loan-credit

August 22, 2024

1 Machine Learning on Big Data using PySpark

---

2 Initiate and Configure Spark

[ ]:
```
# Using ! to execute a command in the command line or terminal
# Using pip3 to interact with the Python package manager for Python 3.x
# Using install to specify that we want to install a package
# Install the PySpark library, which is the Python API for Apache Spark


!pip3 install pyspark
```

```
Requirement already satisfied: pyspark in /usr/local/lib/python3.10/dist-
packages (3.5.2)
Requirement already satisfied: py4j==0.10.9.7 in /usr/local/lib/python3.10/dist-
packages (from pyspark) (0.10.9.7)
```

3 Configure Google Drive

[ ]:
```
from google.colab import drive

# Mounting the Google Drive
drive.mount('/content/drive')
```

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

[ ]:
```
from pyspark.sql import SparkSession

# Creating a SparkSession named 'spark' to interact with Spark
# The 'master' parameter is set to "local[*]", which means Spark will run in
 ↪local mode using all available cores
# The 'appName' parameter is set to 'Loan Credit' to give a name
```

```python
# The 'getOrCreate()' method ensures that if an existing SparkSession is␣
 ↪available, it will be reused; otherwise, a new one will be created


spark = SparkSession.builder \
        .master("local[*]") \
        .appName('Loan Credit') \
        .config("spark.driver.memory", "14g") \
        .config("spark.kryoserializer.buffer.max", "1g") \
        .getOrCreate()
```

[ ]:
```python
# load spark dataframe
sdf = spark.read.csv('/content/drive/MyDrive/Loan_credit/loan_credit.
 ↪csv',inferSchema=True, header =True)

# View available details in this spark dataframe
sdf.describe().show()
```

```
+-------+------------------+-------------------+------------------+-----------+-
-----------+---------------+------------------+-----------------+---------------
--+------------------+-----------------+---------------+----------------+-------
-------------+------------------+-----------------+--------------------------+--
-----------------+------------------+------------------+-------------------+----
--------------+--------------------+------------------+-------------------+-----
--------------+-------------------+------------------+--------------------+-----
-------------+--------------------+---------------------------+-----------------
---------+-----------------------+--------------------------+-------------------
-------+---------------------------+----------------------+---------------------
-+-----------------------+-----------------+--------------------+---------------
-----+--------------------+------------------+-------------------+--------------
-------------+------------------+--------------------+-------------------+------
-------------+-------------------+-------------------+-------------------+------
--------------+-------------------+-----------------------+-------------------+-
------------------+-------------------+----------------------------+------------
-------+-------------------+-------------------+-------------------+------------
-------+-------------------+-------------------+---------------------+----------
---------+------------------------+--------------------+-------------------+----
---------------+----------------------------+-------------------+---------------
----+-------------------+-------------------+------------------+----------------
---+-------------------+---------------------+-------------------+--------------
----------+--------------------+--------------------+--------------+------------
-------+------------------+-------------------+------------------------+--------
----------------+------------------------+------------------------+-------------
---------+--------------------+------------------+--------------------+---------
-----------+------------------+--------------------+-------------------+--------
-----------+--------------------+--------------------+--------------------+-----
---------------+--------------------+--------------------+--------------------+-
```

```
-------------------+--------------------+--------------------+------------------
--+--------------------+--------------------------+-------------------------+---
-----------------------+-------------------------+-------------------------+----
----------------------+
|summary|        SK_ID_CURR|
TARGET|NAME_CONTRACT_TYPE|CODE_GENDER|FLAG_OWN_CAR|FLAG_OWN_REALTY|
CNT_CHILDREN| AMT_INCOME_TOTAL|       AMT_CREDIT|       AMT_ANNUITY|
AMT_GOODS_PRICE|NAME_TYPE_SUITE|NAME_INCOME_TYPE| NAME_EDUCATION_TYPE|NAME_FAMIL
Y_STATUS|NAME_HOUSING_TYPE|REGION_POPULATION_RELATIVE|         DAYS_BIRTH|
DAYS_EMPLOYED| DAYS_REGISTRATION|    DAYS_ID_PUBLISH|       OWN_CAR_AGE|
FLAG_MOBIL|    FLAG_EMP_PHONE|    FLAG_WORK_PHONE|   FLAG_CONT_MOBILE|
FLAG_PHONE|        FLAG_EMAIL|     OCCUPATION_TYPE|   CNT_FAM_MEMBERS|REGION_RAT
ING_CLIENT|REGION_RATING_CLIENT_W_CITY|WEEKDAY_APPR_PROCESS_START|HOUR_APPR_PROC
ESS_START|REG_REGION_NOT_LIVE_REGION|REG_REGION_NOT_WORK_REGION|LIVE_REGION_NOT_
WORK_REGION|REG_CITY_NOT_LIVE_CITY|REG_CITY_NOT_WORK_CITY|LIVE_CITY_NOT_WORK_CIT
Y|ORGANIZATION_TYPE|        EXT_SOURCE_1|        EXT_SOURCE_2|
EXT_SOURCE_3|    APARTMENTS_AVG|   BASEMENTAREA_AVG|YEARS_BEGINEXPLUATATION_AVG|
YEARS_BUILD_AVG|      COMMONAREA_AVG|      ELEVATORS_AVG|      ENTRANCES_AVG|
FLOORSMAX_AVG|      FLOORSMIN_AVG|       LANDAREA_AVG|LIVINGAPARTMENTS_AVG|
LIVINGAREA_AVG|NONLIVINGAPARTMENTS_AVG|  NONLIVINGAREA_AVG|    APARTMENTS_MODE|
BASEMENTAREA_MODE|YEARS_BEGINEXPLUATATION_MODE|   YEARS_BUILD_MODE|
COMMONAREA_MODE|     ELEVATORS_MODE|     ENTRANCES_MODE|     FLOORSMAX_MODE|
FLOORSMIN_MODE|      LANDAREA_MODE|LIVINGAPARTMENTS_MODE|
LIVINGAREA_MODE|NONLIVINGAPARTMENTS_MODE|  NONLIVINGAREA_MODE|
APARTMENTS_MEDI|  BASEMENTAREA_MEDI|YEARS_BEGINEXPLUATATION_MEDI|
YEARS_BUILD_MEDI|    COMMONAREA_MEDI|     ELEVATORS_MEDI|     ENTRANCES_MEDI|
FLOORSMAX_MEDI|     FLOORSMIN_MEDI|      LANDAREA_MEDI|LIVINGAPARTMENTS_MEDI|
LIVINGAREA_MEDI|NONLIVINGAPARTMENTS_MEDI|  NONLIVINGAREA_MEDI|
FONDKAPREMONT_MODE|HOUSETYPE_MODE|     TOTALAREA_MODE|WALLSMATERIAL_MODE|EMERGEN
CYSTATE_MODE|OBS_30_CNT_SOCIAL_CIRCLE|DEF_30_CNT_SOCIAL_CIRCLE|OBS_60_CNT_SOCIAL
_CIRCLE|DEF_60_CNT_SOCIAL_CIRCLE|DAYS_LAST_PHONE_CHANGE|     FLAG_DOCUMENT_2|
FLAG_DOCUMENT_3|     FLAG_DOCUMENT_4|     FLAG_DOCUMENT_5|   FLAG_DOCUMENT_6|
FLAG_DOCUMENT_7|    FLAG_DOCUMENT_8|    FLAG_DOCUMENT_9|    FLAG_DOCUMENT_10|
FLAG_DOCUMENT_11|    FLAG_DOCUMENT_12|    FLAG_DOCUMENT_13|    FLAG_DOCUMENT_14|
FLAG_DOCUMENT_15|    FLAG_DOCUMENT_16|    FLAG_DOCUMENT_17|    FLAG_DOCUMENT_18|
FLAG_DOCUMENT_19|    FLAG_DOCUMENT_20|    FLAG_DOCUMENT_21|AMT_REQ_CREDIT_BUREAU
_HOUR|AMT_REQ_CREDIT_BUREAU_DAY|AMT_REQ_CREDIT_BUREAU_WEEK|AMT_REQ_CREDIT_BUREAU
_MON|AMT_REQ_CREDIT_BUREAU_QRT|AMT_REQ_CREDIT_BUREAU_YEAR|
+-------+------------------+-------------------+------------------+-----------+-
-----------+---------------+------------------+-----------------+---------------
--+------------------+-----------------+---------------+----------------+-------
-------------+------------------+-----------------+--------------------------+--
-----------------+------------------+------------------+-------------------+----
--------------+--------------------+------------------+-------------------+-----
--------------+-------------------+------------------+--------------------+-----
-------------+--------------------+---------------------------+-----------------
---------+-----------------------+--------------------------+-------------------
-------+---------------------------+----------------------+---------------------
```

```
-+-----------------------+-----------------+--------------------+---------------
-----+--------------------+------------------+-------------------+--------------
-------------+------------------+--------------------+-------------------+------
-------------+-------------------+-------------------+-------------------+------
--------------+-------------------+-----------------------+-------------------+-
------------------+-------------------+----------------------------+------------
-------+-------------------+-------------------+-------------------+------------
-------+-------------------+-------------------+---------------------+----------
---------+------------------------+--------------------+-------------------+----
---------------+----------------------------+-------------------+---------------
----+-------------------+-------------------+------------------+----------------
---+-------------------+---------------------+-------------------+--------------
----------+--------------------+--------------------+--------------+------------
-------+------------------+-------------------+------------------------+--------
----------------+------------------------+------------------------+-------------
---------+--------------------+------------------+--------------------+---------
-----------+------------------+--------------------+-------------------+--------
-----------+--------------------+--------------------+--------------------+-----
---------------+--------------------+--------------------+--------------------+-
-------------------+--------------------+--------------------+------------------
--+--------------------+--------------------------+-------------------------+---
-----------------------+-------------------------+-------------------------+----
----------------------+
|  count|           307511|             307511|             307511|     307511|
307511|         307511|             307511|            307511|            307511|
307499|           307233|         306219|           307511|              307511|
307511|           307511|                      307511|             307511|
307511|            307511|             307511|             104582|
307511|            307511|             307511|              307511|
307511|            307511|              211120|             307509|
307511|                     307511|                      307511|
307511|                    307511|                      307511|
307511|                307511|                307511|                307511|
307511|              134133|             306851|              246546|
151450|             127568|                  157504|             103023|
92646|             143620|             152683|             154491|
98869|             124921|               97312|             153161|
93997|             137829|             151450|             127568|
157504|             103023|              92646|             143620|
152683|             154491|              98869|             124921|
97312|             153161|                  93997|             137829|
151450|             127568|                   157504|             103023|
92646|             143620|             152683|            154491|
98869|             124921|                97312|             153161|
93997|              137829|               97216|        153214|
159080|            151170|             161756|                  306490|
306490|                  306490|                   306490|               307510|
307511|            307511|              307511|              307511|
```

```
307511|              307511|             307511|             307511|
307511|              307511|              307511|              307511|
307511|              307511|              307511|              307511|
307511|              307511|              307511|              307511|
265992|                  265992|                   265992|
265992|                  265992|                   265992|
|   mean|278180.51857657125|0.08072881945686496|             NULL|       NULL|
NULL|          NULL|0.4170517477423572|168797.9192969845|599025.9997057016|2710
8.573909183444|538396.2074288895|          NULL|           NULL|
NULL|             NULL|            NULL|
0.020868112057780934|-16036.995066843137|
63815.04590404896|-4986.120327538419|-2994.2023732484367|12.061090818687727|
0.9999967480838083|0.8198893698111612| 0.1993684778755882|
0.9981334001060125|0.28106636835755466|0.0567199222141647|            NULL|
2.152665450442101|  2.0524631639193394|        2.031520823645333|
NULL|     12.063418869568894|      0.015144173704355291|
0.05076891558350758|       0.04065870814377372|   0.07817281333025486|
0.23045354475124466|    0.17955455252007246|            NULL|
0.5021298056566625|  0.5143926741308462|
0.5108529061799736|0.1174404991746445|0.08844221905179989|        0.97773485816
23645|0.7524714325927212|0.044620715411350736|0.07894151232418994|0.149724670068
02942|0.22628190703666962|0.23189350049054858|0.06633318417239682|
0.10077477495067477|0.10739901933259748|   0.008808672617211424|0.02835775707579
679|0.11423100693298159|0.08754321224758546|         0.977065372942917| 0.7596
373227337593|0.04255313775014573|0.07448973610917839|0.14519265864566963|0.22231
504747850783|0.22805849255074997|0.06495768445657664|
0.10564485674942427|0.10597505043712181|    0.008076387544283543|0.0270223196859
87662|0.11784992076592789|0.08795485466574696|         0.9777522640695162| 0.75
57462721916565|0.04459510178529025|0.07807784431137849|0.14921278072867808|0.225
8965900926409|0.23162493804933906|0.06716874904939923|
0.101954473240714|0.10860673604899407|
0.008651013330212998|0.028235920597261845|             NULL|
NULL|0.10254666268544141|             NULL|             NULL|
1.4222454239942575|      0.1434206662533851|      1.4052921791901856|
0.10004894123788705|    -962.8587883320868|4.227491049100682…|0.71002338127741
77|8.129790479039775E-5|0.015114906458630749|0.0880553866365756|1.91863055305338
6…|0.08137595077899652|0.00389579559755586|2.276341334131137E-5|0.003912055178
513939|6.503832383231819E-6|0.003525077151711646|0.002936480321029…|0.00120971
2823281…|0.009928100133003373|2.666571277125046E-4|0.008129790479039774|5.9510
06630657115E-4|5.072989258920819E-4|3.349473677364387E-4|
0.006402448193930645|     0.007000210532647…|        0.0343619356973142|
0.26739526000781977|      0.26547414959848414|         1.899974435321363|
| stddev|102790.17534842492| 0.2724186456483937|             NULL|       NULL|
NULL|          NULL|0.7221213844376252|
237123.146278856|402490.7769958547|14493.737315118331|369446.4605400569|
NULL|            NULL|                NULL|              NULL|            NULL|
0.01383128012270469|  4363.988631785563|141275.76651872732|3522.8863209630713| 1
509.4504190030234|11.944811582242762|0.001803307015351…|0.3842801989387643|0.3
```

```
9952622815022637|0.04316389414243243|0.44952054685675824|0.2313070397227074|
NULL|0.9106815691792965|    0.50903390281568|         0.5027370329147685|
NULL|      3.265832255437871|       0.12212647628215208|
0.21952582879696073|         0.19749861882842362|    0.2684437723734045|
0.4211238359138974|     0.3838166153855962|             NULL|
0.21106224927392478| 0.19106015498493661|
0.1948443644637489|0.1082402913003223|0.08243815873568511|
0.05922331435836284|0.1132799266322471|
0.07603574505040939|0.13457600110034426|0.10004912076035916|
0.1446406995480039|0.16138028880013738|0.08118364070179362|
0.09257613396049744|0.11056452318371332|
0.04773166205034792|0.06952318332123608|0.10793603908753294|0.08430717486924556|
0.06457543708048012|0.11011102734194829|0.07444452253839157|0.13225614415050654|
0.10097698816024664|0.14370940659531573|0.16115977149547595|0.08175027780843545|
0.09788044657879377| 0.1118452658778338|     0.04627626621983563|
0.07025385904394457|0.10907590600115318|0.08217874951463422|        0.0598973185
05119646|0.11206630964404388|0.07614426224091464|0.13446714769067492|0.100368394
49763224|0.1450670259193515|0.16193354145715627|0.08216701028007206|
0.09364233271153832|0.11226025867534756|     0.04741472790780271|
0.07016648150682483|                NULL|          NULL|0.10746232414961876|
NULL|                NULL|       2.400988746109006|       0.446698429381529|
2.379803351979381|       0.3622908039755736|
826.8084870406562|0.006501789045489776|0.4537519684327391|0.009016183216550884|
0.12201022281354136|  0.28337589286299|0.013850157677017406|0.27341204894451293|
0.06229471080039372|0.004771055354069212|
0.06242406326684522|0.002550257091597865| 0.05926771807375312|
0.05410976737642885| 0.03475993882769264|
0.09914416233784908|0.016327488741596615|
0.08979823610939562|0.024387465065862254|
0.02251762026844612|0.018298531822437628|       0.08384912844747722|
0.1107574063243545|       0.20468487581282466|       0.9160023961526181|
0.7940556483207578|        1.8692949981815572|
|    min|            100002|                  0|        Cash loans|          F|
N|              N|                 0|          25650.0|          45000.0|
1615.5|          40500.0|       Children|     Businessman|     Academic degree|
Civil marriage|  Co-op apartment|                   2.9E-4|             -25229|
-17912|          -24672.0|            -7197|              0.0|
0|                 0|                   0|                  0|
0|                 0|         Accountants|              1.0|
1|                        1|                    FRIDAY|
0|                        0|                         0|
0|                    0|                    0|                      0|
Advertising|0.014568132412445587|8.173616518884397E-8|5.272652387098817E-4|
0.0|                0.0|                          0.0|                0.0|
0.0|                0.0|                 0.0|                 0.0|
0.0|                0.0|                  0.0|                 0.0|
0.0|                0.0|                 0.0|                 0.0|
0.0|                0.0|                 0.0|                 0.0|
```

```
0.0|              0.0|              0.0|              0.0|
0.0|              0.0|                  0.0|              0.0|
0.0|              0.0|                     0.0|              0.0|
0.0|              0.0|              0.0|             0.0|
0.0|              0.0|                0.0|              0.0|
0.0|               0.0|       not specified|block of flats|
0.0|           Block|             No|                   0.0|
0.0|                  0.0|                    0.0|              -4292.0|
0|             0|               0|               0|
0|              0|               0|              0|
0|              0|                0|               0|
0|              0|                0|               0|
0|              0|                0|               0|
0.0|                  0.0|                    0.0|
0.0|                  0.0|                    0.0|
|    max|            456255|                  1|   Revolving loans|        XNA|
Y|           Y|                19|           1.17E8|       4050000.0|
258025.5|        4050000.0|  Unaccompanied|         Working|Secondary /
secon…|              Widow|     With parents|                0.072508|
-7489|           365243|           0.0|               0|
91.0|                 1|              1|               1|
1|              1|               1|Waiters/barmen staff|
20.0|                3|                    3|
WEDNESDAY|                  23|                     1|
1|                     1|                  1|                   1|
1|          XNA|  0.9626927705613059|  0.8549996664047012|
0.8960095494948396|           1.0|               1.0|
1.0|             1.0|              1.0|              1.0|
1.0|              1.0|              1.0|              1.0|
1.0|              1.0|                 1.0|              1.0|
1.0|              1.0|                     1.0|              1.0|
1.0|              1.0|              1.0|              1.0|
1.0|              1.0|               1.0|              1.0|
1.0|               1.0|              1.0|              1.0|
1.0|              1.0|              1.0|              1.0|
1.0|             1.0|              1.0|              1.0|
1.0|              1.0|                  1.0|              1.0|reg oper
spec acc…|terraced house|              1.0|             Wooden|
Yes|                 348.0|                    34.0|                344.0|
24.0|                0.0|                  1|               1|
1|              1|              1|               1|
1|             1|               1|               1|
1|              1|               1|               1|
1|              1|               1|               1|
1|              1|                  4.0|                  9.0|
8.0|                 27.0|                  261.0|
25.0|
+-------+-----------------+-----------------+----------------+----------+-
```

```
-----------+---------------+------------------+-----------------+---------------
--+------------------+-----------------+---------------+----------------+-------
-------------+------------------+-----------------+--------------------------+--
-----------------+------------------+------------------+-------------------+----
--------------+--------------------+------------------+-------------------+-----
--------------+-------------------+------------------+--------------------+-----
-------------+--------------------+---------------------------+-----------------
---------+-----------------------+--------------------------+-------------------
-------+---------------------------+----------------------+---------------------
-+-----------------------+-----------------+--------------------+---------------
-----+--------------------+------------------+-------------------+--------------
-------------+------------------+--------------------+-------------------+------
-------------+-------------------+-------------------+-------------------+------
--------------+-------------------+-----------------------+-------------------+-
------------------+-------------------+----------------------------+------------
-------+-------------------+-------------------+-------------------+------------
-------+-------------------+-------------------+---------------------+----------
---------+------------------------+--------------------+-------------------+----
---------------+----------------------------+-------------------+---------------
----+-------------------+-------------------+------------------+----------------
---+-------------------+---------------------+-------------------+--------------
----------+--------------------+--------------------+--------------+------------
-------+------------------+-------------------+------------------------+--------
----------------+------------------------+------------------------+-------------
---------+--------------------+------------------+--------------------+---------
-----------+------------------+--------------------+-------------------+--------
-----------+--------------------+--------------------+--------------------+-----
---------------+--------------------+--------------------+--------------------+-
-------------------+--------------------+--------------------+------------------
--+--------------------+--------------------------+-------------------------+---
-----------------------+-------------------------+-------------------------+----
----------------------+
```

