities	label
0	yes	1.0
1	No	0.0
2	yes	1.0
3	No	0.0
4	yes	1.0

4.5 e) WHAT IS THIS STEP DOING?

Errors

[32]:
```
scaler = StandardScaler()\
         .setInputCol ("vectorized_features")\
         .setOutputCol ("features")

scaler_model=scaler.fit(label_indexer_df)
scaler_df=scaler_model.transform(label_indexer_df)
pd.set_option('display.max_colwidth', 40)
scaler_df.select("vectorized_features","features").toPandas().head(5)
```

```
---------------------------------------------------------------------------
Py4JJavaError                             Traceback (most recent call last)
Input In [32], in <cell line: 5>()
      1 scaler = StandardScaler()\
      2          .setInputCol ("vectorized_features")\
      3          .setOutputCol ("features")
----> 5 scaler_model=scaler.fit(label_indexer_df)
      6 scaler_df=scaler_model.transform(label_indexer_df)
      7 pd.set_option('display.max_colwidth', 40)

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pyspark\ml\base.
 ↪py:161, in Estimator.fit(self, dataset, params)
    159         return self.copy(params)._fit(dataset)
    160     else:
--> 161         return self._fit(dataset)
```

```
    162 else:
    163     raise ValueError("Params must be either a param map or a list/tuple
 ↪of param maps, "
    164                      "but got %s." % type(params))

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pyspark\ml\wrapper.
 ↪py:335, in JavaEstimator._fit(self, dataset)
    334 def _fit(self, dataset):
--> 335     java_model = self._fit_java(dataset)
    336     model = self._create_model(java_model)
    337     return self._copyValues(model)

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pyspark\ml\wrapper.
 ↪py:332, in JavaEstimator._fit_java(self, dataset)
    318 """
    319 Fits a Java model to the input dataset.
    320
   (…)
    329     fitted Java model
    330 """
    331 self._transfer_params_to_java()
--> 332 return self._java_obj.fit(dataset._jdf)

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\py4j\java_gateway.
 ↪py:1304, in JavaMember.__call__(self, *args)
   1298 command = proto.CALL_COMMAND_NAME +\
   1299     self.command_header +\
   1300     args_command +\
   1301     proto.END_COMMAND_PART
   1303 answer = self.gateway_client.send_command(command)
-> 1304 return_value = get_return_value(
   1305     answer, self.gateway_client, self.target_id, self.name)
   1307 for temp_arg in temp_args:
   1308     temp_arg._detach()

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pyspark\sql\utils.
 ↪py:111, in capture_sql_exception.<locals>.deco(*a, **kw)
    109 def deco(*a, **kw):
    110     try:
--> 111         return f(*a, **kw)
    112     except py4j.protocol.Py4JJavaError as e:
    113         converted = convert_exception(e.java_exception)
```

```
File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\py4j\protocol.
 ↪py:326, in get_return_value(answer, gateway_client, target_id, name)
    324 value = OUTPUT_CONVERTER[type](answer[2:], gateway_client)
    325 if answer[1] == REFERENCE_TYPE:
--> 326     raise Py4JJavaError(
    327         "An error occurred while calling {0}{1}{2}.\n".
    328         format(target_id, ".", name), value)
    329 else:
    330     raise Py4JError(
    331         "An error occurred while calling {0}{1}{2}. Trace:\n{3}\n".
    332         format(target_id, ".", name, value))

Py4JJavaError: An error occurred while calling o557.fit.
: org.apache.spark.SparkException: Job aborted due to stage failure: Task 0 in␣
 ↪stage 61.0 failed 1 times, most recent failure: Lost task 0.0 in stage 61.0␣
 ↪(TID 861) (DESKTOP-15DALSG.mshome.net executor driver): org.apache.spark.
 ↪SparkException: Failed to execute user defined␣
 ↪function(VectorAssembler$$Lambda$3659/168973317: (struct<Age_encoded:
 ↪struct<type:tinyint,size:int,indices:array<int>,values:
 ↪array<double>>,Pregnancies_double_VectorAssembler_53ccd1a89127:double,Glucose
 ↪double,BloodPressure:double,SkinThickness:double,Insulin:double,BMI:
 ↪double,DiabetesPedigreeFunction:double>) => struct<type:tinyint,size:
 ↪int,indices:array<int>,values:array<double>>)

        at org.apache.spark.sql.catalyst.expressions.
 ↪GeneratedClass$GeneratedIteratorForCodegenStage1.processNext(Unknown Source)

        at org.apache.spark.sql.execution.BufferedRowIterator.
 ↪hasNext(BufferedRowIterator.java:43)

        at org.apache.spark.sql.execution.WholeStageCodegenExec$$anon$1.
 ↪hasNext(WholeStageCodegenExec.scala:755)

        at org.apache.spark.sql.execution.aggregate.ObjectHashAggregateExec.
 ↪$anonfun$doExecute$2(ObjectHashAggregateExec.scala:87)

        at org.apache.spark.sql.execution.aggregate.ObjectHashAggregateExec.
 ↪$anonfun$doExecute$2$adapted(ObjectHashAggregateExec.scala:85)

        at org.apache.spark.rdd.RDD.
 ↪$anonfun$mapPartitionsWithIndexInternal$2(RDD.scala:885)

        at org.apache.spark.rdd.RDD.
