Prepare Dataset for Model Training and Evaluating

Cell Data set

https://s3.amazonaws.com/cell_data/

Checking Pre-Requisites from the Previous `01_setup/` Folder

In [2]:
```
%store -r setup_instance_check_passed
```

In [3]:
```
try:
    setup_instance_check_passed
except NameError:
    print("++++++++++++++++++++++++++++++++")
    print("[ERROR] YOU HAVE TO RUN ALL NOTEBOOKS IN THE SETUP FOLDER FIRST. You are missing Instance Check.")
    print("++++++++++++++++++++++++++++++++")
```

In [4]:
```
print(setup_instance_check_passed)
```

```
True
```

In [5]:
```
%store -r setup_dependencies_passed
```

In [6]:
```
try:
    setup_dependencies_passed
except NameError:
    print("++++++++++++++++++++++++++++++++")
    print("[ERROR] YOU HAVE TO RUN ALL NOTEBOOKS IN THE SETUP FOLDER FIRST. You are missing Setup Dependencies.")
    print("++++++++++++++++++++++++++++++++")
```

In [7]:
```
print(setup_dependencies_passed)
```

```
True
```

In [8]:
```
%store -r setup_s3_bucket_passed
```

In [9]:
```
try:
    setup_s3_bucket_passed
except NameError:
    print("++++++++++++++++++++++++++++++++")
    print("[ERROR] YOU HAVE TO RUN ALL NOTEBOOKS IN THE SETUP FOLDER FIRST. You are missing Setup S3 Bucket.")
    print("++++++++++++++++++++++++++++++++")
```

In [10]:
```
print(setup_s3_bucket_passed)
```

```
True
```

In [11]:
```
%store -r setup_iam_roles_passed
```

In [12]:
```
try:
    setup_iam_roles_passed
except NameError:
    print("++++++++++++++++++++++++++++++++")
    print("[ERROR] YOU HAVE TO RUN ALL NOTEBOOKS IN THE SETUP FOLDER FIRST. You are missing Setup IAM Roles.")
    print("++++++++++++++++++++++++++++++++")
```

In [13]:
```
print(setup_iam_roles_passed)
```

```
True
```

Check if requirements passed

In [14]:
```
if not setup_instance_check_passed:
    print("++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++")
    print("[ERROR] YOU HAVE TO RUN ALL NOTEBOOKS IN THE SETUP FOLDER FIRST. You are missing Instance Check.")
    print("++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++")
if not setup_dependencies_passed:
    print("++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++")
    print("[ERROR] YOU HAVE TO RUN ALL NOTEBOOKS IN THE SETUP FOLDER FIRST. You are missing Setup Dependencies.")
    print("++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++")
if not setup_s3_bucket_passed:
    print("++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++")
    print("[ERROR] YOU HAVE TO RUN ALL NOTEBOOKS IN THE SETUP FOLDER FIRST. You are missing Setup S3 Bucket.")
    print("++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++")
if not setup_iam_roles_passed:
    print("++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++")
    print("[ERROR] YOU HAVE TO RUN ALL NOTEBOOKS IN THE SETUP FOLDER FIRST. You are missing Setup IAM Roles.")
    print("++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++")
```

In [15]:
```
import boto3
import sagemaker
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import json

sess = sagemaker.Session()
bucket = sess.default_bucket()
role = sagemaker.get_execution_role()
region = boto3.Session().region_name

sm = boto3.Session().client(service_name="sagemaker", region_name=region)
```

Download

Let's start by retrieving a subset of the Amazon Customer Reviews dataset.