[ ]:
```
selected_df = sdf.select(
    "SK_ID_CURR",
    "TARGET",
    "NAME_CONTRACT_TYPE",
    "CODE_GENDER",
    "FLAG_OWN_CAR",
    "FLAG_OWN_REALTY",
    "CNT_CHILDREN",
    "AMT_INCOME_TOTAL",
    "AMT_CREDIT",
    "AMT_ANNUITY",
    "AMT_GOODS_PRICE",
    "NAME_INCOME_TYPE",
```

```
    "NAME_EDUCATION_TYPE",
    "NAME_FAMILY_STATUS",
    "NAME_HOUSING_TYPE",
    "DAYS_BIRTH",
    "DAYS_EMPLOYED",
    "DAYS_REGISTRATION",
    "DAYS_ID_PUBLISH",
    "OCCUPATION_TYPE",
    "EXT_SOURCE_1",
    "EXT_SOURCE_2",
    "EXT_SOURCE_3",
    "REGION_RATING_CLIENT",
    "REGION_RATING_CLIENT_W_CITY"
)
```

[ ]:
```
selected_df.show(5)
```

```
+----------+------+------------------+-----------+------------+---------------+-
-----------+----------------+----------+-----------+---------------+------------
----+--------------------+--------------------+-----------------+----------+----
---------+-----------------+---------------+---------------+-------------------+
------------------+-------------------+---------------------+-------------------
-------+
|SK_ID_CURR|TARGET|NAME_CONTRACT_TYPE|CODE_GENDER|FLAG_OWN_CAR|FLAG_OWN_REALTY|C
NT_CHILDREN|AMT_INCOME_TOTAL|AMT_CREDIT|AMT_ANNUITY|AMT_GOODS_PRICE|NAME_INCOME_
TYPE| NAME_EDUCATION_TYPE|   NAME_FAMILY_STATUS|NAME_HOUSING_TYPE|DAYS_BIRTH|DAYS
_EMPLOYED|DAYS_REGISTRATION|DAYS_ID_PUBLISH|OCCUPATION_TYPE|       EXT_SOURCE_1|
EXT_SOURCE_2|
EXT_SOURCE_3|REGION_RATING_CLIENT|REGION_RATING_CLIENT_W_CITY|
+----------+------+------------------+-----------+------------+---------------+-
-----------+----------------+----------+-----------+---------------+------------
----+--------------------+--------------------+-----------------+----------+----
---------+-----------------+---------------+---------------+-------------------+
------------------+-------------------+---------------------+-------------------
-------+
|    100002|     1|        Cash loans|          M|           N|              Y|
0|         202500.0|  406597.5|    24700.5|       351000.0|
Working|Secondary / secon…|Single / not married|House / apartment|     -9461|
-637|           -3648.0|          -2120|
Laborers|0.08303696739132256|0.2629485927471776|0.13937578009978951|
2|                           2|
|    100003|     0|        Cash loans|          F|           N|              N|
0|         270000.0| 1293502.5|    35698.5|      1129500.0|   State servant|
Higher education|             Married|House / apartment|    -16765|
-1188|            -1186.0|           -291|     Core staff|
0.3112673113812225|0.6222457752555098|               NULL|                  1|
1|
```

```
|    100004|     0|   Revolving loans|          M|            Y|              Y|
0|          67500.0|  135000.0|     6750.0|       135000.0|
Working|Secondary / secon…|Single / not married|House / apartment|    -19046|
-225|            -4260.0|             -2531|       Laborers|
NULL|0.5559120833904428| 0.7295666907060153|                  2|
2|
|    100006|     0|        Cash loans|          F|            N|              Y|
0|         135000.0|  312682.5|    29686.5|       297000.0|
Working|Secondary / secon…|      Civil marriage|House / apartment|    -19005|
-3039|            -9833.0|             -2437|       Laborers|
NULL|0.6504416904014653|               NULL|                  2|
2|
|    100007|     0|        Cash loans|          M|            N|              Y|
0|         121500.0|  513000.0|    21865.5|       513000.0|
Working|Secondary / secon…|Single / not married|House / apartment|    -19932|
-3038|            -4311.0|             -3458|     Core staff|
NULL|0.3227382869704046|               NULL|                  2|
2|
+----------+------+------------------+-----------+------------+---------------+-
-----------+----------------+----------+-----------+---------------+------------
----+--------------------+--------------------+-----------------+----------+----
---------+-----------------+----------------+----------------+-------------------+
-------------------+-------------------+---------------------+--------------------
-------+
only showing top 5 rows
```