 ↪$anonfun$mapPartitionsWithIndexInternal$2$adapted(RDD.scala:885)

        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitionsRDD.scala
 ↪52)

        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scala:373)
```

```
        at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)

        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitionsRDD.scala
↪52)

        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scala:373)

        at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)

        at org.apache.spark.shuffle.ShuffleWriteProcessor.
↪write(ShuffleWriteProcessor.scala:59)

        at org.apache.spark.scheduler.ShuffleMapTask.runTask(ShuffleMapTask.
↪scala:99)

        at org.apache.spark.scheduler.ShuffleMapTask.runTask(ShuffleMapTask.
↪scala:52)

        at org.apache.spark.scheduler.Task.run(Task.scala:131)

        at org.apache.spark.executor.Executor$TaskRunner.$anonfun$run$3(Executo .
↪scala:498)

        at org.apache.spark.util.Utils$.tryWithSafeFinally(Utils.scala:1439)

        at org.apache.spark.executor.Executor$TaskRunner.run(Executor.scala:501

        at java.util.concurrent.ThreadPoolExecutor.runWorker(Unknown Source)

        at java.util.concurrent.ThreadPoolExecutor$Worker.run(Unknown Source)

        at java.lang.Thread.run(Unknown Source)

Caused by: org.apache.spark.SparkException: Encountered NaN while assembling a␣
↪row with handleInvalid = "error". Consider
removing NaNs from dataset or using handleInvalid = "keep" or "skip".

        at org.apache.spark.ml.feature.VectorAssembler$.
↪$anonfun$assemble$1(VectorAssembler.scala:264)

        at org.apache.spark.ml.feature.VectorAssembler$.
↪$anonfun$assemble$1$adapted(VectorAssembler.scala:260)

        at scala.collection.IndexedSeqOptimized.foreach(IndexedSeqOptimized.
↪scala:36)
```

```
        at scala.collection.IndexedSeqOptimized.foreach$(IndexedSeqOptimized.
↪scala:33)

        at scala.collection.mutable.WrappedArray.foreach(WrappedArray.scala:38)

        at org.apache.spark.ml.feature.VectorAssembler$.assemble(VectorAssembler.
↪scala:260)

        at org.apache.spark.ml.feature.VectorAssembler.
↪$anonfun$transform$6(VectorAssembler.scala:143)

        … 23 more


Driver stacktrace:

        at org.apache.spark.scheduler.DAGScheduler.
↪failJobAndIndependentStages(DAGScheduler.scala:2303)

        at org.apache.spark.scheduler.DAGScheduler.
↪$anonfun$abortStage$2(DAGScheduler.scala:2252)

        at org.apache.spark.scheduler.DAGScheduler.
↪$anonfun$abortStage$2$adapted(DAGScheduler.scala:2251)

        at scala.collection.mutable.ResizableArray.foreach(ResizableArray.scala
↪62)

        at scala.collection.mutable.ResizableArray.foreach$(ResizableArray.scala:
↪55)

        at scala.collection.mutable.ArrayBuffer.foreach(ArrayBuffer.scala:49)

        at org.apache.spark.scheduler.DAGScheduler.abortStage(DAGScheduler.scala:
↪2251)

        at org.apache.spark.scheduler.DAGScheduler.
↪$anonfun$handleTaskSetFailed$1(DAGScheduler.scala:1124)

        at org.apache.spark.scheduler.DAGScheduler.
↪$anonfun$handleTaskSetFailed$1$adapted(DAGScheduler.scala:1124)

        at scala.Option.foreach(Option.scala:407)

        at org.apache.spark.scheduler.DAGScheduler.
↪handleTaskSetFailed(DAGScheduler.scala:1124)
```

```
        at org.apache.spark.scheduler.DAGSchedulerEventProcessLoop.
↪doOnReceive(DAGScheduler.scala:2490)

        at org.apache.spark.scheduler.DAGSchedulerEventProcessLoop.
↪onReceive(DAGScheduler.scala:2432)

        at org.apache.spark.scheduler.DAGSchedulerEventProcessLoop.
↪onReceive(DAGScheduler.scala:2421)

        at org.apache.spark.util.EventLoop$$anon$1.run(EventLoop.scala:49)

        at org.apache.spark.scheduler.DAGScheduler.runJob(DAGScheduler.scala:90
↪)

        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2196)

        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2217)

        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2236)

        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2261)

        at org.apache.spark.rdd.RDD.$anonfun$collect$1(RDD.scala:1030)

        at org.apache.spark.rdd.RDDOperationScope$.withScope(RDDOperationScope.
↪scala:151)

        at org.apache.spark.rdd.RDDOperationScope$.withScope(RDDOperationScope.
↪scala:112)

        at org.apache.spark.rdd.RDD.withScope(RDD.scala:414)

        at org.apache.spark.rdd.RDD.collect(RDD.scala:1029)

        at org.apache.spark.sql.execution.SparkPlan.executeCollect(SparkPlan.
↪scala:390)

        at org.apache.spark.sql.Dataset.collectFromPlan(Dataset.scala:3709)

        at org.apache.spark.sql.Dataset.$anonfun$head$1(Dataset.scala:2735)

        at org.apache.spark.sql.Dataset.$anonfun$withAction$1(Dataset.scala:370
↪)

        at org.apache.spark.sql.execution.SQLExecution$.
↪$anonfun$withNewExecutionId$5(SQLExecution.scala:103)

        at org.apache.spark.sql.execution.SQLExecution$.