In [16]:
```
!aws s3 cp 's3://sagemaker-us-east-1-614093401978/cell_data/OHSU_BeatAMLWaves1_2_Tyner_ClinicalSummary.csv' ./data/
```

```
download: s3://sagemaker-us-east-1-614093401978/cell_data/OHSU_BeatAMLWaves1_2_Tyner_ClinicalSummary.csv to data/OHS
U_BeatAMLWaves1_2_Tyner_ClinicalSummary.csv
```

In [17]:
```
import csv

df = pd.read_csv(
    's3://sagemaker-us-east-1-614093401978/cell_data/OHSU_BeatAMLWaves1_2_Tyner_ClinicalSummary.csv')
df.shape
```

Out[17]:
```
(672, 159)
```

In [18]:
```
df.head(5)
```

Out[18]:

	LabId	PatientId	consensus_sex	inferred_sex	inferred_ethnicity	centerID	CEBPA_Biallelic	ageAtDiagnosis	isRelapse	isDenovo	...
0	09-00705	163	Male	Male	White	1	n	73.0	False	True	...
1	10-00136	174	Male	Male	White	1	n	69.0	False	True	...
2	10-00172	175	Female	Male	White	1	n	59.0	False	True	...
3	10-00507	45	Female	Female	White	1	n	70.0	False	True	...
4	10-00542	174	Male	Male	White	1	n	69.0	True	False	...

5 rows × 159 columns

In [19]:
```
!aws s3 cp 's3://sagemaker-us-east-1-614093401978/cell_data/OHSU_BeatAMLWaves1_2_Tyner_DrugResponse.csv' ./data/
```

```
download: s3://sagemaker-us-east-1-614093401978/cell_data/OHSU_BeatAMLWaves1_2_Tyner_DrugResponse.csv to data/OHSU_BeatAMLWaves1_2_Tyner_DrugResponse.csv
```

In [20]:
```
df1 = pd.read_csv(
    's3://sagemaker-us-east-1-614093401978/cell_data/OHSU_BeatAMLWaves1_2_Tyner_DrugResponse.csv')
df1.shape
```

Out[20]:
```
(47650, 4)
```

In [21]:
```
df1.head(5)
```

Out[21]:

	inhibitor	lab_id	ic50	auc
0	17-AAG (Tanespimycin)	12-00211	10.000000	225.918025
1	17-AAG (Tanespimycin)	12-00219	0.276661	135.264409
2	17-AAG (Tanespimycin)	12-00258	2.722845	164.561227
3	17-AAG (Tanespimycin)	12-00262	0.123136	111.555971
4	17-AAG (Tanespimycin)	12-00268	10.000000	226.805281

In [22]:
```
clsm = df.replace('', np.NAN)
clsm.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 672 entries, 0 to 671
Columns: 159 entries, LabId to ZRSR2
dtypes: bool(9), float64(22), int64(7), object(121)
memory usage: 793.5+ KB
```