[ ]:
```python
from pyspark.sql.functions import abs, round, col

# Handling Negative values
converted_df = selected_df.withColumn("AGE_YEARS", round(abs(col("DAYS_BIRTH"))␣
↪/ 365, 2)) \
                             .withColumn("DAYS_EMPLOYED_POSITIVE",␣
↪abs(col("DAYS_EMPLOYED"))) \
                             .withColumn("DAYS_REGISTRATION_POSITIVE",␣
↪abs(col("DAYS_REGISTRATION"))) \
                             .withColumn("DAYS_ID_PUBLISH_POSITIVE",␣
↪abs(col("DAYS_ID_PUBLISH"))) \
                             .drop("DAYS_BIRTH", "DAYS_EMPLOYED",␣
↪"DAYS_REGISTRATION", "DAYS_ID_PUBLISH")
```

[ ]:
```python
converted_df.show(5)
```

```
+----------+------+------------------+-----------+------------+---------------+-
-----------+----------------+----------+-----------+---------------+------------
----+--------------------+--------------------+-----------------+---------------
+-------------------+------------------+-------------------+--------------------
```

```
+--------------------------+---------+---------------------+-----------------
--------+-----------------------+
|SK_ID_CURR|TARGET|NAME_CONTRACT_TYPE|CODE_GENDER|FLAG_OWN_CAR|FLAG_OWN_REALTY|C
NT_CHILDREN|AMT_INCOME_TOTAL|AMT_CREDIT|AMT_ANNUITY|AMT_GOODS_PRICE|NAME_INCOME_
TYPE| NAME_EDUCATION_TYPE|
NAME_FAMILY_STATUS|NAME_HOUSING_TYPE|OCCUPATION_TYPE|       EXT_SOURCE_1|
EXT_SOURCE_2|       EXT_SOURCE_3|REGION_RATING_CLIENT|REGION_RATING_CLIENT_W_CIT
Y|AGE_YEARS|DAYS_EMPLOYED_POSITIVE|DAYS_REGISTRATION_POSITIVE|DAYS_ID_PUBLISH_PO
SITIVE|
+----------+------+------------------+-----------+------------+---------------+-
-----------+----------------+----------+-----------+---------------+-----------
----+-------------------+--------------------+-----------------+---------------
+-------------------+------------------+-------------------+-------------------
+--------------------------+---------+---------------------+-----------------
--------+-----------------------+
|    100002|     1|        Cash loans|          M|           N|              Y|
0|        202500.0|  406597.5|    24700.5|       351000.0|
Working|Secondary / secon…|Single / not married|House / apartment|
Laborers|0.08303696739132256|0.2629485927471776|0.13937578009978951|
2|                         2|    25.92|                  637|
3648.0|                     2120|
|    100003|     0|        Cash loans|          F|           N|              N|
0|        270000.0| 1293502.5|    35698.5|      1129500.0|   State servant|
Higher education|            Married|House / apartment|     Core staff|
0.3112673113812225|0.6222457752555098|              NULL|                 1|
1|    45.93|                 1188|                  1186.0|
291|
|    100004|     0|   Revolving loans|          M|           Y|              Y|
0|         67500.0|  135000.0|     6750.0|       135000.0|
Working|Secondary / secon…|Single / not married|House / apartment|
Laborers|              NULL|0.5559120833904428| 0.7295666907060153|
2|                         2|    52.18|                  225|
4260.0|                     2531|
|    100006|     0|        Cash loans|          F|           N|              Y|
0|        135000.0|  312682.5|    29686.5|       297000.0|
Working|Secondary / secon…|      Civil marriage|House / apartment|
Laborers|              NULL|0.6504416904014653|              NULL|
2|                         2|    52.07|                 3039|
9833.0|                     2437|
|    100007|     0|        Cash loans|          M|           N|              Y|
0|        121500.0|  513000.0|    21865.5|       513000.0|
Working|Secondary / secon…|Single / not married|House / apartment|     Core
staff|              NULL|0.3227382869704046|              NULL|
2|                         2|    54.61|                 3038|
4311.0|                     3458|
+----------+------+------------------+-----------+------------+---------------+-
-----------+----------------+----------+-----------+---------------+-----------
----+-------------------+--------------------+-----------------+---------------
```

```
+--------------------+-------------------+--------------------+--------------------
+--------------------------+---------+----------------------+-------------------
--------+-------------------------+
only showing top 5 rows
```

[ ]:
```
from pyspark.sql.functions import isnan, when, count, col

# Check for missing values in each column
counts_missing = converted_df.select([count(when(col(c).isNull() |␣
↪isnan(col(c)), c)).alias(c) for c in converted_df.columns])
counts_missing.show()

# Check for missing values in any row
total_count = converted_df.rdd.map(lambda row: sum([1 for x in row if x ==␣
↪None])).sum()
print("Total missing values in DataFrame: {}".format(total_count))
```

```
+-----------+------+------------------+-----------+------------+---------------+-
-----------+----------------+----------+-----------+---------------+------------
----+-------------------+-------------------+------------------+---------------+--
----------+------------+------------+---------------------+----------------------
-----+---------+----------------------+--------------------------+--------------
----------+
|SK_ID_CURR|TARGET|NAME_CONTRACT_TYPE|CODE_GENDER|FLAG_OWN_CAR|FLAG_OWN_REALTY|C
NT_CHILDREN|AMT_INCOME_TOTAL|AMT_CREDIT|AMT_ANNUITY|AMT_GOODS_PRICE|NAME_INCOME_
TYPE|NAME_EDUCATION_TYPE|NAME_FAMILY_STATUS|NAME_HOUSING_TYPE|OCCUPATION_TYPE|EX
T_SOURCE_1|EXT_SOURCE_2|EXT_SOURCE_3|REGION_RATING_CLIENT|REGION_RATING_CLIENT_W
_CITY|AGE_YEARS|DAYS_EMPLOYED_POSITIVE|DAYS_REGISTRATION_POSITIVE|DAYS_ID_PUBLIS
H_POSITIVE|
+-----------+------+------------------+-----------+------------+---------------+-
-----------+----------------+----------+-----------+---------------+------------
----+-------------------+-------------------+------------------+---------------+--
----------+------------+------------+---------------------+----------------------
-----+---------+----------------------+--------------------------+--------------
----------+
|         0|     0|                 0|          0|           0|              0|
0|               0|         0|         12|            278|                 0|
0|                  0|                 0|           96391|         173378|
660|        60965|                   0|                         0|         0|
0|                         0|                          0|
+-----------+------+------------------+-----------+------------+---------------+-
-----------+----------------+----------+-----------+---------------+------------
----+-------------------+-------------------+------------------+---------------+--
----------+------------+------------+---------------------+----------------------
-----+---------+----------------------+--------------------------+--------------
----------+
```

```
Total missing values in DataFrame: 331684
```

[ ]:
```python
# Handling the Missing values
from pyspark.sql.functions import mean

# We can drop 2 columns with large amount of missing values
sdf_cleaned = converted_df.drop("OCCUPATION_TYPE", "EXT_SOURCE_1")

# Impute missing values in numerical columns with the mean
mean_values = {col: sdf_cleaned.agg(mean(col)).first()[0] for col in␣
↪["EXT_SOURCE_2", "EXT_SOURCE_3", "AMT_ANNUITY", "AMT_GOODS_PRICE"]}
imputed_df = sdf_cleaned.na.fill(mean_values)
```

[ ]:
```python
counts_missing = imputed_df.select([count(when(col(c).isNull() | isnan(col(c)),␣
↪c)).alias(c) for c in imputed_df.columns])
counts_missing.show()

# Check for missing values in any row
total_count = imputed_df.rdd.map(lambda row: sum([1 for x in row if x ==␣
↪None])).sum()
print("Total missing values in DataFrame: {}".format(total_count))
```

```
+----------+------+------------------+-----------+------------+---------------+-
-----------+----------------+----------+-----------+---------------+------------
----+-------------------+------------------+-----------------+------------+-----
-------+--------------------+---------------------------+---------+-------------
---------+--------------------------+------------------------+
|SK_ID_CURR|TARGET|NAME_CONTRACT_TYPE|CODE_GENDER|FLAG_OWN_CAR|FLAG_OWN_REALTY|C
NT_CHILDREN|AMT_INCOME_TOTAL|AMT_CREDIT|AMT_ANNUITY|AMT_GOODS_PRICE|NAME_INCOME_
TYPE|NAME_EDUCATION_TYPE|NAME_FAMILY_STATUS|NAME_HOUSING_TYPE|EXT_SOURCE_2|EXT_S
OURCE_3|REGION_RATING_CLIENT|REGION_RATING_CLIENT_W_CITY|AGE_YEARS|DAYS_EMPLOYED
_POSITIVE|DAYS_REGISTRATION_POSITIVE|DAYS_ID_PUBLISH_POSITIVE|
+----------+------+------------------+-----------+------------+---------------+-
-----------+----------------+----------+-----------+---------------+------------
----+-------------------+------------------+-----------------+------------+-----
-------+--------------------+---------------------------+---------+-------------
---------+--------------------------+------------------------+
|         0|     0|                 0|          0|           0|              0|
0|               0|         0|          0|              0|                0|
0|                  0|                 0|                0|           0|
0|                   0|                          0|        0|
0|                         0|
+----------+------+------------------+-----------+------------+---------------+-
-----------+----------------+----------+-----------+---------------+------------
----+-------------------+------------------+-----------------+------------+-----
-------+--------------------+---------------------------+---------+-------------
---------+--------------------------+------------------------+
```

```
Total missing values in DataFrame: 0
```

[ ]:
```python
# Checking the Schema along with the types
imputed_df.printSchema()
```

```
root
 |-- SK_ID_CURR: integer (nullable = true)
 |-- TARGET: integer (nullable = true)
 |-- NAME_CONTRACT_TYPE: string (nullable = true)
 |-- CODE_GENDER: string (nullable = true)
 |-- FLAG_OWN_CAR: string (nullable = true)
 |-- FLAG_OWN_REALTY: string (nullable = true)
 |-- CNT_CHILDREN: integer (nullable = true)
 |-- AMT_INCOME_TOTAL: double (nullable = true)
 |-- AMT_CREDIT: double (nullable = true)
 |-- AMT_ANNUITY: double (nullable = false)
 |-- AMT_GOODS_PRICE: double (nullable = false)
 |-- NAME_INCOME_TYPE: string (nullable = true)
 |-- NAME_EDUCATION_TYPE: string (nullable = true)
 |-- NAME_FAMILY_STATUS: string (nullable = true)
 |-- NAME_HOUSING_TYPE: string (nullable = true)
 |-- EXT_SOURCE_2: double (nullable = false)
 |-- EXT_SOURCE_3: double (nullable = false)
 |-- REGION_RATING_CLIENT: integer (nullable = true)
 |-- REGION_RATING_CLIENT_W_CITY: integer (nullable = true)
 |-- AGE_YEARS: double (nullable = true)
 |-- DAYS_EMPLOYED_POSITIVE: integer (nullable = true)
 |-- DAYS_REGISTRATION_POSITIVE: double (nullable = true)
 |-- DAYS_ID_PUBLISH_POSITIVE: integer (nullable = true)
```