↪withSQLConfPropagated(SQLExecution.scala:163)
```

```
        at org.apache.spark.sql.execution.SQLExecution$.
↪$anonfun$withNewExecutionId$1(SQLExecution.scala:90)

        at org.apache.spark.sql.SparkSession.withActive(SparkSession.scala:775)

        at org.apache.spark.sql.execution.SQLExecution$.
↪withNewExecutionId(SQLExecution.scala:64)

        at org.apache.spark.sql.Dataset.withAction(Dataset.scala:3698)

        at org.apache.spark.sql.Dataset.head(Dataset.scala:2735)

        at org.apache.spark.sql.Dataset.head(Dataset.scala:2742)

        at org.apache.spark.sql.Dataset.first(Dataset.scala:2749)

        at org.apache.spark.ml.feature.StandardScaler.fit(StandardScaler.scala:
↪113)

        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)

        at sun.reflect.NativeMethodAccessorImpl.invoke(Unknown Source)

        at sun.reflect.DelegatingMethodAccessorImpl.invoke(Unknown Source)

        at java.lang.reflect.Method.invoke(Unknown Source)

        at py4j.reflection.MethodInvoker.invoke(MethodInvoker.java:244)

        at py4j.reflection.ReflectionEngine.invoke(ReflectionEngine.java:357)

        at py4j.Gateway.invoke(Gateway.java:282)

        at py4j.commands.AbstractCommand.invokeMethod(AbstractCommand.java:132)

        at py4j.commands.CallCommand.execute(CallCommand.java:79)

        at py4j.GatewayConnection.run(GatewayConnection.java:238)

        at java.lang.Thread.run(Unknown Source)

Caused by: org.apache.spark.SparkException: Failed to execute user defined␣
↪function(VectorAssembler$$Lambda$3659/168973317: (struct<Age_encoded:
↪struct<type:tinyint,size:int,indices:array<int>,values:
↪array<double>>,Pregnancies_double_VectorAssembler_53ccd1a89127:double,Glucose
↪double,BloodPressure:double,SkinThickness:double,Insulin:double,BMI:
↪double,DiabetesPedigreeFunction:double>) => struct<type:tinyint,size:
↪int,indices:array<int>,values:array<double>>)
```

```
        at org.apache.spark.sql.catalyst.expressions.
↪GeneratedClass$GeneratedIteratorForCodegenStage1.processNext(Unknown Source)

        at org.apache.spark.sql.execution.BufferedRowIterator.
↪hasNext(BufferedRowIterator.java:43)

        at org.apache.spark.sql.execution.WholeStageCodegenExec$$anon$1.
↪hasNext(WholeStageCodegenExec.scala:755)

        at org.apache.spark.sql.execution.aggregate.ObjectHashAggregateExec.
↪$anonfun$doExecute$2(ObjectHashAggregateExec.scala:87)

        at org.apache.spark.sql.execution.aggregate.ObjectHashAggregateExec.
↪$anonfun$doExecute$2$adapted(ObjectHashAggregateExec.scala:85)

        at org.apache.spark.rdd.RDD.
↪$anonfun$mapPartitionsWithIndexInternal$2(RDD.scala:885)

        at org.apache.spark.rdd.RDD.
↪$anonfun$mapPartitionsWithIndexInternal$2$adapted(RDD.scala:885)

        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitionsRDD.scala
↪52)

        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scala:373)

        at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)

        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitionsRDD.scala
↪52)

        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scala:373)

        at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)

        at org.apache.spark.shuffle.ShuffleWriteProcessor.
↪write(ShuffleWriteProcessor.scala:59)

        at org.apache.spark.scheduler.ShuffleMapTask.runTask(ShuffleMapTask.
↪scala:99)

        at org.apache.spark.scheduler.ShuffleMapTask.runTask(ShuffleMapTask.
↪scala:52)

        at org.apache.spark.scheduler.Task.run(Task.scala:131)

        at org.apache.spark.executor.Executor$TaskRunner.$anonfun$run$3(Executo
↪scala:498)
```

```
        at org.apache.spark.util.Utils$.tryWithSafeFinally(Utils.scala:1439)

        at org.apache.spark.executor.Executor$TaskRunner.run(Executor.scala:501

        at java.util.concurrent.ThreadPoolExecutor.runWorker(Unknown Source)

        at java.util.concurrent.ThreadPoolExecutor$Worker.run(Unknown Source)

        … 1 more

Caused by: org.apache.spark.SparkException: Encountered NaN while assembling a␣
 ↪row with handleInvalid = "error". Consider
removing NaNs from dataset or using handleInvalid = "keep" or "skip".

        at org.apache.spark.ml.feature.VectorAssembler$.
 ↪$anonfun$assemble$1(VectorAssembler.scala:264)

        at org.apache.spark.ml.feature.VectorAssembler$.
 ↪$anonfun$assemble$1$adapted(VectorAssembler.scala:260)

        at scala.collection.IndexedSeqOptimized.foreach(IndexedSeqOptimized.
 ↪scala:36)

        at scala.collection.IndexedSeqOptimized.foreach$(IndexedSeqOptimized.
 ↪scala:33)

        at scala.collection.mutable.WrappedArray.foreach(WrappedArray.scala:38)

        at org.apache.spark.ml.feature.VectorAssembler$.assemble(VectorAssembler.
 ↪scala:260)

        at org.apache.spark.ml.feature.VectorAssembler.
 ↪$anonfun$transform$6(VectorAssembler.scala:143)

        … 23 more
```