In [23]:
```
clsm.info(2)
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 672 entries, 0 to 671
Data columns (total 159 columns):
 #    Column                                          Dtype
---   ------                                          -----
 0    LabId                                           object
 1    PatientId                                       int64
 2    consensus_sex                                   object
 3    inferred_sex                                    object
 4    inferred_ethnicity                              object
 5    centerID                                        int64
 6    CEBPA_Biallelic                                 object
 7    ageAtDiagnosis                                  float64
 8    isRelapse                                       bool
 9    isDenovo                                        bool
 10   isTransformed                                   bool
 11   finalFusion                                     object
 12   specificDxAtAcquisition_MDSMPN                  bool
 13   nonAML_MDSMPN_specificDxAtAcquisition           bool
 14   priorMalignancyNonMyeloid                       object
 15   priorMalignancyType                             object
 16   cumulativeChemo                                 object
 17   priorMalignancyRadiationTx                      object
 18   priorMDS                                        object
 19   priorMDSMoreThanTwoMths                         object
 20   priorMDSMPN                                     object
 21   priorMDSMPNMoreThanTwoMths                      object
 22   priorMPN                                        object
 23   priorMPNMoreThanTwoMths                         object
 24   dxAtInclusion                                   object
 25   specificDxAtInclusion                           object
 26   ELN2017                                         object
 27   ELN2008                                         object
 28   dxAtSpecimenAcquisition                         object
 29   specificDxAtAcquisition                         object
 30   ageAtSpecimenAcquisition                        float64
 31   timeOfSampleCollectionRelativeToInclusion       int64
 32   specimenGroups                                  object
 33   specimenType                                    object
 34   rnaSeq                                          object
 35   exomeSeq                                        object
 36   totalDrug                                       object
 37   rnaSeqAnalysis                                  object
 38   analysisExomeSeq                                object
```

```
39  analysisDrug                                    object
40  cumulativeTreatmentTypeCount                    int64
41  cumulativeTreatmentTypes                        object
42  cumulativeTreatmentRegimenCount                 int64
43  cumulativeTreatmentRegimens                     object
44  cumulativeTreatmentStageCount                   int64
45  cumulativeTreatmentStages                       object
46  responseToInductionTx                           object
47  typeInductionTx                                 object
48  responseDurationToInductionTx                   float64
49  mostRecentTreatmentType                         object
50  currentRegimen                                  object
51  currentStage                                    object
52  mostRecentTreatmentDuration                     float64
53  vitalStatus                                     object
54  overallSurvival                                 float64
55  causeOfDeath                                    object
56  any_different_labs                              bool
57  any_different_labs_also_beataml                 bool
58  different_lab_ids                               object
59  different_id_karyotype_interval                 int64
60  %.Basophils.in.PB                               float64
61  %.Blasts.in.BM                                  object
62  %.Blasts.in.PB                                  object
63  %.Eosinophils.in.PB                             float64
64  %.Immature.Granulocytes.in.PB                   float64
65  %.Lymphocytes.in.PB                             float64
66  %.Monocytes.in.PB                               float64
67  %.Neutrophils.in.PB                             float64
68  %.Nucleated.RBCs.in.PB                          float64
69  ALT                                             object
70  AST                                             float64
71  Albumin                                         float64
72  Creatinine                                      float64
73  FAB/Blast.Morphology                            object
74  Hematocrit                                      float64
75  Hemoglobin                                      float64
76  Karyotype                                       object
77  LDH                                             float64
78  MCV                                             float64
79  Other.Cytogenetics                              object
80  Platelet.Count                                  float64
81  Surface.Antigens.(Immunohistochemical.Stains)   object
82  Total.Protein                                   float64
```

83	WBC.Count	float64
84	any_different_cgs	bool
85	any_different_cgs_also_beataml	bool
86	different_cgs_lab_ids	object
87	FLT3-ITD	object
88	NPM1	object
89	ABL1	object
90	ASXL1	object
91	ASXL2	object
92	ATM	object
93	BCOR	object
94	BCORL1	object
95	BRAF	object
96	BRCA2	object
97	CALR	object
98	CBL	object
99	CCND2	object
100	CCND3	object
101	CD36	object
102	CEBPA	object
103	CHEK2	object
104	CIITA	object
105	CREBBP	object
106	CSF3R	object
107	CTCF	object
108	CUX1	object
109	DNMT3A	object
110	EP300	object
111	ETV6	object
112	EZH2	object
113	FBXW7	object
114	FLT3	object
115	GATA1	object
116	GATA2	object
117	IDH1	object
118	IDH2	object
119	IKZF1	object
120	JAK1	object
121	JAK2	object
122	JAK3	object
123	KDM6A	object
124	KIT	object
125	KMT2A	object
126	KMT2D	object

```
 127  KRAS                                               object
 128  MEN1                                               object
 129  MPL                                                object
 130  MUTYH                                              object
 131  MYD88                                              object
 132  NF1                                                object
 133  NOTCH1                                             object
 134  NRAS                                               object
 135  PAX5                                               object
 136  PDGFRB                                             object
 137  PHF6                                               object
 138  POT1                                               object
 139  PRDM1                                              object
 140  PTPN11                                             object
 141  RAD21                                              object
 142  ROS1                                               object
 143  RUNX1                                              object
 144  SETBP1                                             object
 145  SF3B1                                              object
 146  SMC1A                                              object
 147  SOCS1                                              object
 148  SRSF2                                              object
 149  STAG2                                              object
 150  STAT3                                              object
 151  SUZ12                                              object
 152  TCL1A                                              object
 153  TET2                                               object
 154  TP53                                               object
 155  TYK2                                               object
 156  U2AF1                                              object
 157  WT1                                                object
 158  ZRSR2                                              object
dtypes: bool(9), float64(22), int64(7), object(121)
memory usage: 793.5+ KB
```