[ ]:
```python
from pyspark.ml.feature import StringIndexer
from pyspark.sql.types import StringType

# convert all the string features into numerical values
string_features=[field.name for field in imputed_df.schema.fields if␣
 ↪isinstance(field.dataType, StringType)]

for feature in string_features:
    indexer = StringIndexer(inputCol=feature, outputCol=feature+"_numeric")
    imputed_df = indexer.fit(imputed_df).transform(imputed_df)
imputed_df = imputed_df.drop(*string_features)

imputed_df.show()
```

```
+----------+------+------------+----------------+----------+-----------+--------
-------+-------------------+-------------------+--------------------+-----------
```

```
----------------+---------+----------------------+--------------------------+---
---------------------+--------------------------+-------------------+-----------
---------+-----------------------+------------------------+----------------------
------+--------------------------+-------------------------+
|SK_ID_CURR|TARGET|CNT_CHILDREN|AMT_INCOME_TOTAL|AMT_CREDIT|AMT_ANNUITY|AMT_GOOD
S_PRICE|      EXT_SOURCE_2|       EXT_SOURCE_3|REGION_RATING_CLIENT|REGION_RATI
NG_CLIENT_W_CITY|AGE_YEARS|DAYS_EMPLOYED_POSITIVE|DAYS_REGISTRATION_POSITIVE|DAY
S_ID_PUBLISH_POSITIVE|NAME_CONTRACT_TYPE_numeric|CODE_GENDER_numeric|FLAG_OWN_CA
R_numeric|FLAG_OWN_REALTY_numeric|NAME_INCOME_TYPE_numeric|NAME_EDUCATION_TYPE_n
umeric|NAME_FAMILY_STATUS_numeric|NAME_HOUSING_TYPE_numeric|
+----------+------+------------+----------------+----------+-----------+--------
-------+------------------+-------------------+--------------------+------------
---------------+---------+----------------------+--------------------------+----
--------------------+--------------------------+-------------------+------------
--------+-----------------------+------------------------+----------------------
-----+--------------------------+-------------------------+
|    100002|     1|           0|        202500.0|  406597.5|    24700.5|
351000.0| 0.2629485927471776|0.13937578009978951|                   2|
2|    25.92|                   637|                    3648.0|
2120|                       0.0|                1.0|                 0.0|
0.0|                     0.0|                        0.0|
1.0|                      0.0|
|    100003|     0|           0|        270000.0| 1293502.5|    35698.5|
1129500.0| 0.6222457752555098| 0.5108529061799736|                   1|
1|    45.93|                  1188|                    1186.0|
291|                       0.0|                0.0|                 0.0|
1.0|                     3.0|                        1.0|
0.0|                      0.0|
|    100004|     0|           0|         67500.0|  135000.0|     6750.0|
135000.0| 0.5559120833904428| 0.7295666907060153|                   2|
2|    52.18|                   225|                    4260.0|
2531|                       1.0|                1.0|                 1.0|
0.0|                     0.0|                        0.0|
1.0|                      0.0|
|    100006|     0|           0|        135000.0|  312682.5|    29686.5|
297000.0| 0.6504416904014653| 0.5108529061799736|                   2|
2|    52.07|                  3039|                    9833.0|
2437|                       0.0|                0.0|                 0.0|
0.0|                     0.0|                        0.0|
2.0|                      0.0|
|    100007|     0|           0|        121500.0|  513000.0|    21865.5|
513000.0| 0.3227382869704046| 0.5108529061799736|                   2|
2|    54.61|                  3038|                    4311.0|
3458|                       0.0|                1.0|                 0.0|
0.0|                     0.0|                        0.0|
1.0|                      0.0|
|    100008|     0|           0|         99000.0|  490495.5|    27517.5|
454500.0| 0.3542247319929012| 0.6212263380626669|                   2|
```

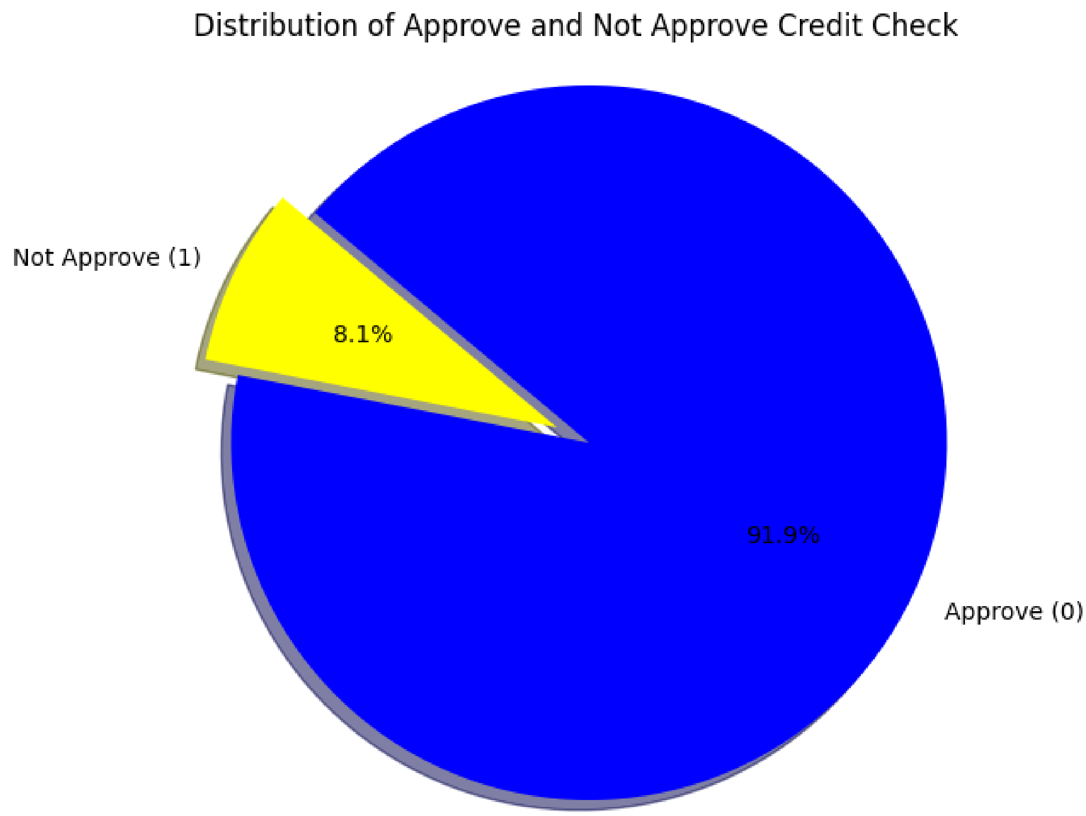
```
2|    46.41|                 1588|                    4970.0|
477|                      0.0|               1.0|                0.0|
0.0|                    3.0|                         0.0|
0.0|                     0.0|
|    100009|     0|           1|        171000.0| 1560726.0|    41301.0|
1395000.0| 0.7239998516953141| 0.4920600938649263|                 2|
2|    37.75|                 3130|                    1213.0|
619|                      0.0|               0.0|                1.0|
0.0|                    1.0|                         1.0|
0.0|                     0.0|
|    100010|     0|           0|        360000.0| 1530000.0|    42075.0|
1530000.0| 0.7142792864482229| 0.5406544504453575|                 3|
3|    51.64|                  449|                    4597.0|
2379|                     0.0|               1.0|                1.0|
0.0|                    3.0|                         1.0|
0.0|                     0.0|
|    100011|     0|           0|        112500.0| 1019610.0|    33826.5|
913500.0|0.20574728800732814| 0.7517237147741489|                 2|
2|    55.07|               365243|                    7427.0|
3514|                     0.0|               0.0|                0.0|
0.0|                    2.0|                         0.0|
0.0|                     0.0|
|    100012|     0|           0|        135000.0|  405000.0|    20250.0|
405000.0| 0.7466436294590924| 0.5108529061799736|                 2|
2|    39.64|                 2019|                   14437.0|
3992|                     1.0|               1.0|                0.0|
0.0|                    0.0|                         0.0|
1.0|                     0.0|
|    100014|     0|           1|        112500.0|  652500.0|    21177.0|
652500.0| 0.6518623334244781|  0.363945238612397|                 2|
2|    27.94|                  679|                    4427.0|
738|                      0.0|               0.0|                0.0|
0.0|                    0.0|                         1.0|
0.0|                     0.0|
|    100015|     0|           0|      38419.155|  148365.0|    10678.5|
135000.0| 0.5551831615131809| 0.6528965519806539|                 2|
2|    55.94|               365243|                    5246.0|
2512|                     0.0|               0.0|                0.0|
0.0|                    2.0|                         0.0|
0.0|                     0.0|
|    100016|     0|           0|         67500.0|   80865.0|     5881.5|
67500.0| 0.7150418188660659| 0.1766525794312139|                 2|
2|    36.82|                 2717|                     311.0|
3227|                     0.0|               0.0|                0.0|
0.0|                    0.0|                         0.0|
0.0|                     0.0|
|    100017|     0|           1|        225000.0|  918468.0|    28966.5|
697500.0| 0.5669066132460429| 0.7700870700124128|                 2|
```

```
2|    38.59|                 3028|                    643.0|
4911|                      0.0|                1.0|                  1.0|
1.0|                    0.0|                         0.0|
0.0|                     0.0|
|    100018|     0|           0|        189000.0|  773680.5|    32778.0|
679500.0| 0.6426562048311103| 0.5108529061799736|                   2|
1|    39.95|                  203|                    615.0|
2056|                      0.0|                0.0|                  0.0|
0.0|                    0.0|                         0.0|
0.0|                     0.0|
|    100019|     0|           0|        157500.0|  299772.0|    20160.0|
247500.0|   0.34663398139668| 0.6785676886853644|                   3|
3|    23.91|                 1157|                   3494.0|
1368|                      0.0|                1.0|                  1.0|
0.0|                    0.0|                         0.0|
1.0|                     3.0|
|    100020|     0|           0|        108000.0|  509602.5|    26149.5|
387000.0| 0.2363778398884225|0.06210303783729682|                   2|
2|    35.43|                 1317|                   6392.0|
3866|                      0.0|                1.0|                  0.0|
1.0|                    0.0|                         0.0|
0.0|                     0.0|
|    100021|     0|           1|         81000.0|  270000.0|    13500.0|
270000.0| 0.6835133461914255| 0.5108529061799736|                   2|
2|    26.78|                  191|                   4143.0|
2427|                      1.0|                0.0|                  0.0|
0.0|                    0.0|                         0.0|
0.0|                     0.0|
|    100022|     0|           0|        112500.0|  157500.0|     7875.0|
157500.0| 0.7064284028871654| 0.5567274263630174|                   1|
1|    48.54|                 7804|                   8751.0|
1259|                      1.0|                0.0|                  0.0|
0.0|                    0.0|                         0.0|
4.0|                     0.0|
|    100023|     0|           1|         90000.0|  544491.0|    17563.5|
454500.0| 0.5866171400119664| 0.4776491548517548|                   2|
2|    31.09|                 2038|                   1021.0|
3964|                      0.0|                0.0|                  0.0|
0.0|                    3.0|                         1.0|
1.0|                     0.0|
+----------+------+------------+----------------+----------+-----------+--------
-------+-------------------+-------------------+--------------------+-----------
----------------+---------+----------------------+--------------------------+---
---------------------+-------------------------+-------------------+-----------
---------+-----------------------+-----------------------+---------------------
------+-------------------------+-------------------------+
only showing top 20 rows
```

4 Advanced Data Analysis and Exploraton

Further data analysis and research have revealed that our dataset suffers from a class imbalance issue, where the minority class "Not Approve" represents only 8.1% of the data, whereas the majority class "Approve" dominates with 91.9%. This skewed distribution is illustrated in the pie chart below.

[ ]:
```python
import matplotlib.pyplot as plt

# Calculate counts of 1s and 0s
once_count = imputed_df.where(imputed_df["TARGET"] == 1).count()
zeros_count = imputed_df.where(imputed_df["TARGET"] == 0).count()

labels = 'Not Approve (1)', 'Approve (0)'
sizes = [once_count, zeros_count]
colors = ['yellow', 'blue']
explode_ratio = (0.1, 0)

# Plot
plt.figure(figsize=(8, 6))
plt.pie(sizes, explode=explode_ratio, labels=labels, colors=colors, autopct='%1.
 ↪1f%%', shadow=True, startangle=140)
plt.axis('equal')
plt.title('Distribution of Approve and Not Approve Credit Check')
plt.show()
# print(sizes)
```



5 Class Imbalance Mitigation: SMOTE

To address the class imbalance issue, we have chosen to employ the under-sampling method, a type of resampling technique. This approach involves reducing the size of the majority class (Approve) to match the size of the minority class (Not Approve), thereby balancing the class distribution. The resulting pie chart, shown below, illustrates the revised class balance after applying under-sampling to the imbalanced dataset.

