```
#!pip install pyspark
!pip install Spark
```

```
Requirement already satisfied: Spark in /usr/local/lib/python3.10/dist-packages (0.2.1)
```

Task

- Predict if a patient is Hep or not based parameter
- The data set contains laboratory values of blood donors and Hepatitis C patients and demographic values like age

```
 # Load our Pkgs
from pyspark import SparkContext
```

```
#sc.stop()
sc = SparkContext(master='local[2]')
```

```
# Spark UI
sc
```

SparkContext

Spark UI

Version
    v3.4.1
Master
    local[2]
AppName
    pyspark-shell

```
# Load Pkgs
from pyspark.sql import SparkSession
```

```
# Spark
spark = SparkSession.builder.appName("MLwithSpark").getOrCreate()
```

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

```
# Load our dataset
df = spark.read.csv("/content/drive/My Drive/hcvdata.csv", header=True, inferSchema=True)
```

```
# Preview Dataset
df.show()
```

```
+---+-------------+---+---+----+----+----+----+----+-----+----+-----+----+----+
|_c0|     Category|Age|Sex| ALB| ALP| ALT| AST| BIL|  CHE|CHOL| CREA| GGT|PROT|
+---+-------------+---+---+----+----+----+----+----+-----+----+-----+----+----+
|  1|0=Blood Donor| 32|  m|38.5|52.5| 7.7|22.1| 7.5| 6.93|3.23|106.0|12.1|  69|
|  2|0=Blood Donor| 32|  m|38.5|70.3|  18|24.7| 3.9|11.17| 4.8| 74.0|15.6|76.5|
|  3|0=Blood Donor| 32|  m|46.9|74.7|36.2|52.6| 6.1| 8.84| 5.2| 86.0|33.2|79.3|
|  4|0=Blood Donor| 32|  m|43.2|  52|30.6|22.6|18.9| 7.33|4.74| 80.0|33.8|75.7|
|  5|0=Blood Donor| 32|  m|39.2|74.1|32.6|24.8| 9.6| 9.15|4.32| 76.0|29.9|68.7|
|  6|0=Blood Donor| 32|  m|41.6|43.3|18.5|19.7|12.3| 9.92|6.05|111.0|91.0|  74|
|  7|0=Blood Donor| 32|  m|46.3|41.3|17.5|17.8| 8.5| 7.01|4.79| 70.0|16.9|74.5|
|  8|0=Blood Donor| 32|  m|42.2|41.9|35.8|31.1|16.1| 5.82| 4.6|109.0|21.5|67.1|
|  9|0=Blood Donor| 32|  m|50.9|65.5|23.2|21.2| 6.9| 8.69| 4.1| 83.0|13.7|71.3|
| 10|0=Blood Donor| 32|  m|42.4|86.3|20.3|20.0|35.2| 5.46|4.45| 81.0|15.9|69.9|
| 11|0=Blood Donor| 32|  m|44.3|52.3|21.7|22.4|17.2| 4.15|3.57| 78.0|24.1|75.4|
| 12|0=Blood Donor| 33|  m|46.4|68.2|10.3|20.0| 5.7| 7.36| 4.3| 79.0|18.7|68.6|
| 13|0=Blood Donor| 33|  m|36.3|78.6|23.6|22.0| 7.0| 8.56|5.38| 78.0|19.4|68.7|
| 14|0=Blood Donor| 33|  m|  39|51.7|15.9|24.0| 6.8| 6.46|3.38| 65.0| 7.0|70.4|
| 15|0=Blood Donor| 33|  m|38.7|39.8|22.5|23.0| 4.1| 4.63|4.97| 63.0|15.2|71.9|
| 16|0=Blood Donor| 33|  m|41.8|  65|33.1|38.0| 6.6| 8.83|4.43| 71.0|24.0|72.7|
| 17|0=Blood Donor| 33|  m|40.9|  73|17.2|22.9|10.0| 6.98|5.22| 90.0|14.7|72.4|
| 18|0=Blood Donor| 33|  m|45.2|88.3|32.4|31.2|10.1| 9.78|5.51|102.0|48.5|76.5|
| 19|0=Blood Donor| 33|  m|36.6|57.1|38.9|40.3|24.9| 9.62| 5.5|112.0|27.6|69.3|
| 20|0=Blood Donor| 33|  m|  42|63.1|32.6|34.9|11.2| 7.01|4.05|105.0|19.1|68.1|
+---+-------------+---+---+----+----+----+----+----+-----+----+-----+----+----+
only showing top 20 rows
```

```
 # check for columns
print(df.columns)
```

```
['_c0', 'Category', 'Age', 'Sex', 'ALB', 'ALP', 'ALT', 'AST', 'BIL', 'CHE', 'CHOL', 'CREA', 'GGT', 'PROT']
```

```
 # Rearrange
df = df.select('Age', 'Sex', 'ALB', 'ALP', 'ALT', 'AST', 'BIL', 'CHE', 'CHOL','CREA', 'GGT', 'PROT','Category')
```

```
df.show(5)
```

```
+---+---+----+----+----+----+----+-----+----+-----+----+----+------------+
|Age|Sex| ALB| ALP| ALT| AST| BIL|  CHE|CHOL| CREA| GGT|PROT|    Category|
+---+---+----+----+----+----+----+-----+----+-----+----+----+------------+
| 32|  m|38.5|52.5| 7.7|22.1| 7.5| 6.93|3.23|106.0|12.1|  69|0=Blood Donor|
| 32|  m|38.5|70.3|  18|24.7| 3.9|11.17| 4.8| 74.0|15.6|76.5|0=Blood Donor|
| 32|  m|46.9|74.7|36.2|52.6| 6.1| 8.84| 5.2| 86.0|33.2|79.3|0=Blood Donor|