5 WHAT IS THIS STEP DOING? what is the pipeline?

[33]:
```
pipeline_stages=Pipeline()\
                .
 ↪setStages([stringIndexer,encoder,assembler,label_indexer,scaler])
pipeline_model=pipeline_stages.fit(df3)
pipeline_df=pipeline_model.transform(df3)
```

```
---------------------------------------------------------------------------
Py4JJavaError                             Traceback (most recent call last)
Input In [33], in <cell line: 3>()
      1 pipeline_stages=Pipeline()\
      2                 .
 ↪setStages([stringIndexer,encoder,assembler,label_indexer,scaler])
----> 3 pipeline_model=pipeline_stages.fit(df3)
      4 pipeline_df=pipeline_model.transform(df3)

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pyspark\ml\base.
 ↪py:161, in Estimator.fit(self, dataset, params)
    159         return self.copy(params)._fit(dataset)
    160     else:
--> 161         return self._fit(dataset)
    162 else:
    163     raise ValueError("Params must be either a param map or a list/tuple
 ↪of param maps, "
    164                      "but got %s." % type(params))

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pyspark\ml\pipeline.
 ↪py:114, in Pipeline._fit(self, dataset)
    112     dataset = stage.transform(dataset)
    113 else:  # must be an Estimator
--> 114     model = stage.fit(dataset)
    115     transformers.append(model)
    116     if i < indexOfLastEstimator:

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pyspark\ml\base.
 ↪py:161, in Estimator.fit(self, dataset, params)
    159         return self.copy(params)._fit(dataset)
    160     else:
--> 161         return self._fit(dataset)
    162 else:
    163     raise ValueError("Params must be either a param map or a list/tuple
 ↪of param maps, "
    164                      "but got %s." % type(params))
```

```
File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pyspark\ml\wrapper.
 ↪py:335, in JavaEstimator._fit(self, dataset)
    334 def _fit(self, dataset):
--> 335     java_model = self._fit_java(dataset)
    336     model = self._create_model(java_model)
    337     return self._copyValues(model)

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pyspark\ml\wrapper.
 ↪py:332, in JavaEstimator._fit_java(self, dataset)
    318 """
    319 Fits a Java model to the input dataset.
    320
   (…)
    329     fitted Java model
    330 """
    331 self._transfer_params_to_java()
--> 332 return self._java_obj.fit(dataset._jdf)

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\py4j\java_gateway.
 ↪py:1304, in JavaMember.__call__(self, *args)
   1298 command = proto.CALL_COMMAND_NAME +\
   1299     self.command_header +\
   1300     args_command +\
   1301     proto.END_COMMAND_PART
   1303 answer = self.gateway_client.send_command(command)
-> 1304 return_value = get_return_value(
   1305     answer, self.gateway_client, self.target_id, self.name)
   1307 for temp_arg in temp_args:
   1308     temp_arg._detach()

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pyspark\sql\utils.
 ↪py:111, in capture_sql_exception.<locals>.deco(*a, **kw)
    109 def deco(*a, **kw):
    110     try:
--> 111         return f(*a, **kw)
    112     except py4j.protocol.Py4JJavaError as e:
    113         converted = convert_exception(e.java_exception)

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\py4j\protocol.
 ↪py:326, in get_return_value(answer, gateway_client, target_id, name)
    324 value = OUTPUT_CONVERTER[type](answer[2:], gateway_client)
    325 if answer[1] == REFERENCE_TYPE:
--> 326     raise Py4JJavaError(
    327         "An error occurred while calling {0}{1}{2}.\n".
```

```
    328         format(target_id, ".", name), value)
    329 else:
    330     raise Py4JError(