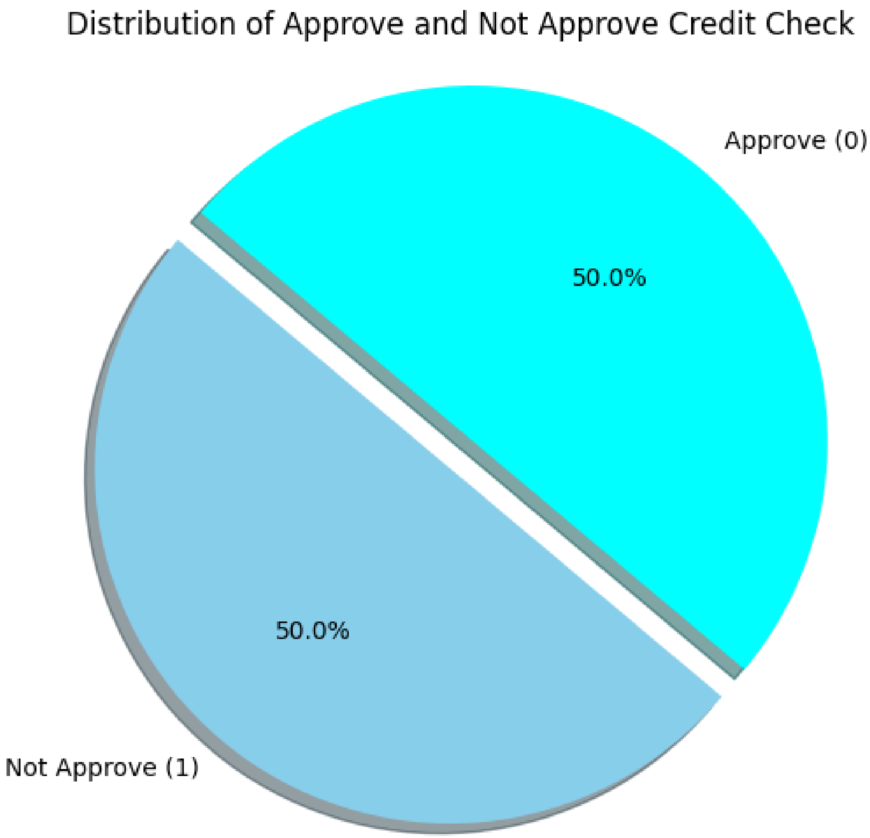
[ ]:
```
from pyspark.sql.functions import col
from imblearn.over_sampling import SMOTE
import pandas as pd

under_fraction = 0.15

# Under-sample the majority class (TARGET = 0)
undersampled_majority_df = imputed_df.where(col("TARGET") == 0).sample(False,␣
↪under_fraction, seed=42)

# Cache the undersampled dataframe to optimize
undersampled_majority_df.cache()
```

```
# Extract minority class (TARGET = 1)
m_df = imputed_df.where(col("TARGET") == 1)

minority_df = undersampled_majority_df.union(m_df).toPandas()

# Separate features and target
X_minority = minority_df.drop(columns=['TARGET'])
y_minority = minority_df['TARGET']

# Apply SMOTE
smote = SMOTE(sampling_strategy=1.0, random_state=42)
X_smote, y_smote = smote.fit_resample(X_minority, y_minority)

# Combine the SMOTE output with the target
smote_df = pd.DataFrame(X_smote, columns=X_minority.columns)
smote_df['TARGET'] = y_smote

# Convert the SMOTE df back to a Spark df
balanced_spark_df = spark.createDataFrame(smote_df)
balanced_df = balanced_spark_df

# Cache the balanced dataframe for better performance
balanced_df.cache()
```

[ ]: DataFrame[SK_ID_CURR: bigint, CNT_CHILDREN: bigint, AMT_INCOME_TOTAL: double, AMT_CREDIT: double, AMT_ANNUITY: double, AMT_GOODS_PRICE: double, EXT_SOURCE_2: double, EXT_SOURCE_3: double, REGION_RATING_CLIENT: bigint, REGION_RATING_CLIENT_W_CITY: bigint, AGE_YEARS: double, DAYS_EMPLOYED_POSITIVE: bigint, DAYS_REGISTRATION_POSITIVE: double, DAYS_ID_PUBLISH_POSITIVE: bigint, NAME_CONTRACT_TYPE_numeric: double, CODE_GENDER_numeric: double, FLAG_OWN_CAR_numeric: double, FLAG_OWN_REALTY_numeric: double, NAME_INCOME_TYPE_numeric: double, NAME_EDUCATION_TYPE_numeric: double, NAME_FAMILY_STATUS_numeric: double, NAME_HOUSING_TYPE_numeric: double, TARGET: bigint]

[ ]:
```
once_count = balanced_df.where(balanced_df["TARGET"] == 1).count()
zeros_count = balanced_df.where(balanced_df["TARGET"] == 0).count()

labels = 'Not Approve (1)', 'Approve (0)'
sizes = [once_count, zeros_count]
colors = ['skyblue', 'cyan']
explode_ratio = (0.1, 0)

# Plot
plt.figure(figsize=(8, 6))
```

```
plt.pie(sizes, explode=explode_ratio, labels=labels, colors=colors, autopct='%1.
    ↪1f%%', shadow=True, startangle=140)
plt.axis('equal')
plt.title('Distribution of Approve and Not Approve Credit Check')
plt.show()

# print(sizes)
```



[ ]:
```
from pyspark.sql.functions import col, log1p
from pyspark.sql import functions as F

# Calculate skewness for each feature
skewness_df = balanced_df.select(
    [F.skewness(col(feature)).alias(feature) for feature in balanced_df.columns␣
    ↪if feature != "TARGET"]
).toPandas()

# Define a threshold for skewness
skew_threshold = 1.0
```

```
# skewed features
skewed_features = skewness_df.columns[(skewness_df.abs() > skew_threshold).
 ↪values[0]]
print(skewed_features)

# Apply log1p transformation to skewed features
for feature in skewed_features:
    balanced_df = balanced_df.withColumn(f"log_{feature}", log1p(col(feature)))

updated_features = [f"log_{feature}" if feature in skewed_features else feature␣
 ↪for feature in balanced_df.columns if feature != "TARGET"]
print(updated_features)
```

```
Index(['CNT_CHILDREN', 'AMT_INCOME_TOTAL', 'AMT_CREDIT', 'AMT_ANNUITY',
       'AMT_GOODS_PRICE', 'DAYS_EMPLOYED_POSITIVE',
       'NAME_CONTRACT_TYPE_numeric', 'NAME_INCOME_TYPE_numeric',
       'NAME_EDUCATION_TYPE_numeric', 'NAME_FAMILY_STATUS_numeric',
       'NAME_HOUSING_TYPE_numeric'],
      dtype='object')
['SK_ID_CURR', 'log_CNT_CHILDREN', 'log_AMT_INCOME_TOTAL', 'log_AMT_CREDIT',
'log_AMT_ANNUITY', 'log_AMT_GOODS_PRICE', 'EXT_SOURCE_2', 'EXT_SOURCE_3',
'REGION_RATING_CLIENT', 'REGION_RATING_CLIENT_W_CITY', 'AGE_YEARS',
'log_DAYS_EMPLOYED_POSITIVE', 'DAYS_REGISTRATION_POSITIVE',
'DAYS_ID_PUBLISH_POSITIVE', 'log_NAME_CONTRACT_TYPE_numeric',
'CODE_GENDER_numeric', 'FLAG_OWN_CAR_numeric', 'FLAG_OWN_REALTY_numeric',
'log_NAME_INCOME_TYPE_numeric', 'log_NAME_EDUCATION_TYPE_numeric',
'log_NAME_FAMILY_STATUS_numeric', 'log_NAME_HOUSING_TYPE_numeric',
'log_CNT_CHILDREN', 'log_AMT_INCOME_TOTAL', 'log_AMT_CREDIT', 'log_AMT_ANNUITY',
'log_AMT_GOODS_PRICE', 'log_DAYS_EMPLOYED_POSITIVE',
'log_NAME_CONTRACT_TYPE_numeric', 'log_NAME_INCOME_TYPE_numeric',
'log_NAME_EDUCATION_TYPE_numeric', 'log_NAME_FAMILY_STATUS_numeric',
'log_NAME_HOUSING_TYPE_numeric']
```

6 Combine all the features in one single feature vector

There are multiple approaches to creating a single feature vector, but we opt for the VectorAssembler method. This feature transformer accepts a range of input column types, including all numeric types, boolean, and vector types. In PySpark, VectorAssembler plays a crucial role in data preparation, especially when handling large-scale datasets.

[ ]:
```
from pyspark.ml.feature import VectorAssembler

assembler = VectorAssembler(inputCols=updated_features,outputCol="features")
```

7 Dataset Splitting into Traning and Testing Sets

[ ]:
```
from pyspark.ml.feature import StandardScaler

train, test = balanced_df.randomSplit([0.8, 0.2], seed=47)

# Transforms the features
standardscaler = StandardScaler(inputCol="features",␣
 ↪outputCol="Scaled_features")
```

In this work, we have systematically selected and implemented two distinct prediction models, each tailored to address the specific requirements of loan credit risk assessment. By employing these models in tandem, we aim to develop a robust strategy that enhances the accuracy and efficiency of credit risk evaluation, ultimately contributing to a more secure and reliable financial environment.

The Random Forest Classifier is a pivotal component in our multi-model approach, renowned for its robustness and precision in evaluating loan creditworthiness. By harnessing the strengths of ensemble learning, this model significantly enhances the reliability and accuracy of our credit risk assessment system.

[ ]:
```
from pyspark.ml.classification import RandomForestClassifier
from pyspark.ml import Pipeline

# Define rf classifier
rf = RandomForestClassifier(labelCol="TARGET", featuresCol="Scaled_features")
rf_pipeline = Pipeline(stages=[assembler, standardscaler, rf])
```

Our Gradient-Boosted Trees (GBT) model is a game-changer in our multi-model approach. It's incredibly skilled at uncovering hidden patterns and complex relationships in the data, which helps us make more accurate predictions about loan credit risk. The way it works is by combining the strengths of multiple models, refining its predictions with each iteration, and continuously learning from the data. This adaptability is what makes GBT so valuable - it can handle a wide range of features and interactions, ensuring that our creditworthiness evaluations are precise and reliable. In short, GBT is a powerhouse that helps us make better decisions and provide more accurate assessments.

[ ]:
```
from pyspark.ml.classification import GBTClassifier

# Define GBT classifier
gbt = GBTClassifier(labelCol="TARGET", featuresCol="Scaled_features")
gbt_pipeline = Pipeline(stages=[assembler, standardscaler, gbt])
```

Model parameter tuning is a crucial step in optimizing the performance of a machine learning model. This process involves adjusting the model's factors to achieve the best possible results. Hyperparameter tuning is a key aspect of this process, where different values are systematically tested for each parameter, and the combination that yields the highest performance on a validation set is selected. This iterative process ensures that the model is fine-tuned to make accurate predictions and improve its overall reliability.

To optimize the performance of our loan credit risk assessment models, we employed the robust technique of cross-validation (CV) to tune the model parameters. This method is instrumental in evaluating the generalizability and performance of machine learning models. By dividing the dataset into smaller subsets, known as folds, and iteratively training and testing the models on different combinations of training and testing datasets, CV provides a comprehensive assessment of the model's ability to generalize to unseen data. During the training process, a portion of the dataset is reserved as a test set to evaluate the model's performance on unseen data, which is the primary objective of CV. By applying CV to all three models, we ensured that the optimal parameters were selected to achieve the best possible performance.

[ ]:
```
from pyspark.ml.evaluation import BinaryClassificationEvaluator
from pyspark.ml.tuning import CrossValidator, ParamGridBuilder


# Parameter grid
random_forest_params = ParamGridBuilder() \
    .addGrid(rf.numTrees, [10, 50, 100]) \
    .addGrid(rf.maxDepth, [5, 10, 20]) \
    .build()

# Cross-validation
rf_evaluator = BinaryClassificationEvaluator(labelCol="TARGET")
rf_crossval = CrossValidator(estimator=rf_pipeline,␣
↪estimatorParamMaps=random_forest_params, evaluator=rf_evaluator,numFolds=2)

# Find the best model
rf_cv_model = rf_crossval.fit(train)
rf_best_model = rf_cv_model.bestModel

# Make predictions
rf_predictions = rf_best_model.transform(test)
```

[ ]:
```
from pyspark.ml.evaluation import BinaryClassificationEvaluator
from pyspark.ml.tuning import ParamGridBuilder, CrossValidator

# Parameter grid
gbt_params = gbt_params = ParamGridBuilder() \
    .addGrid(gbt.maxIter, [10]) \
    .addGrid(gbt.maxDepth, [5]) \
    .addGrid(gbt.stepSize, [0.1]) \
    .build()


# Cross-validation
gbt_evaluator = BinaryClassificationEvaluator(labelCol="TARGET")
gbt_crossval = CrossValidator(estimator=gbt_pipeline,␣
↪estimatorParamMaps=gbt_params, evaluator=gbt_evaluator, numFolds=2)
```

```
# Find the best model
gbt_cv_model = gbt_crossval.fit(train)
gbt_best_model = gbt_cv_model.bestModel

# Make predictions
gbt_predictions = gbt_best_model.transform(test)
```

When building a classification model, we need to measure its performance. Here are four key metrics to help us do that:

Accuracy: How often is the model correct?