| 32|  m|43.2|  52|30.6|22.6|18.9| 7.33|4.74| 80.0|33.8|75.7|0=Blood Donor|
| 32|  m|39.2|74.1|32.6|24.8| 9.6| 9.15|4.32| 76.0|29.9|68.7|0=Blood Donor|
+---+---+----+----+----+----+----+-----+----+-----+----+----+------------+
only showing top 5 rows
```

```
# Check for datatypes
# Before InferSchema
df.dtypes
```

```
[('Age', 'int'),
 ('Sex', 'string'),
 ('ALB', 'string'),
 ('ALP', 'string'),
 ('ALT', 'string'),
 ('AST', 'double'),
 ('BIL', 'double'),
 ('CHE', 'double'),
 ('CHOL', 'string'),
 ('CREA', 'double'),
 ('GGT', 'double'),
 ('PROT', 'string'),
 ('Category', 'string')]
```

```
 # Check for the Schema
df.printSchema()
```

```
root
 |-- Age: integer (nullable = true)
 |-- Sex: string (nullable = true)
 |-- ALB: string (nullable = true)
 |-- ALP: string (nullable = true)
 |-- ALT: string (nullable = true)
 |-- AST: double (nullable = true)
 |-- BIL: double (nullable = true)
 |-- CHE: double (nullable = true)
 |-- CHOL: string (nullable = true)
 |-- CREA: double (nullable = true)
 |-- GGT: double (nullable = true)
 |-- PROT: string (nullable = true)
 |-- Category: string (nullable = true)
```

```
 # Descriptive summary
print(df.describe().show())
```

```
+-------+------------------+----+-----------------+------------------+------------------+-----------------+------------------+----------
|summary|               Age| Sex|              ALB|               ALP|               ALT|              AST|               BIL|
+-------+------------------+----+-----------------+------------------+------------------+-----------------+------------------+----------
|  count|               615| 615|              615|               615|               615|              615|               615|
|   mean| 47.40813008130081|null|41.62019543973941| 68.28391959798999| 28.45081433224754|34.78634146341462|11.396747967479675| 8.1966341
| stddev|10.055105445519239|null|5.780629404103076|26.028315300123676|25.469688813870942|33.09069033855156|19.673149805846588|2.20565727
|    min|                19|   f|             14.9|             100.4|               0.9|             10.6|               0.8|
|    max|                77|   m|               NA|                NA|                NA|            324.0|             254.0|
+-------+------------------+----+-----------------+------------------+------------------+-----------------+------------------+----------

None
```

```
# Value Count
df.groupBy('Category').count().show()
```

```
+--------------------+-----+
|            Category|count|
+--------------------+-----+
|       0=Blood Donor|  533|
|         3=Cirrhosis|   30|
|          2=Fibrosis|   21|
|0s=suspect Blood ...|    7|
|         1=Hepatitis|   24|
+--------------------+-----+
```

Feature Engineering

- Numberical Values
- Vectorization
- Scaling

```
import pyspark.ml
```

```
dir(pyspark.ml)
```

```
['Estimator',
 'Model',
 'Pipeline',
 'PipelineModel',
 'PredictionModel',
 'Predictor',
 'TorchDistributor',
 'Transformer',
 'UnaryTransformer',
 '__all__',
 '__builtins__',
 '__cached__',
 '__doc__',
 '__file__',
 '__loader__',
 '__name__',
 '__package__',
 '__path__',
 '__spec__',
 'base',
 'classification',
 'clustering',
 'common',
 'evaluation',
 'feature',
 'fpm',
 'image',
 'linalg',
 'param',
 'pipeline',
 'recommendation',
 'regression',
 'stat',
 'torch',
 'tree',
 'tuning',
 'util',
 'wrapper']
```

```
# Load ML Pkgs
from pyspark.ml.feature import VectorAssembler,StringIndexer
```

```
df.show(4)
```

```
+---+---+----+----+----+----+----+-----+----+-----+----+----+-------------+
|Age|Sex| ALB| ALP| ALT| AST| BIL|  CHE|CHOL| CREA| GGT|PROT|     Category|