    331         "An error occurred while calling {0}{1}{2}. Trace:\n{3}\n".
    332         format(target_id, ".", name, value))

Py4JJavaError: An error occurred while calling o557.fit.
: org.apache.spark.SparkException: Job aborted due to stage failure: Task 0 in␣
↪stage 67.0 failed 1 times, most recent failure: Lost task 0.0 in stage 67.0␣
↪(TID 866) (DESKTOP-15DALSG.mshome.net executor driver): org.apache.spark.
↪SparkException: Failed to execute user defined␣
↪function(VectorAssembler$$Lambda$3659/168973317: (struct<Age_encoded:
↪struct<type:tinyint,size:int,indices:array<int>,values:
↪array<double>>,Pregnancies_double_VectorAssembler_53ccd1a89127:double,Glucose
↪double,BloodPressure:double,SkinThickness:double,Insulin:double,BMI:
↪double,DiabetesPedigreeFunction:double>) => struct<type:tinyint,size:
↪int,indices:array<int>,values:array<double>>)

        at org.apache.spark.sql.catalyst.expressions.
↪GeneratedClass$GeneratedIteratorForCodegenStage1.processNext(Unknown Source)

        at org.apache.spark.sql.execution.BufferedRowIterator.
↪hasNext(BufferedRowIterator.java:43)

        at org.apache.spark.sql.execution.WholeStageCodegenExec$$anon$1.
↪hasNext(WholeStageCodegenExec.scala:755)

        at org.apache.spark.sql.execution.aggregate.ObjectHashAggregateExec.
↪$anonfun$doExecute$2(ObjectHashAggregateExec.scala:87)

        at org.apache.spark.sql.execution.aggregate.ObjectHashAggregateExec.
↪$anonfun$doExecute$2$adapted(ObjectHashAggregateExec.scala:85)

        at org.apache.spark.rdd.RDD.
↪$anonfun$mapPartitionsWithIndexInternal$2(RDD.scala:885)

        at org.apache.spark.rdd.RDD.
↪$anonfun$mapPartitionsWithIndexInternal$2$adapted(RDD.scala:885)

        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitionsRDD.scala
↪52)

        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scala:373)

        at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)

        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitionsRDD.scala
↪52)

        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scala:373)
```

```
        at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)

        at org.apache.spark.shuffle.ShuffleWriteProcessor.
↪write(ShuffleWriteProcessor.scala:59)

        at org.apache.spark.scheduler.ShuffleMapTask.runTask(ShuffleMapTask.
↪scala:99)

        at org.apache.spark.scheduler.ShuffleMapTask.runTask(ShuffleMapTask.
↪scala:52)

        at org.apache.spark.scheduler.Task.run(Task.scala:131)

        at org.apache.spark.executor.Executor$TaskRunner.$anonfun$run$3(Executo .
↪scala:498)

        at org.apache.spark.util.Utils$.tryWithSafeFinally(Utils.scala:1439)

        at org.apache.spark.executor.Executor$TaskRunner.run(Executor.scala:501

        at java.util.concurrent.ThreadPoolExecutor.runWorker(Unknown Source)

        at java.util.concurrent.ThreadPoolExecutor$Worker.run(Unknown Source)

        at java.lang.Thread.run(Unknown Source)

Caused by: org.apache.spark.SparkException: Encountered NaN while assembling a␣
↪row with handleInvalid = "error". Consider
removing NaNs from dataset or using handleInvalid = "keep" or "skip".

        at org.apache.spark.ml.feature.VectorAssembler$.
↪$anonfun$assemble$1(VectorAssembler.scala:264)

        at org.apache.spark.ml.feature.VectorAssembler$.
↪$anonfun$assemble$1$adapted(VectorAssembler.scala:260)

        at scala.collection.IndexedSeqOptimized.foreach(IndexedSeqOptimized.
↪scala:36)

        at scala.collection.IndexedSeqOptimized.foreach$(IndexedSeqOptimized.
↪scala:33)

        at scala.collection.mutable.WrappedArray.foreach(WrappedArray.scala:38)

        at org.apache.spark.ml.feature.VectorAssembler$.assemble(VectorAssembler .
↪scala:260)
```

```
        at org.apache.spark.ml.feature.VectorAssembler.