Precision: How many positive predictions are actually true?

Recall: How many actual positives are correctly identified?

F1-Score: A balanced measure of precision and recall.

[ ]:
```
# Performance of the model
from pyspark.ml.evaluation import MulticlassClassificationEvaluator

# Define the evaluator
evaluator = MulticlassClassificationEvaluator(labelCol="TARGET")

# Set the evaluation metric names
evaluator.setMetricName("accuracy")
acc = evaluator.evaluate(rf_predictions)

evaluator.setMetricName("precisionByLabel")
precision = evaluator.evaluate(rf_predictions)

evaluator.setMetricName("recallByLabel")
recall = evaluator.evaluate(rf_predictions)

f1 = 2 * (precision * recall) / (precision + recall)

print(f"Accuracy: {acc}")
print(f"Precision: {precision}")
print(f"Recall: {recall}")
print(f"F1-Score: {f1}")
```

```
Accuracy: 0.7555358724534986
Precision: 0.7320875600640684
Recall: 0.8078237304112171
F1-Score: 0.7680932108447232
```

[ ]:
```
from pyspark.ml.evaluation import MulticlassClassificationEvaluator
```

```python
evaluator.setMetricName("accuracy")
acc = evaluator.evaluate(gbt_predictions)

evaluator.setMetricName("precisionByLabel")
precision = evaluator.evaluate(gbt_predictions)

evaluator.setMetricName("recallByLabel")
recall = evaluator.evaluate(gbt_predictions)

f1 = 2 * (precision * recall) / (precision + recall)

print(f"Accuracy: {acc}")
print(f"Precision: {precision}")
print(f"Recall: {recall}")
print(f"F1-Score: {f1}")
```

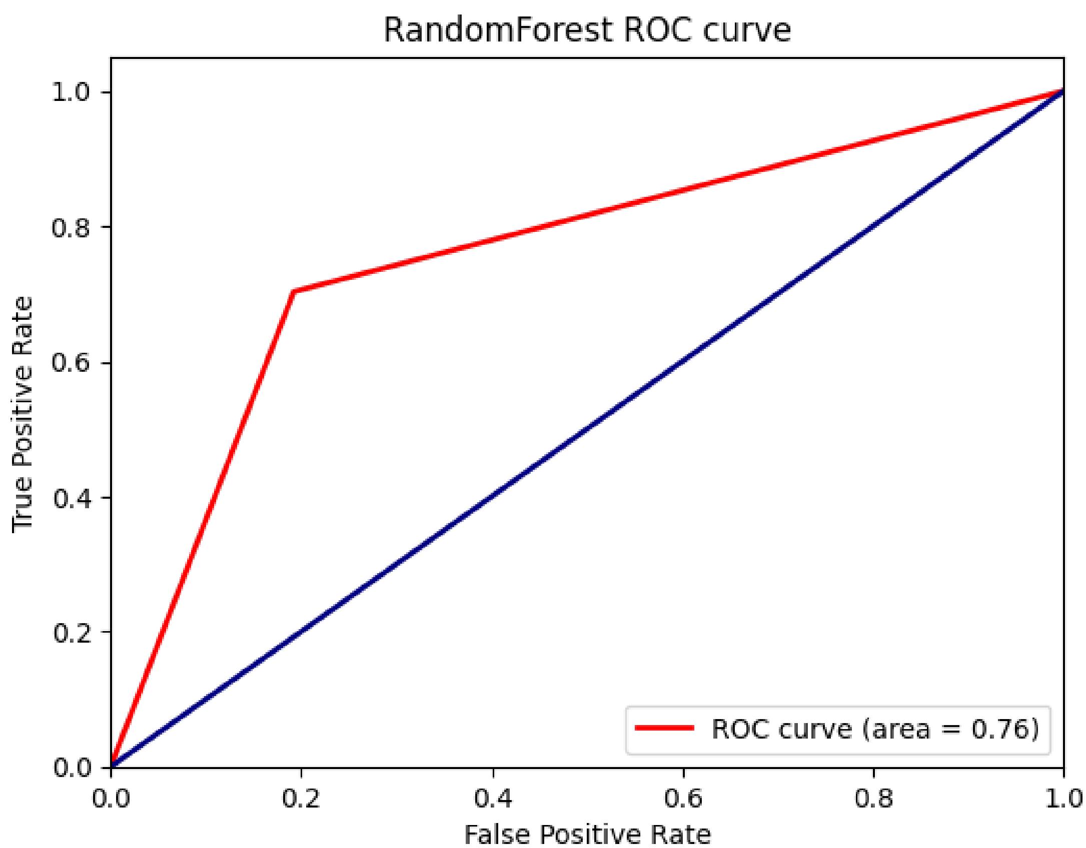
```
Accuracy: 0.7271922054915855
Precision: 0.7119833351606184
Recall: 0.7651702603982562
F1-Score: 0.7376192639709223
```

A powerful visualization is the key to unlocking the story behind the data, and it plays a vital role in the ongoing quest to detect, investigate, and prevent loan defaults, while also uncovering new opportunities to optimize lending strategies and improve customer relationships.

This Receiver Operating Characteristic (ROC) curve plots the True Positive Rate against the False Positive Rate at different thresholds, showcasing the performance of our RandomForest model in predicting loan creditworthiness.

[ ]:
```python
import pandas as pd
import seaborn as sns
from sklearn.metrics import confusion_matrix
import matplotlib.pyplot as plt
from sklearn.metrics import roc_curve, auc
from sklearn.metrics import precision_recall_curve

# pandas df
y_true = rf_predictions.select("TARGET").toPandas()
y_prediction = rf_predictions.select("prediction").toPandas()



# ROC Curve
fpr, tpr, _ = roc_curve(y_true, y_prediction)
roc_auc = auc(fpr, tpr)

# Plot ROC curve
plt.figure()
```

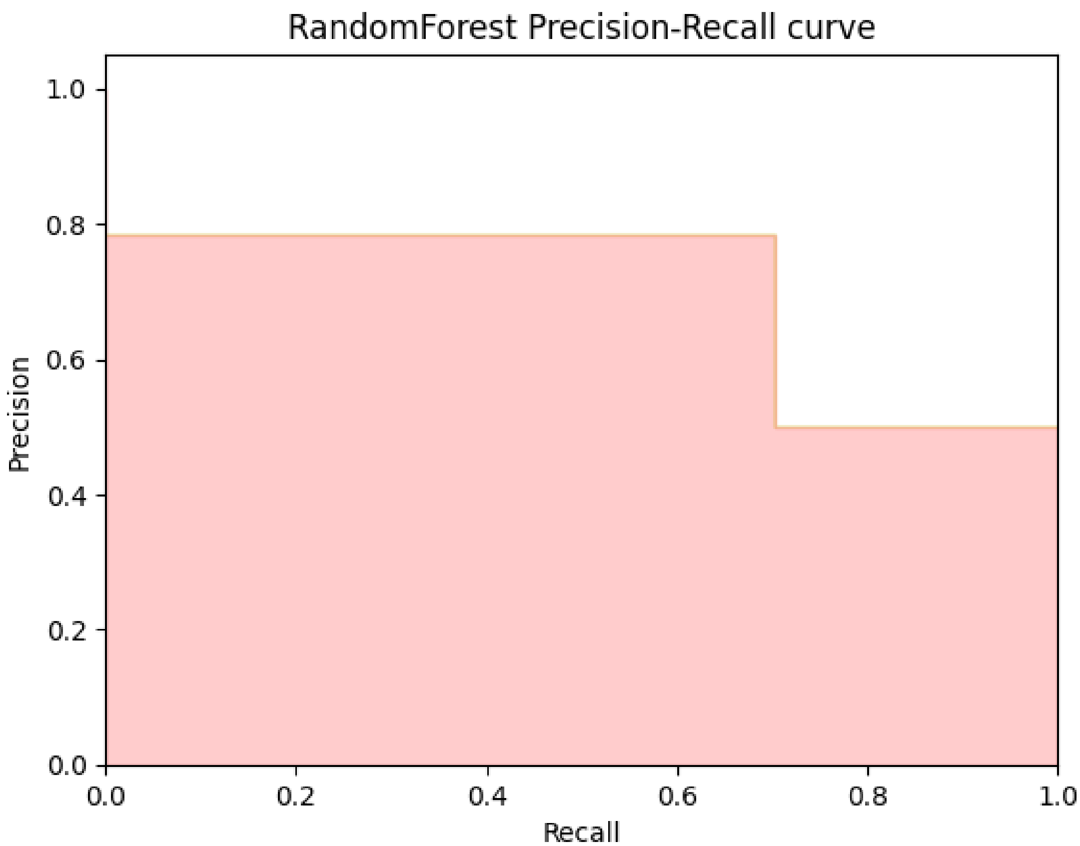
```
plt.plot(fpr, tpr, color='red', lw=2, label='ROC curve (area = %0.2f)' %␣
↪roc_auc)
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='-')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('RandomForest ROC curve')
plt.legend(loc="lower right")
plt.show()
```



This Precision-Recall curve plots the trade-off between Precision (the proportion of true positives among all predicted positives) and Recall (the proportion of true positives among all actual positives) at different thresholds.

[ ]:
```
precision, recall, _ = precision_recall_curve(y_true, y_prediction)

# Plot precision-recall curve
plt.figure()
plt.step(recall, precision, color='y', alpha=0.2, where='post')
```

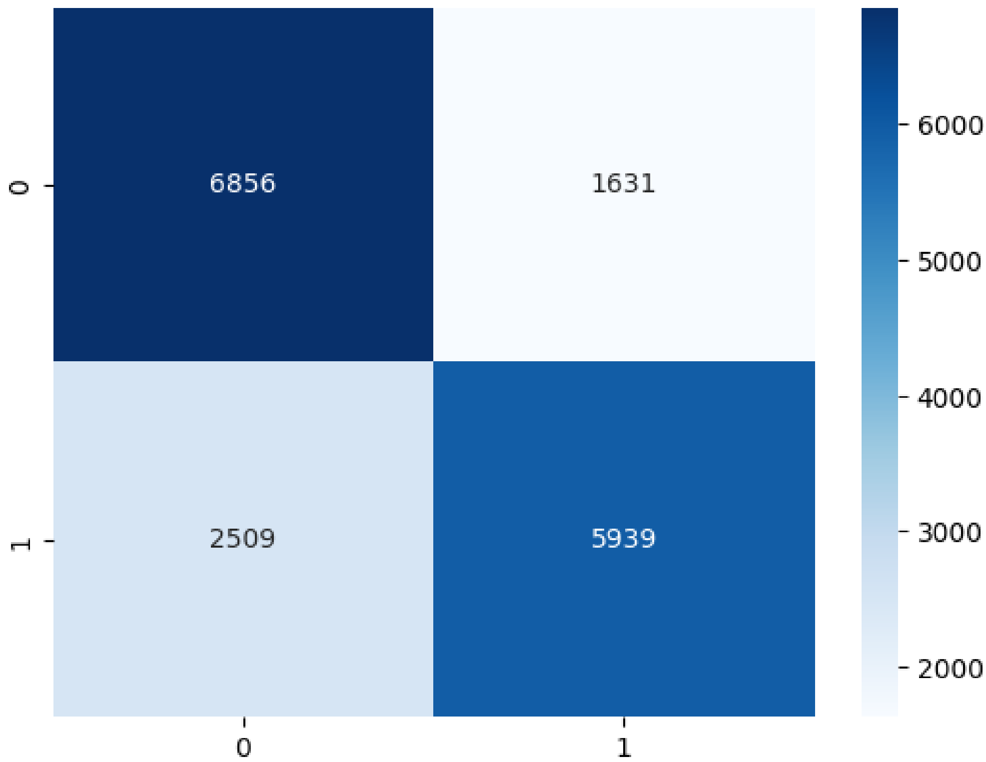
```
plt.fill_between(recall, precision, step='post', alpha=0.2, color='r')
plt.xlabel('Recall')
plt.ylabel('Precision')
plt.ylim([0.0, 1.05])
plt.xlim([0.0, 1.0])
plt.title('RandomForest Precision-Recall curve')
plt.show()
```