+---+---+----+----+----+----+----+-----+----+-----+----+----+-------------+
| 32|  m|38.5|52.5| 7.7|22.1| 7.5| 6.93|3.23|106.0|12.1|  69|0=Blood Donor|
| 32|  m|38.5|70.3|  18|24.7| 3.9|11.17| 4.8| 74.0|15.6|76.5|0=Blood Donor|
| 32|  m|46.9|74.7|36.2|52.6| 6.1| 8.84| 5.2| 86.0|33.2|79.3|0=Blood Donor|
| 32|  m|43.2|  52|30.6|22.6|18.9| 7.33|4.74| 80.0|33.8|75.7|0=Blood Donor|
+---+---+----+----+----+----+----+-----+----+-----+----+----+-------------+
only showing top 4 rows
```

```
# Unique Values for Sex
df.select('Sex').distinct().show()
```

```
+---+
|Sex|
+---+
|  m|
|  f|
+---+
```

```
# Convert the string into numerical code
# label encoding
genderEncoder = StringIndexer(inputCol='Sex',outputCol='Gender').fit(df)
```

```
df = genderEncoder.transform(df)
```

```
df.show(5)
```

```
+---+---+----+----+----+----+----+-----+----+-----+----+----+-------------+------+
|Age|Sex| ALB| ALP| ALT| AST| BIL|  CHE|CHOL| CREA| GGT|PROT|     Category|Gender|
+---+---+----+----+----+----+----+-----+----+-----+----+----+-------------+------+
| 32|  m|38.5|52.5| 7.7|22.1| 7.5| 6.93|3.23|106.0|12.1|  69|0=Blood Donor|   0.0|
| 32|  m|38.5|70.3|  18|24.7| 3.9|11.17| 4.8| 74.0|15.6|76.5|0=Blood Donor|   0.0|
| 32|  m|46.9|74.7|36.2|52.6| 6.1| 8.84| 5.2| 86.0|33.2|79.3|0=Blood Donor|   0.0|
| 32|  m|43.2|  52|30.6|22.6|18.9| 7.33|4.74| 80.0|33.8|75.7|0=Blood Donor|   0.0|
| 32|  m|39.2|74.1|32.6|24.8| 9.6| 9.15|4.32| 76.0|29.9|68.7|0=Blood Donor|   0.0|
+---+---+----+----+----+----+----+-----+----+-----+----+----+-------------+------+
only showing top 5 rows
```

```
# Encoding for Category
# Label Encoding
catEncoder = StringIndexer(inputCol='Category',outputCol='Target').fit(df)
df = catEncoder.transform(df)
```

```
df.show(5)
```

```
+---+---+----+----+----+----+----+-----+----+-----+----+----+-------------+------+------+
|Age|Sex| ALB| ALP| ALT| AST| BIL|  CHE|CHOL| CREA| GGT|PROT|     Category|Gender|Target|
+---+---+----+----+----+----+----+-----+----+-----+----+----+-------------+------+------+
| 32|  m|38.5|52.5| 7.7|22.1| 7.5| 6.93|3.23|106.0|12.1|  69|0=Blood Donor|   0.0|   0.0|
| 32|  m|38.5|70.3|  18|24.7| 3.9|11.17| 4.8| 74.0|15.6|76.5|0=Blood Donor|   0.0|   0.0|
| 32|  m|46.9|74.7|36.2|52.6| 6.1| 8.84| 5.2| 86.0|33.2|79.3|0=Blood Donor|   0.0|   0.0|
| 32|  m|43.2|  52|30.6|22.6|18.9| 7.33|4.74| 80.0|33.8|75.7|0=Blood Donor|   0.0|   0.0|
| 32|  m|39.2|74.1|32.6|24.8| 9.6| 9.15|4.32| 76.0|29.9|68.7|0=Blood Donor|   0.0|   0.0|
+---+---+----+----+----+----+----+-----+----+-----+----+----+-------------+------+------+
only showing top 5 rows
```

```
# Get the labels
catEncoder.labels
```

```
['0=Blood Donor',
 '3=Cirrhosis',
 '1=Hepatitis',
 '2=Fibrosis',
 '0s=suspect Blood Donor']
```

```
# IndexToString
from pyspark.ml.feature import IndexToString
```

```
converter = IndexToString(inputCol='Target',outputCol='orig_cat')
```

```
converted_df = converter.transform(df)
```

```
converted_df.show()
```

```
+---+---+----+----+----+----+----+-----+----+-----+----+----+-------------+------+------+-------------+
|Age|Sex| ALB| ALP| ALT| AST| BIL|  CHE|CHOL| CREA| GGT|PROT|     Category|Gender|Target|     orig_cat|
+---+---+----+----+----+----+----+-----+----+-----+----+----+-------------+------+------+-------------+
| 32|  m|38.5|52.5| 7.7|22.1| 7.5| 6.93|3.23|106.0|12.1|  69|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
| 32|  m|38.5|70.3|  18|24.7| 3.9|11.17| 4.8| 74.0|15.6|76.5|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
| 32|  m|46.9|74.7|36.2|52.6| 6.1| 8.84| 5.2| 86.0|33.2|79.3|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