↪$anonfun$transform$6(VectorAssembler.scala:143)

        … 23 more


Driver stacktrace:

        at org.apache.spark.scheduler.DAGScheduler.
↪failJobAndIndependentStages(DAGScheduler.scala:2303)

        at org.apache.spark.scheduler.DAGScheduler.
↪$anonfun$abortStage$2(DAGScheduler.scala:2252)

        at org.apache.spark.scheduler.DAGScheduler.
↪$anonfun$abortStage$2$adapted(DAGScheduler.scala:2251)

        at scala.collection.mutable.ResizableArray.foreach(ResizableArray.scala
↪62)

        at scala.collection.mutable.ResizableArray.foreach$(ResizableArray.scal :
↪55)

        at scala.collection.mutable.ArrayBuffer.foreach(ArrayBuffer.scala:49)

        at org.apache.spark.scheduler.DAGScheduler.abortStage(DAGScheduler.scal :
↪2251)

        at org.apache.spark.scheduler.DAGScheduler.
↪$anonfun$handleTaskSetFailed$1(DAGScheduler.scala:1124)

        at org.apache.spark.scheduler.DAGScheduler.
↪$anonfun$handleTaskSetFailed$1$adapted(DAGScheduler.scala:1124)

        at scala.Option.foreach(Option.scala:407)

        at org.apache.spark.scheduler.DAGScheduler.
↪handleTaskSetFailed(DAGScheduler.scala:1124)

        at org.apache.spark.scheduler.DAGSchedulerEventProcessLoop.
↪doOnReceive(DAGScheduler.scala:2490)

        at org.apache.spark.scheduler.DAGSchedulerEventProcessLoop.
↪onReceive(DAGScheduler.scala:2432)

        at org.apache.spark.scheduler.DAGSchedulerEventProcessLoop.
↪onReceive(DAGScheduler.scala:2421)
```

```
        at org.apache.spark.util.EventLoop$$anon$1.run(EventLoop.scala:49)

        at org.apache.spark.scheduler.DAGScheduler.runJob(DAGScheduler.scala:90
        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2196)

        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2217)

        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2236)

        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2261)

        at org.apache.spark.rdd.RDD.$anonfun$collect$1(RDD.scala:1030)

        at org.apache.spark.rdd.RDDOperationScope$.withScope(RDDOperationScope.
↪scala:151)

        at org.apache.spark.rdd.RDDOperationScope$.withScope(RDDOperationScope.
↪scala:112)

        at org.apache.spark.rdd.RDD.withScope(RDD.scala:414)

        at org.apache.spark.rdd.RDD.collect(RDD.scala:1029)

        at org.apache.spark.sql.execution.SparkPlan.executeCollect(SparkPlan.
↪scala:390)

        at org.apache.spark.sql.Dataset.collectFromPlan(Dataset.scala:3709)

        at org.apache.spark.sql.Dataset.$anonfun$head$1(Dataset.scala:2735)

        at org.apache.spark.sql.Dataset.$anonfun$withAction$1(Dataset.scala:370
        at org.apache.spark.sql.execution.SQLExecution$.
↪$anonfun$withNewExecutionId$5(SQLExecution.scala:103)

        at org.apache.spark.sql.execution.SQLExecution$.
↪withSQLConfPropagated(SQLExecution.scala:163)

        at org.apache.spark.sql.execution.SQLExecution$.
↪$anonfun$withNewExecutionId$1(SQLExecution.scala:90)

        at org.apache.spark.sql.SparkSession.withActive(SparkSession.scala:775)

        at org.apache.spark.sql.execution.SQLExecution$.
↪withNewExecutionId(SQLExecution.scala:64)

        at org.apache.spark.sql.Dataset.withAction(Dataset.scala:3698)
```

```
        at org.apache.spark.sql.Dataset.head(Dataset.scala:2735)

        at org.apache.spark.sql.Dataset.head(Dataset.scala:2742)

        at org.apache.spark.sql.Dataset.first(Dataset.scala:2749)

        at org.apache.spark.ml.feature.StandardScaler.fit(StandardScaler.scala:
↪113)

        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)

        at sun.reflect.NativeMethodAccessorImpl.invoke(Unknown Source)

        at sun.reflect.DelegatingMethodAccessorImpl.invoke(Unknown Source)

        at java.lang.reflect.Method.invoke(Unknown Source)