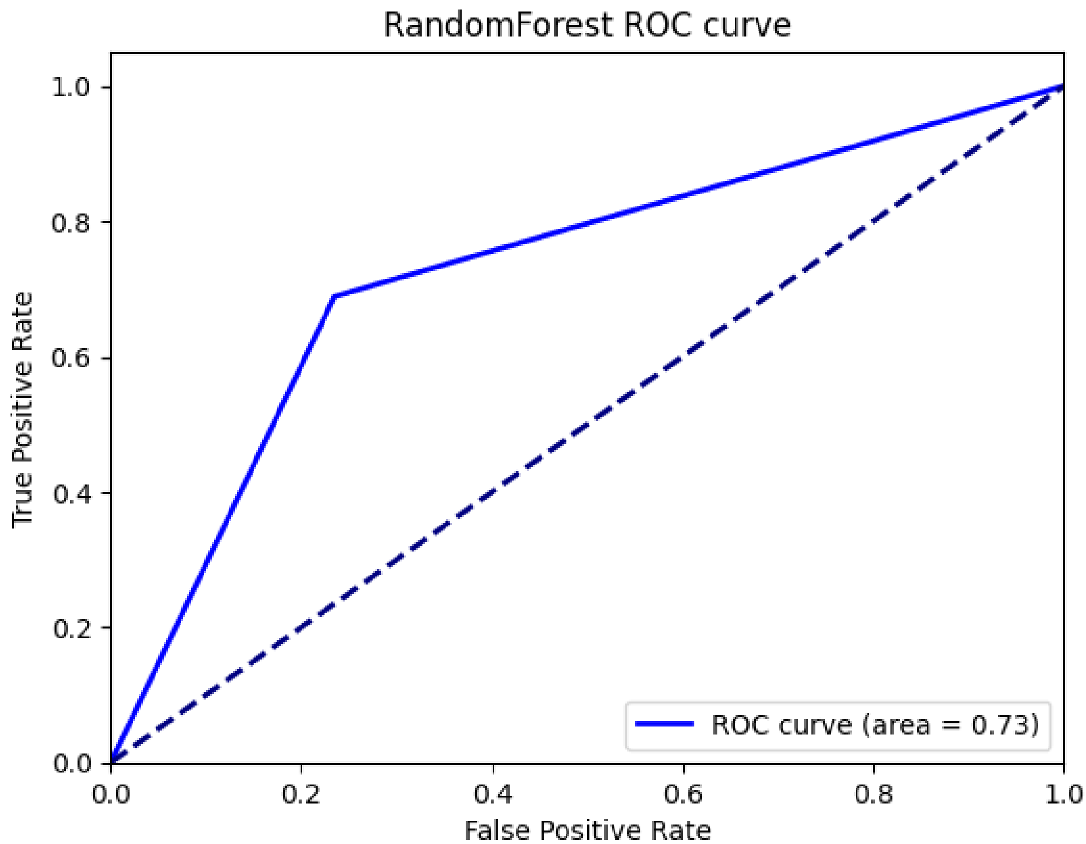
This Confusion Matrix displays the number of true positives (correctly predicted good loans), true negatives (correctly predicted bad loans), false positives (bad loans misclassified as good), and false negatives (good loans misclassified as bad) made by our RandomForest model. The diagonal elements represent the number of correct predictions, while off-diagonal elements represent the number of incorrect predictions. This matrix provides a summary of the model's performance and helps identify areas for improvement.

[ ]:
```
# Confusion matrix
confusion_matrix = confusion_matrix(y_true, y_prediction)

sns.heatmap(confusion_matrix, annot=True, fmt="d", cmap="Blues")
```

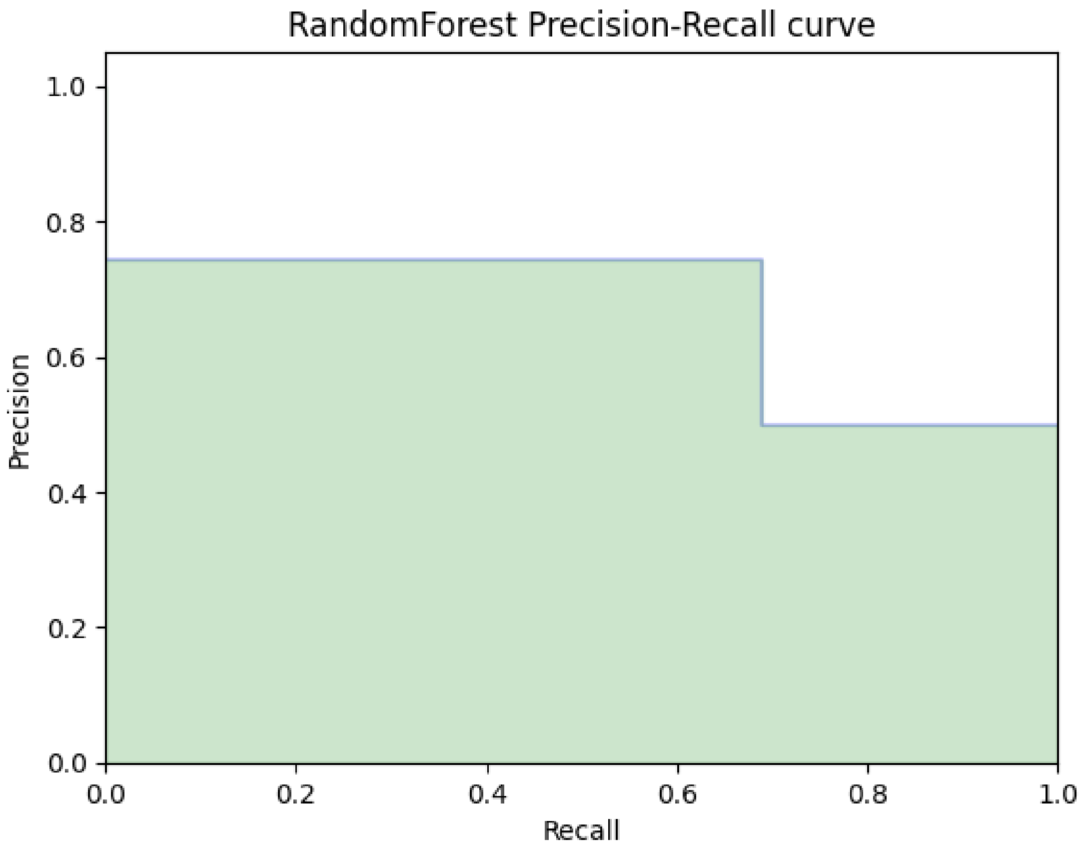
[ ]: <Axes: >



[ ]:
```python
y_true = gbt_predictions.select("TARGET").toPandas()
y_prediction = gbt_predictions.select("prediction").toPandas()


# ROC Curve
fpr, tpr, _ = roc_curve(y_true, y_prediction)
roc_auc = auc(fpr, tpr)

# Plot ROC curve
plt.figure()
plt.plot(fpr, tpr, color='blue', lw=2, label='ROC curve (area = %0.2f)' %␣
 ↪roc_auc)
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('RandomForest ROC curve')
plt.legend(loc="lower right")
plt.show()
```

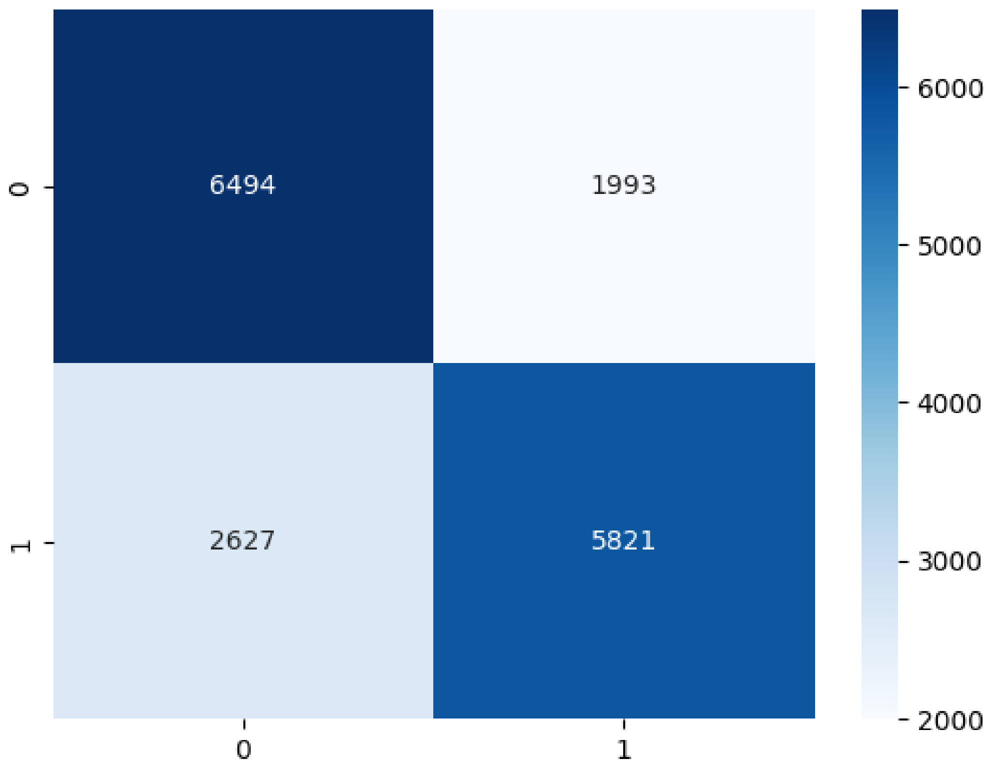


[ ]:
```
precision, recall, _ = precision_recall_curve(y_true, y_prediction)

# Plot precision-recall curve
plt.figure()
plt.step(recall, precision, color='b', alpha=0.2, where='post')
plt.fill_between(recall, precision, step='post', alpha=0.2, color='g')
plt.xlabel('Recall')
plt.ylabel('Precision')
plt.ylim([0.0, 1.05])
plt.xlim([0.0, 1.0])
plt.title('RandomForest Precision-Recall curve')
plt.show()
```



[ ]:
```
from sklearn.metrics import confusion_matrix as cm

# Confusion matrix
conf_mat = cm(y_true, y_prediction)

sns.heatmap(conf_mat, annot=True, fmt="d", cmap="Blues")
```

[ ]: <Axes: >



#Legal

Our Loan Credit project prioritizes compliance with regulatory requirements, ensuring that our use of financial data is both responsible and secure. We've implemented robust data governance practices to safeguard sensitive information and maintain the trust of our stakeholders. By adhering to relevant laws and regulations, such as the FCRA and ECOA, we demonstrate our commitment to upholding the highest standards of data protection and privacy.

#Social

The Loan Credit project has the potential to make a meaningful impact on people's lives by providing more accurate and inclusive credit scoring. By leveraging advanced data analysis and machine learning techniques, we aim to create a more level playing field for individuals and businesses seeking access to credit. Our project promotes financial inclusion, economic opportunities, and social mobility, ultimately contributing to a more equitable society.

#Ethical

At the heart of our Loan Credit project is a deep commitment to ethics and fairness. We recognize the risks of bias in credit scoring models and have taken proactive steps to mitigate them. Our approach emphasizes transparency, accountability, and explainability, ensuring that our models are fair, reliable, and free from discriminatory practices. By prioritizing ethics, we build trust with our stakeholders and foster a more responsible and sustainable financial ecosystem.

#Professional

Our Loan Credit project showcases our expertise in data analysis, machine learning, and software development. We've applied cutting-edge techniques and tools to create a robust and scalable solution that meets the highest standards of quality and performance. Through our project, we demonstrate our ability to work collaboratively, think critically, and communicate complex ideas effectively, highlighting our professionalism and dedication to delivering exceptional results.