| 32|  m|43.2|  52|30.6|22.6|18.9| 7.33|4.74| 80.0|33.8|75.7|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
| 32|  m|39.2|74.1|32.6|24.8| 9.6| 9.15|4.32| 76.0|29.9|68.7|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
| 32|  m|41.6|43.3|18.5|19.7|12.3| 9.92|6.05|111.0|91.0|  74|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
| 32|  m|46.3|41.3|17.5|17.8| 8.5| 7.01|4.79| 70.0|16.9|74.5|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
| 32|  m|42.2|41.9|35.8|31.1|16.1| 5.82| 4.6|109.0|21.5|67.1|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
| 32|  m|50.9|65.5|23.2|21.2| 6.9| 8.69| 4.1| 83.0|13.7|71.3|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
| 32|  m|42.4|86.3|20.3|20.0|35.2| 5.46|4.45| 81.0|15.9|69.9|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
| 32|  m|44.3|52.3|21.7|22.4|17.2| 4.15|3.57| 78.0|24.1|75.4|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
| 33|  m|46.4|68.2|10.3|20.0| 5.7| 7.36| 4.3| 79.0|18.7|68.6|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
| 33|  m|36.3|78.6|23.6|22.0| 7.0| 8.56|5.38| 78.0|19.4|68.7|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
| 33|  m|  39|51.7|15.9|24.0| 6.8| 6.46|3.38| 65.0| 7.0|70.4|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
| 33|  m|38.7|39.8|22.5|23.0| 4.1| 4.63|4.97| 63.0|15.2|71.9|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
| 33|  m|41.8|  65|33.1|38.0| 6.6| 8.83|4.43| 71.0|24.0|72.7|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
| 33|  m|40.9|  73|17.2|22.9|10.0| 6.98|5.22| 90.0|14.7|72.4|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
| 33|  m|45.2|88.3|32.4|31.2|10.1| 9.78|5.51|102.0|48.5|76.5|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
| 33|  m|36.6|57.1|38.9|40.3|24.9| 9.62| 5.5|112.0|27.6|69.3|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
| 33|  m|  42|63.1|32.6|34.9|11.2| 7.01|4.05|105.0|19.1|68.1|0=Blood Donor|   0.0|   0.0|0=Blood Donor|
+---+---+----+----+----+----+----+-----+----+-----+----+----+-------------+------+------+-------------+
only showing top 20 rows
```

```
 ### Feature
df.show()
```

```
+---+---+----+----+----+----+----+-----+----+-----+----+----+-------------+------+------+
|Age|Sex| ALB| ALP| ALT| AST| BIL|  CHE|CHOL| CREA| GGT|PROT|     Category|Gender|Target|
+---+---+----+----+----+----+----+-----+----+-----+----+----+-------------+------+------+
| 32|  m|38.5|52.5| 7.7|22.1| 7.5| 6.93|3.23|106.0|12.1|  69|0=Blood Donor|   0.0|   0.0|
| 32|  m|38.5|70.3|  18|24.7| 3.9|11.17| 4.8| 74.0|15.6|76.5|0=Blood Donor|   0.0|   0.0|
| 32|  m|46.9|74.7|36.2|52.6| 6.1| 8.84| 5.2| 86.0|33.2|79.3|0=Blood Donor|   0.0|   0.0|
| 32|  m|43.2|  52|30.6|22.6|18.9| 7.33|4.74| 80.0|33.8|75.7|0=Blood Donor|   0.0|   0.0|
| 32|  m|39.2|74.1|32.6|24.8| 9.6| 9.15|4.32| 76.0|29.9|68.7|0=Blood Donor|   0.0|   0.0|
| 32|  m|41.6|43.3|18.5|19.7|12.3| 9.92|6.05|111.0|91.0|  74|0=Blood Donor|   0.0|   0.0|
| 32|  m|46.3|41.3|17.5|17.8| 8.5| 7.01|4.79| 70.0|16.9|74.5|0=Blood Donor|   0.0|   0.0|
| 32|  m|42.2|41.9|35.8|31.1|16.1| 5.82| 4.6|109.0|21.5|67.1|0=Blood Donor|   0.0|   0.0|
| 32|  m|50.9|65.5|23.2|21.2| 6.9| 8.69| 4.1| 83.0|13.7|71.3|0=Blood Donor|   0.0|   0.0|
| 32|  m|42.4|86.3|20.3|20.0|35.2| 5.46|4.45| 81.0|15.9|69.9|0=Blood Donor|   0.0|   0.0|
| 32|  m|44.3|52.3|21.7|22.4|17.2| 4.15|3.57| 78.0|24.1|75.4|0=Blood Donor|   0.0|   0.0|
| 33|  m|46.4|68.2|10.3|20.0| 5.7| 7.36| 4.3| 79.0|18.7|68.6|0=Blood Donor|   0.0|   0.0|
| 33|  m|36.3|78.6|23.6|22.0| 7.0| 8.56|5.38| 78.0|19.4|68.7|0=Blood Donor|   0.0|   0.0|
| 33|  m|  39|51.7|15.9|24.0| 6.8| 6.46|3.38| 65.0| 7.0|70.4|0=Blood Donor|   0.0|   0.0|