In [24]:
```
!pip install klib
```

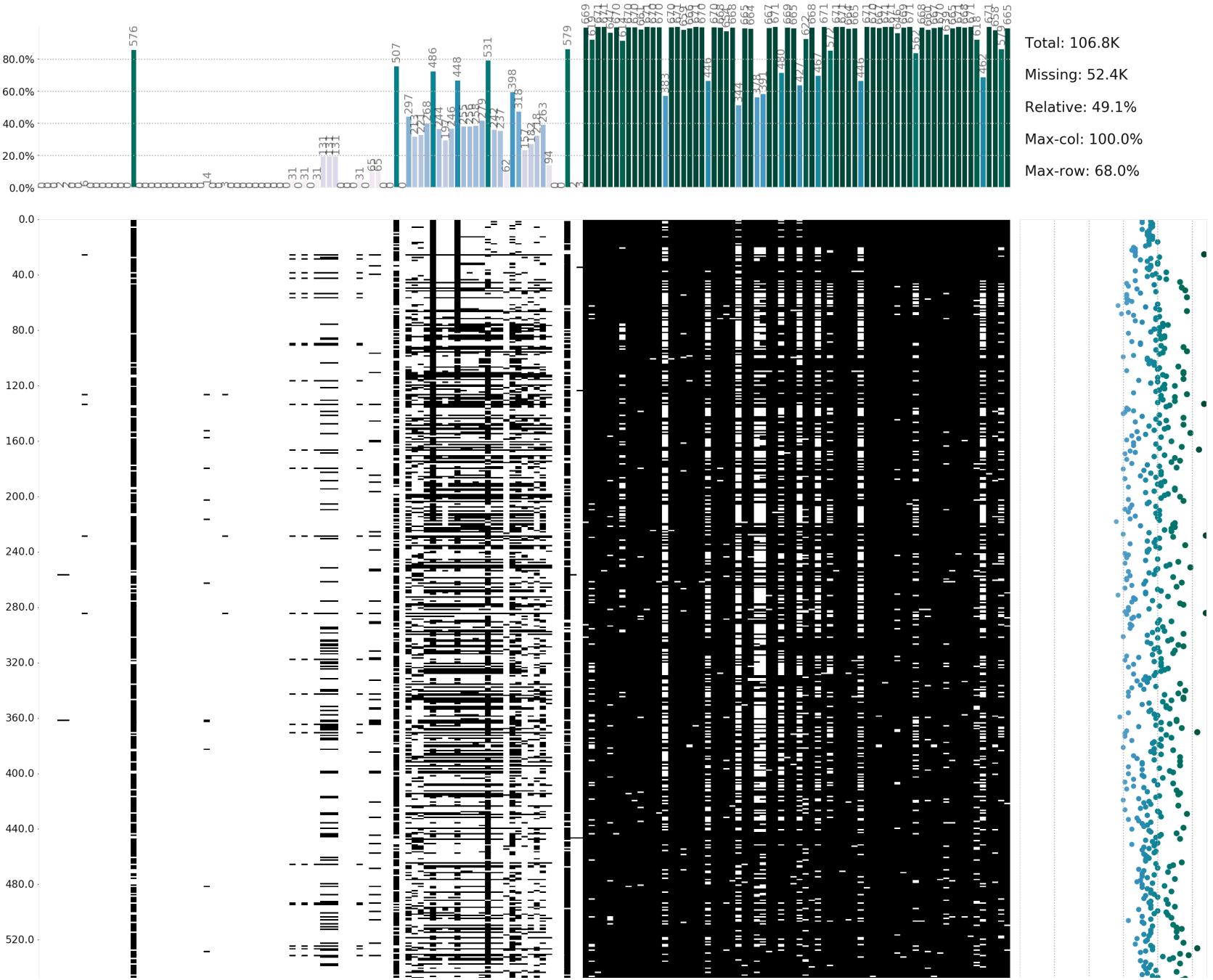
```
Collecting klib
  Using cached klib-1.0.1-py3-none-any.whl (20 kB)
Requirement already satisfied: scipy<2.0.0,>=1.1.0 in /opt/conda/lib/python3.7/site-packages (from klib) (1.4.1)
Requirement already satisfied: numpy<2.0.0,>=1.16.3 in /opt/conda/lib/python3.7/site-packages (from klib) (1.21.6)
Collecting seaborn<0.12.0,>=0.11.1
  Using cached seaborn-0.11.2-py3-none-any.whl (292 kB)
Requirement already satisfied: matplotlib<4.0.0,>=3.0.3 in /opt/conda/lib/python3.7/site-packages (from klib) (3.1.
3)
Requirement already satisfied: pandas<2.0.0,>=1.1.2 in /opt/conda/lib/python3.7/site-packages (from klib) (1.3.5)
Requirement already satisfied: Jinja2<4.0.0,>=3.0.3 in /opt/conda/lib/python3.7/site-packages (from klib) (3.1.2)
Requirement already satisfied: MarkupSafe>=2.0 in /opt/conda/lib/python3.7/site-packages (from Jinja2<4.0.0,>=3.0.3-
>klib) (2.1.2)
Requirement already satisfied: cycler>=0.10 in /opt/conda/lib/python3.7/site-packages (from matplotlib<4.0.0,>=3.0.3
->klib) (0.10.0)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /opt/conda/lib/python3.7/site-packages (f
rom matplotlib<4.0.0,>=3.0.3->klib) (2.4.6)
Requirement already satisfied: kiwisolver>=1.0.1 in /opt/conda/lib/python3.7/site-packages (from matplotlib<4.0.0,>=
3.0.3->klib) (1.1.0)
Requirement already satisfied: python-dateutil>=2.1 in /opt/conda/lib/python3.7/site-packages (from matplotlib<4.0.
0,>=3.0.3->klib) (2.8.2)
Requirement already satisfied: pytz>=2017.3 in /opt/conda/lib/python3.7/site-packages (from pandas<2.0.0,>=1.1.2->kl
ib) (2019.3)
Requirement already satisfied: six in /opt/conda/lib/python3.7/site-packages (from cycler>=0.10->matplotlib<4.0.0,>=
3.0.3->klib) (1.14.0)
Requirement already satisfied: setuptools in /opt/conda/lib/python3.7/site-packages (from kiwisolver>=1.0.1->matplot
lib<4.0.0,>=3.0.3->klib) (59.3.0)
Installing collected packages: seaborn, klib
  Attempting uninstall: seaborn
    Found existing installation: seaborn 0.10.0
    Uninstalling seaborn-0.10.0:
      Successfully uninstalled seaborn-0.10.0
Successfully installed klib-1.0.1 seaborn-0.11.2
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system p
ackage manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv

[notice] A new release of pip is available: 23.0.1 -> 23.1
[notice] To update, run: pip install --upgrade pip
```