        at py4j.reflection.MethodInvoker.invoke(MethodInvoker.java:244)

        at py4j.reflection.ReflectionEngine.invoke(ReflectionEngine.java:357)

        at py4j.Gateway.invoke(Gateway.java:282)

        at py4j.commands.AbstractCommand.invokeMethod(AbstractCommand.java:132)

        at py4j.commands.CallCommand.execute(CallCommand.java:79)

        at py4j.GatewayConnection.run(GatewayConnection.java:238)

        at java.lang.Thread.run(Unknown Source)

Caused by: org.apache.spark.SparkException: Failed to execute user defined␣
↪function(VectorAssembler$$Lambda$3659/168973317: (struct<Age_encoded:
↪struct<type:tinyint,size:int,indices:array<int>,values:
↪array<double>>,Pregnancies_double_VectorAssembler_53ccd1a89127:double,Glucose
↪double,BloodPressure:double,SkinThickness:double,Insulin:double,BMI:
↪double,DiabetesPedigreeFunction:double>) => struct<type:tinyint,size:
↪int,indices:array<int>,values:array<double>>)

        at org.apache.spark.sql.catalyst.expressions.
↪GeneratedClass$GeneratedIteratorForCodegenStage1.processNext(Unknown Source)

        at org.apache.spark.sql.execution.BufferedRowIterator.
↪hasNext(BufferedRowIterator.java:43)

        at org.apache.spark.sql.execution.WholeStageCodegenExec$$anon$1.
↪hasNext(WholeStageCodegenExec.scala:755)
```

```
	at org.apache.spark.sql.execution.aggregate.ObjectHashAggregateExec.
↪$anonfun$doExecute$2(ObjectHashAggregateExec.scala:87)

	at org.apache.spark.sql.execution.aggregate.ObjectHashAggregateExec.
↪$anonfun$doExecute$2$adapted(ObjectHashAggregateExec.scala:85)

	at org.apache.spark.rdd.RDD.
↪$anonfun$mapPartitionsWithIndexInternal$2(RDD.scala:885)

	at org.apache.spark.rdd.RDD.
↪$anonfun$mapPartitionsWithIndexInternal$2$adapted(RDD.scala:885)

	at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitionsRDD.scala
↪52)

	at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scala:373)

	at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)

	at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitionsRDD.scala
↪52)

	at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scala:373)

	at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)

	at org.apache.spark.shuffle.ShuffleWriteProcessor.
↪write(ShuffleWriteProcessor.scala:59)

	at org.apache.spark.scheduler.ShuffleMapTask.runTask(ShuffleMapTask.
↪scala:99)

	at org.apache.spark.scheduler.ShuffleMapTask.runTask(ShuffleMapTask.
↪scala:52)

	at org.apache.spark.scheduler.Task.run(Task.scala:131)

	at org.apache.spark.executor.Executor$TaskRunner.$anonfun$run$3(Executo
↪scala:498)

	at org.apache.spark.util.Utils$.tryWithSafeFinally(Utils.scala:1439)

	at org.apache.spark.executor.Executor$TaskRunner.run(Executor.scala:501

	at java.util.concurrent.ThreadPoolExecutor.runWorker(Unknown Source)

	at java.util.concurrent.ThreadPoolExecutor$Worker.run(Unknown Source)
```

```
        … 1 more

Caused by: org.apache.spark.SparkException: Encountered NaN while assembling a␣
 ↪row with handleInvalid = "error". Consider
removing NaNs from dataset or using handleInvalid = "keep" or "skip".

        at org.apache.spark.ml.feature.VectorAssembler$.
 ↪$anonfun$assemble$1(VectorAssembler.scala:264)

        at org.apache.spark.ml.feature.VectorAssembler$.
 ↪$anonfun$assemble$1$adapted(VectorAssembler.scala:260)

        at scala.collection.IndexedSeqOptimized.foreach(IndexedSeqOptimized.
 ↪scala:36)

        at scala.collection.IndexedSeqOptimized.foreach$(IndexedSeqOptimized.
 ↪scala:33)

        at scala.collection.mutable.WrappedArray.foreach(WrappedArray.scala:38)

        at org.apache.spark.ml.feature.VectorAssembler$.assemble(VectorAssembler.
 ↪scala:260)

        at org.apache.spark.ml.feature.VectorAssembler.
 ↪$anonfun$transform$6(VectorAssembler.scala:143)

        … 23 more
```