| 33|  m|38.7|39.8|22.5|23.0| 4.1| 4.63|4.97| 63.0|15.2|71.9|0=Blood Donor|   0.0|   0.0|
| 33|  m|41.8|  65|33.1|38.0| 6.6| 8.83|4.43| 71.0|24.0|72.7|0=Blood Donor|   0.0|   0.0|
| 33|  m|40.9|  73|17.2|22.9|10.0| 6.98|5.22| 90.0|14.7|72.4|0=Blood Donor|   0.0|   0.0|
| 33|  m|45.2|88.3|32.4|31.2|10.1| 9.78|5.51|102.0|48.5|76.5|0=Blood Donor|   0.0|   0.0|
| 33|  m|36.6|57.1|38.9|40.3|24.9| 9.62| 5.5|112.0|27.6|69.3|0=Blood Donor|   0.0|   0.0|
| 33|  m|  42|63.1|32.6|34.9|11.2| 7.01|4.05|105.0|19.1|68.1|0=Blood Donor|   0.0|   0.0|
+---+---+----+----+----+----+----+-----+----+-----+----+----+-------------+------+------+
only showing top 20 rows
```

```
print(df.columns)
```

```
['Age', 'Sex', 'ALB', 'ALP', 'ALT', 'AST', 'BIL', 'CHE', 'CHOL', 'CREA', 'GGT', 'PROT', 'Category', 'Gender', 'Target']
```

```
 df.dtypes
```

```
[('Age', 'int'),
 ('Sex', 'string'),
 ('ALB', 'string'),
 ('ALP', 'string'),
 ('ALT', 'string'),
 ('AST', 'double'),
 ('BIL', 'double'),
 ('CHE', 'double'),
 ('CHOL', 'string'),
 ('CREA', 'double'),
 ('GGT', 'double'),
 ('PROT', 'string'),
 ('Category', 'string'),
 ('Gender', 'double'),
 ('Target', 'double')]
```

```
df2 = df.select('Age','Gender', 'ALB', 'ALP', 'ALT', 'AST', 'BIL', 'CHE','CHOL', 'CREA', 'GGT', 'PROT', 'Target')
```

```
df2.printSchema()
```

```
root
 |-- Age: integer (nullable = true)
 |-- Gender: double (nullable = false)
 |-- ALB: string (nullable = true)
 |-- ALP: string (nullable = true)
 |-- ALT: string (nullable = true)
 |-- AST: double (nullable = true)
 |-- BIL: double (nullable = true)
 |-- CHE: double (nullable = true)
 |-- CHOL: string (nullable = true)
 |-- CREA: double (nullable = true)
 |-- GGT: double (nullable = true)
 |-- PROT: string (nullable = true)
 |-- Target: double (nullable = false)
```

```
# df2.fillna(0,subset=['col1'])
df2 = df2.toPandas().replace('NA',0).astype(float)
```

```
type(df2)
```

```
pandas.core.frame.DataFrame
```

```
type(df)
```

```
pyspark.sql.dataframe.DataFrame
```

```
# Convert To PySpark Dataframe
new_df = spark.createDataFrame(df2)
```

```
new_df.show()
```

```
+----+------+----+----+----+----+----+-----+----+-----+----+----+------+
| Age|Gender| ALB| ALP| ALT| AST| BIL|  CHE|CHOL| CREA| GGT|PROT|Target|
+----+------+----+----+----+----+----+-----+----+-----+----+----+------+
|32.0|   0.0|38.5|52.5| 7.7|22.1| 7.5| 6.93|3.23|106.0|12.1|69.0|   0.0|
|32.0|   0.0|38.5|70.3|18.0|24.7| 3.9|11.17| 4.8| 74.0|15.6|76.5|   0.0|
|32.0|   0.0|46.9|74.7|36.2|52.6| 6.1| 8.84| 5.2| 86.0|33.2|79.3|   0.0|
|32.0|   0.0|43.2|52.0|30.6|22.6|18.9| 7.33|4.74| 80.0|33.8|75.7|   0.0|
|32.0|   0.0|39.2|74.1|32.6|24.8| 9.6| 9.15|4.32| 76.0|29.9|68.7|   0.0|
|32.0|   0.0|41.6|43.3|18.5|19.7|12.3| 9.92|6.05|111.0|91.0|74.0|   0.0|
|32.0|   0.0|46.3|41.3|17.5|17.8| 8.5| 7.01|4.79| 70.0|16.9|74.5|   0.0|
|32.0|   0.0|42.2|41.9|35.8|31.1|16.1| 5.82| 4.6|109.0|21.5|67.1|   0.0|
|32.0|   0.0|50.9|65.5|23.2|21.2| 6.9| 8.69| 4.1| 83.0|13.7|71.3|   0.0|
|32.0|   0.0|42.4|86.3|20.3|20.0|35.2| 5.46|4.45| 81.0|15.9|69.9|   0.0|
|32.0|   0.0|44.3|52.3|21.7|22.4|17.2| 4.15|3.57| 78.0|24.1|75.4|   0.0|
|33.0|   0.0|46.4|68.2|10.3|20.0| 5.7| 7.36| 4.3| 79.0|18.7|68.6|   0.0|
|33.0|   0.0|36.3|78.6|23.6|22.0| 7.0| 8.56|5.38| 78.0|19.4|68.7|   0.0|
|33.0|   0.0|39.0|51.7|15.9|24.0| 6.8| 6.46|3.38| 65.0| 7.0|70.4|   0.0|
|33.0|   0.0|38.7|39.8|22.5|23.0| 4.1| 4.63|4.97| 63.0|15.2|71.9|   0.0|
|33.0|   0.0|41.8|65.0|33.1|38.0| 6.6| 8.83|4.43| 71.0|24.0|72.7|   0.0|
|33.0|   0.0|40.9|73.0|17.2|22.9|10.0| 6.98|5.22| 90.0|14.7|72.4|   0.0|