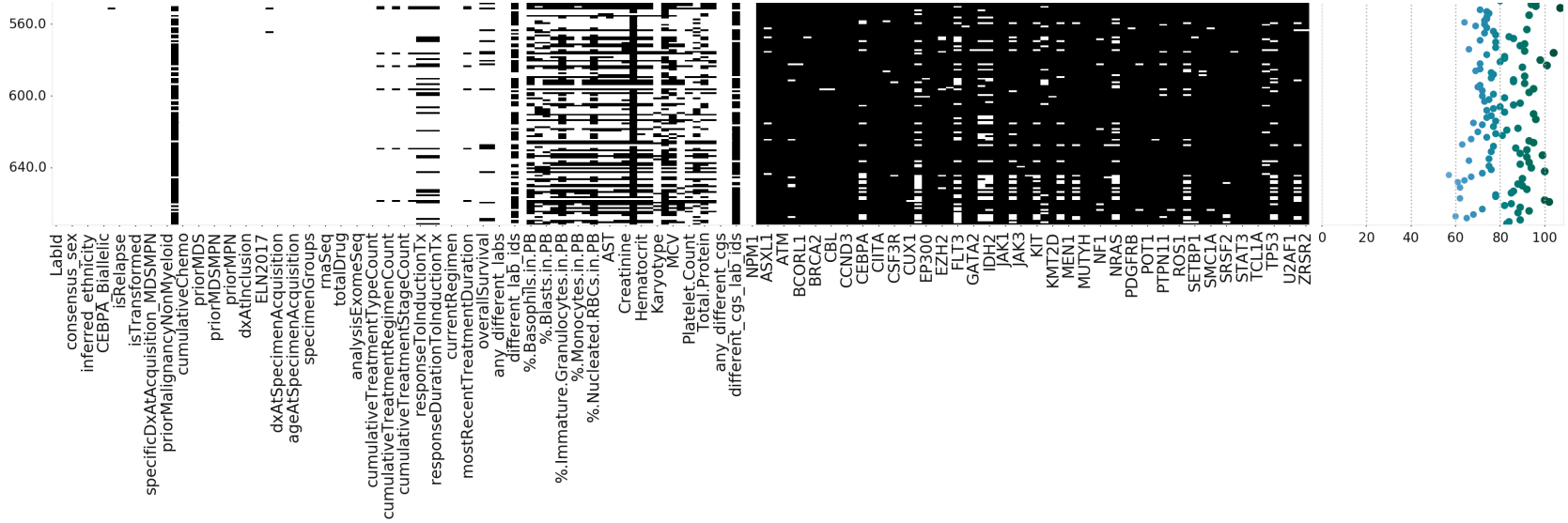
In [25]:
```
import numpy as np
import seaborn as sns
import klib
import matplotlib.pyplot as plt

%matplotlib inline
%config InlineBackend.figure_format='retina'
```