[ ]:
```
pipeline_df.toPandas().head()
```

[ ]:
```
pipeline_df.printSchema()
```

[ ]:
```
df=pipeline_df
```

6 WHAT IS THIS STEP DOING?

splitting the data into test data and training data

[34]:
```
train, test = df.randomSplit([0.8, 0.2], seed = 2018)
print("Training Dataset Count: " + str(train.count()))
print("Test Dataset Count: " + str(test.count()))
```

```
Training Dataset Count: 617
Test Dataset Count: 151
```

[35]:
```
train.groupby("HasDiabities").count().show()
```

```
+------------+-----+
|HasDiabities|count|
+------------+-----+
|          No|  394|
|         yes|  223|
+------------+-----+
```

7 WHAT IS THIS STEP DOING?

Applying Linear regression

[36]:
```
from pyspark.ml.classification import LogisticRegression
lr = LogisticRegression(featuresCol = 'features', labelCol = 'label', maxIter=5)
lrModel = lr.fit(train)
predictions = lrModel.transform(test)
#predictions_train = lrModel.transform(train)
predictions.select('label', 'features',  'rawPrediction', 'prediction',␣
 ↪'probability').toPandas().head(5)
```

```
---------------------------------------------------------------------------
IllegalArgumentException                  Traceback (most recent call last)
Input In [36], in <cell line: 3>()
      1 from pyspark.ml.classification import LogisticRegression
      2 lr = LogisticRegression(featuresCol = 'features', labelCol = 'label',␣
 ↪maxIter=5)
----> 3 lrModel = lr.fit(train)
      4 predictions = lrModel.transform(test)
      5 #predictions_train = lrModel.transform(train)

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pyspark\ml\base.
 ↪py:161, in Estimator.fit(self, dataset, params)
    159         return self.copy(params)._fit(dataset)
    160     else:
--> 161         return self._fit(dataset)
    162 else:
    163     raise ValueError("Params must be either a param map or a list/tuple
 ↪of param maps, "
    164                      "but got %s." % type(params))

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pyspark\ml\wrapper.
 ↪py:335, in JavaEstimator._fit(self, dataset)
    334 def _fit(self, dataset):
--> 335     java_model = self._fit_java(dataset)
```

```
    336     model = self._create_model(java_model)
    337     return self._copyValues(model)

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pyspark\ml\wrapper.
 ↪py:332, in JavaEstimator._fit_java(self, dataset)
    318 """
    319 Fits a Java model to the input dataset.
    320
   (…)
    329     fitted Java model
    330 """
    331 self._transfer_params_to_java()
--> 332 return self._java_obj.fit(dataset._jdf)

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\py4j\java_gateway.
 ↪py:1304, in JavaMember.__call__(self, *args)
   1298 command = proto.CALL_COMMAND_NAME +\
   1299     self.command_header +\
   1300     args_command +\
   1301     proto.END_COMMAND_PART
   1303 answer = self.gateway_client.send_command(command)
-> 1304 return_value = get_return_value(
   1305     answer, self.gateway_client, self.target_id, self.name)
   1307 for temp_arg in temp_args:
   1308     temp_arg._detach()

File ~\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.
 ↪10_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\pyspark\sql\utils.
 ↪py:117, in capture_sql_exception.<locals>.deco(*a, **kw)
    113 converted = convert_exception(e.java_exception)
    114 if not isinstance(converted, UnknownException):
    115     # Hide where the exception came from that shows a non-Pythonic
    116     # JVM exception message.
--> 117     raise converted from None
    118 else:
    119     raise

IllegalArgumentException: features does not exist. Available: Pregnancies,␣
 ↪Glucose, BloodPressure, SkinThickness, Insulin, BMI, DiabetesPedigreeFunction␣
 ↪HasDiabities, Age_udf
```

[37]:
```
lrModel.coefficients
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Input In [37], in <cell line: 1>()
```

```
----> 1 lrModel.coefficients

NameError: name 'lrModel' is not defined
```

8 WHAT DO WE CONCLUDE FROM THE Confusion Matrix and accuracy?

[40]:
```
class_names=[1.0,0.0]
import itertools
def plot_confusion_matrix(cm, classes,
                          normalize=False,
                          title='Confusion matrix',
                          cmap=plt.cm.Blues):
    """
    This function prints and plots the confusion matrix.
    Normalization can be applied by setting `normalize=True`.
    """
    if normalize:
        cm = cm.astype('float') / cm.sum(axis=1)[:, np.newaxis]
        print("Normalized confusion matrix")
    else:
        print('Confusion matrix, without normalization')

    print(cm)

    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title)
    plt.colorbar()
    tick_marks = np.arange(len(classes))
    plt.xticks(tick_marks, classes, rotation=45)
    plt.yticks(tick_marks, classes)

    fmt = '.2f' if normalize else 'd'
    thresh = cm.max() / 2.
    for i, j in itertools.product(range(cm.shape[0]), range(cm.shape[1])):
        plt.text(j, i, format(cm[i, j], fmt),
                 horizontalalignment="center",
                 color="white" if cm[i, j] > thresh else "black")

    plt.tight_layout()
    plt.ylabel('True label')
    plt.xlabel('Predicted label')
```

[41]:
```
y_true = predictions.select("label")
y_true = y_true.toPandas()
```

```
y_pred = predictions.select("prediction")
y_pred = y_pred.toPandas()

cnf_matrix = confusion_matrix(y_true, y_pred,labels=class_names)
#cnf_matrix
plt.figure()
plot_confusion_matrix(cnf_matrix, classes=class_names,
                      title='Confusion matrix')
plt.show()
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Input In [41], in <cell line: 1>()
----> 1 y_true = predictions.select("label")
      2 y_true = y_true.toPandas()
      4 y_pred = predictions.select("prediction")

NameError: name 'predictions' is not defined
```

[42]:
```
accuracy = predictions.filter(predictions.label == predictions.prediction).
 ↪count() / float(predictions.count())
print("Accuracy : ",accuracy)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Input In [42], in <cell line: 1>()
----> 1 accuracy = predictions.filter(predictions.label == predictions.
 ↪prediction).count() / float(predictions.count())
      2 print("Accuracy : ",accuracy)

NameError: name 'predictions' is not defined
```

9 The end!

[ ]:
```

```