|33.0|   0.0|45.2|88.3|32.4|31.2|10.1| 9.78|5.51|102.0|48.5|76.5|   0.0|
|33.0|   0.0|36.6|57.1|38.9|40.3|24.9| 9.62| 5.5|112.0|27.6|69.3|   0.0|
|33.0|   0.0|42.0|63.1|32.6|34.9|11.2| 7.01|4.05|105.0|19.1|68.1|   0.0|
+----+------+----+----+----+----+----+-----+----+-----+----+----+------+
only showing top 20 rows
```

```
# Check For DTypes and Schema
new_df.printSchema()
```

```
root
 |-- Age: double (nullable = true)
 |-- Gender: double (nullable = true)
 |-- ALB: double (nullable = true)
 |-- ALP: double (nullable = true)
 |-- ALT: double (nullable = true)
 |-- AST: double (nullable = true)
 |-- BIL: double (nullable = true)
 |-- CHE: double (nullable = true)
 |-- CHOL: double (nullable = true)
 |-- CREA: double (nullable = true)
 |-- GGT: double (nullable = true)
 |-- PROT: double (nullable = true)
 |-- Target: double (nullable = true)
```

```
required_features = ['Age','Gender', 'ALB', 'ALP', 'ALT', 'AST', 'BIL', 'CHE','CHOL', 'CREA', 'GGT', 'PROT', 'Target']
```

```
# VectorAsm
vec_assembler =VectorAssembler(inputCols = required_features, outputCol = 'features')
```

```
vec_df = vec_assembler.transform(new_df)
```

```
vec_df.show(5)
```

```
+----+------+----+----+----+----+----+-----+----+-----+----+----+------+--------------------+
| Age|Gender| ALB| ALP| ALT| AST| BIL|  CHE|CHOL| CREA| GGT|PROT|Target|            features|
+----+------+----+----+----+----+----+-----+----+-----+----+----+------+--------------------+
|32.0|   0.0|38.5|52.5| 7.7|22.1| 7.5| 6.93|3.23|106.0|12.1|69.0|   0.0|[32.0,0.0,38.5,52...|
|32.0|   0.0|38.5|70.3|18.0|24.7| 3.9|11.17| 4.8| 74.0|15.6|76.5|   0.0|[32.0,0.0,38.5,70...|
|32.0|   0.0|46.9|74.7|36.2|52.6| 6.1| 8.84| 5.2| 86.0|33.2|79.3|   0.0|[32.0,0.0,46.9,74...|
|32.0|   0.0|43.2|52.0|30.6|22.6|18.9| 7.33|4.74| 80.0|33.8|75.7|   0.0|[32.0,0.0,43.2,52...|
|32.0|   0.0|39.2|74.1|32.6|24.8| 9.6| 9.15|4.32| 76.0|29.9|68.7|   0.0|[32.0,0.0,39.2,74...|
+----+------+----+----+----+----+----+-----+----+-----+----+----+------+--------------------+
only showing top 5 rows
```

```
#Train, test, split
train_df,test_df = vec_df.randomSplit([0.7,0.3])
```

```
train_df.count()
```

```
431
```

```
train_df.show(4)
```

```
+----+------+----+----+----+----+----+-----+----+-----+----+----+------+--------------------+
| Age|Gender| ALB| ALP| ALT| AST| BIL|  CHE|CHOL| CREA| GGT|PROT|Target|            features|
+----+------+----+----+----+----+----+-----+----+-----+----+----+------+--------------------+
|32.0|   0.0|38.5|70.3|18.0|24.7| 3.9|11.17| 4.8| 74.0|15.6|76.5|   0.0|[32.0,0.0,38.5,70...|
|32.0|   0.0|39.2|74.1|32.6|24.8| 9.6| 9.15|4.32| 76.0|29.9|68.7|   0.0|[32.0,0.0,39.2,74...|
|32.0|   0.0|41.6|43.3|18.5|19.7|12.3| 9.92|6.05|111.0|91.0|74.0|   0.0|[32.0,0.0,41.6,43...|
|32.0|   0.0|42.2|41.9|35.8|31.1|16.1| 5.82| 4.6|109.0|21.5|67.1|   0.0|[32.0,0.0,42.2,41...|
+----+------+----+----+----+----+----+-----+----+-----+----+----+------+--------------------+
only showing top 4 rows
```

Model Building

- Pyspark.ml: DataFrame
- Pyspark.mllib: RDD /Legacy

```
from pyspark.ml.classification import LogisticRegression,DecisionTreeClassifier
```

```
# Logist Model
lr = LogisticRegression(featuresCol='features',labelCol='Target')
```

```
lr_model = lr.fit(train_df)
```

```
y_pred = lr_model.transform(test_df)
```

```
y_pred.show()
```

```
+----+------+----+----+----+----+----+-----+----+-----+----+----+------+--------------------+--------------------+--------------------+-
| Age|Gender| ALB| ALP| ALT| AST| BIL|  CHE|CHOL| CREA| GGT|PROT|Target|            features|       rawPrediction|         probability|p