8 Report HTML template

[ ]:
```
# install nbconvert (if facing the conversion error)
!pip3 install nbconvert
```

```
Requirement already satisfied: nbconvert in /usr/local/lib/python3.10/dist-
packages (6.5.4)
Requirement already satisfied: lxml in /usr/local/lib/python3.10/dist-packages
(from nbconvert) (4.9.4)
Requirement already satisfied: beautifulsoup4 in /usr/local/lib/python3.10/dist-
packages (from nbconvert) (4.12.3)
Requirement already satisfied: bleach in /usr/local/lib/python3.10/dist-packages
(from nbconvert) (6.1.0)
Requirement already satisfied: defusedxml in /usr/local/lib/python3.10/dist-
packages (from nbconvert) (0.7.1)
Requirement already satisfied: entrypoints>=0.2.2 in
/usr/local/lib/python3.10/dist-packages (from nbconvert) (0.4)
Requirement already satisfied: jinja2>=3.0 in /usr/local/lib/python3.10/dist-
packages (from nbconvert) (3.1.4)
Requirement already satisfied: jupyter-core>=4.7 in
/usr/local/lib/python3.10/dist-packages (from nbconvert) (5.7.2)
Requirement already satisfied: jupyterlab-pygments in
/usr/local/lib/python3.10/dist-packages (from nbconvert) (0.3.0)
Requirement already satisfied: MarkupSafe>=2.0 in
/usr/local/lib/python3.10/dist-packages (from nbconvert) (2.1.5)
Requirement already satisfied: mistune<2,>=0.8.1 in
/usr/local/lib/python3.10/dist-packages (from nbconvert) (0.8.4)
Requirement already satisfied: nbclient>=0.5.0 in
/usr/local/lib/python3.10/dist-packages (from nbconvert) (0.10.0)
Requirement already satisfied: nbformat>=5.1 in /usr/local/lib/python3.10/dist-
packages (from nbconvert) (5.10.4)
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-
packages (from nbconvert) (24.1)
Requirement already satisfied: pandocfilters>=1.4.1 in
/usr/local/lib/python3.10/dist-packages (from nbconvert) (1.5.1)
Requirement already satisfied: pygments>=2.4.1 in
/usr/local/lib/python3.10/dist-packages (from nbconvert) (2.16.1)
Requirement already satisfied: tinycss2 in /usr/local/lib/python3.10/dist-
packages (from nbconvert) (1.3.0)
Requirement already satisfied: traitlets>=5.0 in /usr/local/lib/python3.10/dist-
packages (from nbconvert) (5.7.1)
```

```
Requirement already satisfied: platformdirs>=2.5 in
/usr/local/lib/python3.10/dist-packages (from jupyter-core>=4.7->nbconvert)
(4.2.2)
Requirement already satisfied: jupyter-client>=6.1.12 in
/usr/local/lib/python3.10/dist-packages (from nbclient>=0.5.0->nbconvert)
(6.1.12)
Requirement already satisfied: fastjsonschema>=2.15 in
/usr/local/lib/python3.10/dist-packages (from nbformat>=5.1->nbconvert) (2.20.0)
Requirement already satisfied: jsonschema>=2.6 in
/usr/local/lib/python3.10/dist-packages (from nbformat>=5.1->nbconvert) (4.23.0)
Requirement already satisfied: soupsieve>1.2 in /usr/local/lib/python3.10/dist-
packages (from beautifulsoup4->nbconvert) (2.6)
Requirement already satisfied: six>=1.9.0 in /usr/local/lib/python3.10/dist-
packages (from bleach->nbconvert) (1.16.0)
Requirement already satisfied: webencodings in /usr/local/lib/python3.10/dist-
packages (from bleach->nbconvert) (0.5.1)
Requirement already satisfied: attrs>=22.2.0 in /usr/local/lib/python3.10/dist-
packages (from jsonschema>=2.6->nbformat>=5.1->nbconvert) (24.2.0)
Requirement already satisfied: jsonschema-specifications>=2023.03.6 in
/usr/local/lib/python3.10/dist-packages (from
jsonschema>=2.6->nbformat>=5.1->nbconvert) (2023.12.1)
Requirement already satisfied: referencing>=0.28.4 in
/usr/local/lib/python3.10/dist-packages (from
jsonschema>=2.6->nbformat>=5.1->nbconvert) (0.35.1)
Requirement already satisfied: rpds-py>=0.7.1 in /usr/local/lib/python3.10/dist-
packages (from jsonschema>=2.6->nbformat>=5.1->nbconvert) (0.20.0)
Requirement already satisfied: pyzmq>=13 in /usr/local/lib/python3.10/dist-
packages (from jupyter-client>=6.1.12->nbclient>=0.5.0->nbconvert) (24.0.1)
Requirement already satisfied: python-dateutil>=2.1 in
/usr/local/lib/python3.10/dist-packages (from jupyter-
client>=6.1.12->nbclient>=0.5.0->nbconvert) (2.8.2)
Requirement already satisfied: tornado>=4.1 in /usr/local/lib/python3.10/dist-
packages (from jupyter-client>=6.1.12->nbclient>=0.5.0->nbconvert) (6.3.3)
```

[ ]:
```
# convert ipynb to html
!jupyter nbconvert --to html Your_Group_ID_CRWK_CN7030.ipynb

from nbconvert import HTMLExporter
import nbformat

notebook_filename = '/content/drive/MyDrive/Loan_credit/loan_credit.ipynb'

# Read the notebook
with open(notebook_filename, 'r', encoding='utf-8') as notebook_file:
    notebook_content = notebook_file.read()

# Convert notebook to HTML
```

```python
notebook = nbformat.reads(notebook_content, as_version=4)
html_exporter = HTMLExporter()
(body, resources) = html_exporter.from_notebook_node(notebook)

# Save the HTML content to a file
html_filename = notebook_filename.replace('.ipynb', '.html')
with open(html_filename, 'w', encoding='utf-8') as html_file:
    html_file.write(body)
```

```
[NbConvertApp] WARNING | pattern 'Your_Group_ID_CRWK_CN7030.ipynb' matched no
files
This application is used to convert notebook files (*.ipynb)
        to various other formats.

        WARNING: THE COMMANDLINE INTERFACE MAY CHANGE IN FUTURE RELEASES.

Options
=======
The options below are convenience aliases to configurable class-options,
as listed in the "Equivalent to" description-line of the aliases.
To see all configurable class-options for some <cmd>, use:
    <cmd> --help-all

--debug
    set log level to logging.DEBUG (maximize logging output)
    Equivalent to: [--Application.log_level=10]
--show-config
    Show the application's configuration (human-readable format)
    Equivalent to: [--Application.show_config=True]
--show-config-json
    Show the application's configuration (json format)
    Equivalent to: [--Application.show_config_json=True]
--generate-config
    generate default config file
    Equivalent to: [--JupyterApp.generate_config=True]
-y
    Answer yes to any questions instead of prompting.
    Equivalent to: [--JupyterApp.answer_yes=True]
--execute
    Execute the notebook prior to export.
    Equivalent to: [--ExecutePreprocessor.enabled=True]
--allow-errors
    Continue notebook execution even if one of the cells throws an error and
include the error message in the cell output (the default behaviour is to abort
conversion). This flag is only relevant if '--execute' was specified, too.
    Equivalent to: [--ExecutePreprocessor.allow_errors=True]
--stdin
```

```
    read a single notebook file from stdin. Write the resulting notebook with
default basename 'notebook.*'
    Equivalent to: [--NbConvertApp.from_stdin=True]
--stdout
    Write notebook output to stdout instead of files.
    Equivalent to: [--NbConvertApp.writer_class=StdoutWriter]
--inplace
    Run nbconvert in place, overwriting the existing notebook (only
            relevant when converting to notebook format)
    Equivalent to: [--NbConvertApp.use_output_suffix=False
--NbConvertApp.export_format=notebook --FilesWriter.build_directory=]
--clear-output
    Clear output of current file and save in place,
            overwriting the existing notebook.
    Equivalent to: [--NbConvertApp.use_output_suffix=False
--NbConvertApp.export_format=notebook --FilesWriter.build_directory=
--ClearOutputPreprocessor.enabled=True]
--no-prompt
    Exclude input and output prompts from converted document.
    Equivalent to: [--TemplateExporter.exclude_input_prompt=True
--TemplateExporter.exclude_output_prompt=True]
--no-input
    Exclude input cells and output prompts from converted document.
            This mode is ideal for generating code-free reports.
    Equivalent to: [--TemplateExporter.exclude_output_prompt=True
--TemplateExporter.exclude_input=True
--TemplateExporter.exclude_input_prompt=True]
--allow-chromium-download
    Whether to allow downloading chromium if no suitable version is found on the
system.
    Equivalent to: [--WebPDFExporter.allow_chromium_download=True]
--disable-chromium-sandbox
    Disable chromium security sandbox when converting to PDF..
    Equivalent to: [--WebPDFExporter.disable_sandbox=True]
--show-input
    Shows code input. This flag is only useful for dejavu users.
    Equivalent to: [--TemplateExporter.exclude_input=False]
--embed-images
    Embed the images as base64 dataurls in the output. This flag is only useful
for the HTML/WebPDF/Slides exports.
    Equivalent to: [--HTMLExporter.embed_images=True]
--sanitize-html
    Whether the HTML in Markdown cells and cell outputs should be sanitized..
    Equivalent to: [--HTMLExporter.sanitize_html=True]
--log-level=<Enum>
    Set the log level by value or name.
    Choices: any of [0, 10, 20, 30, 40, 50, 'DEBUG', 'INFO', 'WARN', 'ERROR',
'CRITICAL']
```

```
    Default: 30
    Equivalent to: [--Application.log_level]
--config=<Unicode>
    Full path of a config file.
    Default: ''
    Equivalent to: [--JupyterApp.config_file]
--to=<Unicode>
    The export format to be used, either one of the built-in formats
            ['asciidoc', 'custom', 'html', 'latex', 'markdown', 'notebook',
'pdf', 'python', 'rst', 'script', 'slides', 'webpdf']
            or a dotted object name that represents the import path for an
            ``Exporter`` class
    Default: ''
    Equivalent to: [--NbConvertApp.export_format]
--template=<Unicode>
    Name of the template to use
    Default: ''
    Equivalent to: [--TemplateExporter.template_name]
--template-file=<Unicode>
    Name of the template file to use
    Default: None
    Equivalent to: [--TemplateExporter.template_file]
--theme=<Unicode>
    Template specific theme(e.g. the name of a JupyterLab CSS theme distributed
    as prebuilt extension for the lab template)
    Default: 'light'
    Equivalent to: [--HTMLExporter.theme]
--sanitize_html=<Bool>
    Whether the HTML in Markdown cells and cell outputs should be sanitized.This
    should be set to True by nbviewer or similar tools.
    Default: False
    Equivalent to: [--HTMLExporter.sanitize_html]
--writer=<DottedObjectName>
    Writer class used to write the
                                    results of the conversion
    Default: 'FilesWriter'
    Equivalent to: [--NbConvertApp.writer_class]
--post=<DottedOrNone>
    PostProcessor class used to write the
                                    results of the conversion
    Default: ''
    Equivalent to: [--NbConvertApp.postprocessor_class]
--output=<Unicode>
    overwrite base name use for output files.
                can only be used when converting one notebook at a time.
    Default: ''
    Equivalent to: [--NbConvertApp.output_base]
--output-dir=<Unicode>
```

```
    Directory to write output(s) to. Defaults
                                  to output to the directory of each notebook.
To recover
                                  previous default behaviour (outputting to the
current
                                  working directory) use . as the flag value.
    Default: ''
    Equivalent to: [--FilesWriter.build_directory]
--reveal-prefix=<Unicode>
    The URL prefix for reveal.js (version 3.x).
            This defaults to the reveal CDN, but can be any url pointing to a
copy
            of reveal.js.
            For speaker notes to work, this must be a relative path to a local
            copy of reveal.js: e.g., "reveal.js".
            If a relative path is given, it must be a subdirectory of the
            current directory (from which the server is run).
            See the usage documentation
            (https://nbconvert.readthedocs.io/en/latest/usage.html#reveal-js-
html-slideshow)
            for more details.
    Default: ''
    Equivalent to: [--SlidesExporter.reveal_url_prefix]
--nbformat=<Enum>
    The nbformat version to write.
            Use this to downgrade notebooks.
    Choices: any of [1, 2, 3, 4]
    Default: 4
    Equivalent to: [--NotebookExporter.nbformat_version]

Examples
--------

    The simplest way to use nbconvert is

            > jupyter nbconvert mynotebook.ipynb --to html

            Options include ['asciidoc', 'custom', 'html', 'latex', 'markdown',
'notebook', 'pdf', 'python', 'rst', 'script', 'slides', 'webpdf'].

            > jupyter nbconvert --to latex mynotebook.ipynb

            Both HTML and LaTeX support multiple output templates. LaTeX
includes
            'base', 'article' and 'report'.  HTML includes 'basic', 'lab' and
            'classic'. You can specify the flavor of the format used.

            > jupyter nbconvert --to html --template lab mynotebook.ipynb
```

```
            You can also pipe the output to stdout, rather than a file

            > jupyter nbconvert mynotebook.ipynb --stdout

            PDF is generated via latex

            > jupyter nbconvert mynotebook.ipynb --to pdf

            You can get (and serve) a Reveal.js-powered slideshow

            > jupyter nbconvert myslides.ipynb --to slides --post serve

            Multiple notebooks can be given at the command line in a couple of
            different ways:

            > jupyter nbconvert notebook*.ipynb
            > jupyter nbconvert notebook1.ipynb notebook2.ipynb

            or you can specify the notebooks list in a config file, containing::

                c.NbConvertApp.notebooks = ["my_notebook.ipynb"]

            > jupyter nbconvert --config mycfg.py

To see all available configurables, use `--help-all`.
```