+----+------+----+----+----+----+----+-----+----+-----+----+----+------+--------------------+--------------------+--------------------+-
|32.0|   0.0|38.5|52.5| 7.7|22.1| 7.5| 6.93|3.23|106.0|12.1|69.0|   0.0|[32.0,0.0,38.5,52...|[152.153536377364...|[1.0,1.2102270124...|
|32.0|   0.0|46.3|41.3|17.5|17.8| 8.5| 7.01|4.79| 70.0|16.9|74.5|   0.0|[32.0,0.0,46.3,41...|[163.974003571383...|[1.0,1.4184343127...|
|32.0|   0.0|46.9|74.7|36.2|52.6| 6.1| 8.84| 5.2| 86.0|33.2|79.3|   0.0|[32.0,0.0,46.9,74...|[135.171696661936...|[1.0,1.1118933465...|
|33.0|   0.0|41.8|65.0|33.1|38.0| 6.6| 8.83|4.43| 71.0|24.0|72.7|   0.0|[33.0,0.0,41.8,65...|[152.851129201431...|[1.0,1.2491039398...|
|33.0|   0.0|46.4|68.2|10.3|20.0| 5.7| 7.36| 4.3| 79.0|18.7|68.6|   0.0|[33.0,0.0,46.4,68...|[183.033383912316...|[1.0,1.1821949982...|
|34.0|   0.0|43.6|58.9|47.1|31.1|18.5| 9.14|4.99| 95.0|22.2|69.3|   0.0|[34.0,0.0,43.6,58...|[163.223651455903...|[1.0,9.0453078279...|
|34.0|   0.0|44.6|84.1|19.6|29.8| 5.8|  7.6|5.07| 95.0| 9.9|71.9|   0.0|[34.0,0.0,44.6,84...|[171.189791895205...|[1.0,1.2358281353...|
|34.0|   0.0|44.8|77.7|36.9|31.0|19.5|10.51|5.59| 80.0|23.7|78.9|   0.0|[34.0,0.0,44.8,77...|[152.311478009094...|[1.0,1.5486334177...|
|34.0|   0.0|46.1|70.6|35.8|30.0| 7.6|  7.7| 4.2| 93.0|14.3|78.7|   0.0|[34.0,0.0,46.1,70...|[160.454061445685...|[1.0,5.3660910584...|
|35.0|   0.0|27.8|99.0|30.7|27.8| 9.4|  6.8|4.27| 65.0|40.5|80.7|   0.0|[35.0,0.0,27.8,99...|[128.224487479318...|[0.99999999859048...|
|35.0|   0.0|44.7|79.3|53.5|30.8| 9.7|11.39|7.04| 88.0|77.3|77.1|   0.0|[35.0,0.0,44.7,79...|[150.678431331208...|[1.0,9.1045594675...|
|35.0|   0.0|48.7|72.7|24.1|31.0|45.1|  9.4| 3.8| 90.0|20.0|75.8|   0.0|[35.0,0.0,48.7,72...|[149.875533892139...|[1.0,2.6246344838...|
|36.0|   0.0|42.4|47.3|23.0|25.5| 6.1| 9.46|5.29| 79.0|17.5|73.8|   0.0|[36.0,0.0,42.4,47...|[151.814536322398...|[1.0,2.8196659410...|
|37.0|   0.0|38.6|61.2|24.6|31.9| 7.9| 6.02|4.63| 72.0|10.3|56.3|   0.0|[37.0,0.0,38.6,61...|[182.529322676531...|[1.0,1.6817355623...|
|37.0|   0.0|44.8|94.3|32.2|36.7| 6.3| 9.76|4.12|113.0|23.8|72.5|   0.0|[37.0,0.0,44.8,94...|[166.796737069642...|[1.0,5.0842715823...|
|37.0|   0.0|46.1|44.3|42.7|26.5| 6.4|10.86|5.05| 74.0|22.2|73.1|   0.0|[37.0,0.0,46.1,44...|[166.246831912493...|[1.0,5.1324616128...|
|37.0|   0.0|47.9|68.8|40.3|46.9| 6.0| 9.76|6.42| 81.0|22.7|80.6|   0.0|[37.0,0.0,47.9,68...|[141.013759236821...|[1.0,2.4949366294...|
|37.0|   0.0|48.7|62.3|21.0|21.1|41.9| 9.71|4.02| 84.0|16.0|75.1|   0.0|[37.0,0.0,48.7,62...|[158.722166021109...|[1.0,1.0058370997...|
|37.0|   0.0|51.2|84.5|18.8|24.7| 9.9| 8.62|6.59| 94.0|25.3|76.3|   0.0|[37.0,0.0,51.2,84...|[173.981547992496...|[1.0,6.2593408469...|
|38.0|   0.0|45.5|50.2|16.3|22.8|10.9| 8.73|5.88|103.0|13.8|76.1|   0.0|[38.0,0.0,45.5,50...|[148.918163785995...|[1.0,7.0929131137...|
+----+------+----+----+----+----+----+-----+----+-----+----+----+------+--------------------+--------------------+--------------------+-
only showing top 20 rows
```

```
print(y_pred.columns)
```

```
['Age', 'Gender', 'ALB', 'ALP', 'ALT', 'AST', 'BIL', 'CHE', 'CHOL', 'CREA', 'GGT', 'PROT', 'Target', 'features', 'rawPrediction', 'proba
```

```
y_pred.select('target','rawPrediction', 'probability', 'prediction').show()
```

```
+------+--------------------+--------------------+----------+
|target|       rawPrediction|         probability|prediction|
+------+--------------------+--------------------+----------+
|   0.0|[152.153536377364...|[1.0,1.2102270124...|       0.0|
|   0.0|[163.974003571383...|[1.0,1.4184343127...|       0.0|
|   0.0|[135.171696661936...|[1.0,1.1118933465...|       0.0|
|   0.0|[152.851129201431...|[1.0,1.2491039398...|       0.0|
|   0.0|[183.033383912316...|[1.0,1.1821949982...|       0.0|
|   0.0|[163.223651455903...|[1.0,9.0453078279...|       0.0|
|   0.0|[171.189791895205...|[1.0,1.2358281353...|       0.0|
|   0.0|[152.311478009094...|[1.0,1.5486334177...|       0.0|
|   0.0|[160.454061445685...|[1.0,5.3660910584...|       0.0|
|   0.0|[128.224487479318...|[0.99999999859048...|       0.0|
|   0.0|[150.678431331208...|[1.0,9.1045594675...|       0.0|
|   0.0|[149.875533892139...|[1.0,2.6246344838...|       0.0|
|   0.0|[151.814536322398...|[1.0,2.8196659410...|       0.0|
|   0.0|[182.529322676531...|[1.0,1.6817355623...|       0.0|
|   0.0|[166.796737069642...|[1.0,5.0842715823...|       0.0|
|   0.0|[166.246831912493...|[1.0,5.1324616128...|       0.0|
|   0.0|[141.013759236821...|[1.0,2.4949366294...|       0.0|
|   0.0|[158.722166021109...|[1.0,1.0058370997...|       0.0|
|   0.0|[173.981547992496...|[1.0,6.2593408469...|       0.0|
|   0.0|[148.918163785995...|[1.0,7.0929131137...|       0.0|
+------+--------------------+--------------------+----------+
only showing top 20 rows
```

Model Evaluation

```
from pyspark.ml.evaluation import MulticlassClassificationEvaluator
```

```
# How to Check For Accuracy
multi_evaluator = MulticlassClassificationEvaluator(labelCol='Target',metricName='accuracy')
```

```
multi_evaluator.evaluate(y_pred)
```

```
0.9510869565217391
```

```
from pyspark.mllib.evaluation import MulticlassMetrics
```

```
lr_metric = MulticlassMetrics(y_pred['target', 'prediction'].rdd)
```

```
/usr/local/lib/python3.10/dist-packages/pyspark/sql/context.py:157: FutureWarning: Deprecated in 3.0.0. Use SparkSession.builder.getOrCr
  warnings.warn(
dir(lr_metric)
```

```
['__class__',
 '__del__',
 '__delattr__',
 '__dict__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__gt__',
 '__hash__',
 '__init__',
 '__init_subclass__',
 '__le__',
 '__lt__',
 '__module__',
 '__ne__',
 '__new__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__setattr__',
 '__sizeof__',
 '__str__',
 '__subclasshook__',
 '__weakref__',
 '_java_model',
 '_sc',
 'accuracy',
 'call',
 'confusionMatrix',
 'fMeasure',
 'falsePositiveRate',
 'logLoss',
 'precision',
 'recall',
 'truePositiveRate',
 'weightedFMeasure',
 'weightedFalsePositiveRate',
 'weightedPrecision',
 'weightedRecall',
 'weightedTruePositiveRate']
```

```
print("Accuracy",lr_metric.accuracy)
```

```
Accuracy 0.9510869565217391
```

```
print("Precision",lr_metric.precision(1.0))
print("Recall",lr_metric.recall(1.0))
print("F1Score",lr_metric.fMeasure(1.0))
```

```
Precision 0.6875
Recall 1.0
F1Score 0.8148148148148148
```

```
dir(lr_model)
```

```
['__abstractmethods__',
 '__annotations__',
 '__class__',
 '__class_getitem__',
 '__del__',
 '__delattr__',
 '__dict__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__gt__',
 '__hash__',
 '__init__',
 '__init_subclass__',
 '__le__',
 '__lt__',
 '__module__',
 '__ne__',
 '__new__',
 '__orig_bases__',
 '__parameters__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__setattr__',
 '__sizeof__',
 '__slots__',
 '__str__',
 '__subclasshook__',
 '__weakref__',
 '_abc_impl',
 '_call_java',
 '_checkThresholdConsistency',
 '_copyValues',
 '_copy_params',
 '_create_from_java_class',
 '_create_params_from_java',
 '_defaultParamMap',
 '_dummy',
 '_empty_java_param_map',
 '_from_java',
 '_is_protocol',
 '_java_obj',
 '_make_java_param_pair',
 '_new_java_array',
 '_new_java_obj',
 '_paramMap',
 '_params',
 '_randomUID',
 '_resetUid',
 '_resolveParam',
 '_set',
 '_setDefault',
 '_shouldOwn',
 '_testOwnParam',
```