In [26]:
```
klib.missingval_plot(clsm)
```

Out[26]:
```
GridSpec(6, 6)
```

Missing value plot





Select Relevant Features

In [27]:
```
clsm.columns = map(str.lower, clsm.columns)
```

In [28]:
```
clsm_cut = pd.DataFrame(clsm[['labid', 'patientid', 'consensus_sex', 'inferred_ethnicity', 'isrelapse',
                              'istransformed', 'priormalignancynonmyeloid', 'priormds', 'priormdsmpn', 'priormpn',
                              'eln2017', 'dxatspecimenacquisition', 'vitalstatus', 'overallsurvival', '%.blasts.in.bm'
                               '%.blasts.in.pb','flt3-itd', 'npm1']])

clsm_cut.head(5)
```

Out[28]:

	labid	patientid	consensus_sex	inferred_ethnicity	isrelapse	istransformed	priormalignancynonmyeloid	priormds	priormdsmpn	p
0	09-00705	163	Male	White	False	False	n	n	n	
1	10-00136	174	Male	White	False	False	n	n	n	
2	10-00172	175	Female	White	False	False	n	n	n	
3	10-00507	45	Female	White	False	False	n	n	n	
4	10-00542	174	Male	White	True	False	n	n	n	

In [29]:
```
clsm_cut.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 672 entries, 0 to 671
Data columns (total 18 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   labid                       672 non-null    object
 1   patientid                   672 non-null    int64
 2   consensus_sex               672 non-null    object
 3   inferred_ethnicity          670 non-null    object
 4   isrelapse                   672 non-null    bool
 5   istransformed               672 non-null    bool
 6   priormalignancynonmyeloid   672 non-null    object
 7   priormds                    672 non-null    object
 8   priormdsmpn                 672 non-null    object
 9   priormpn                    672 non-null    object
 10  eln2017                     672 non-null    object
 11  dxatspecimenacquisition     672 non-null    object
 12  vitalstatus                 672 non-null    object
 13  overallsurvival             607 non-null    float64
 14  %.blasts.in.bm              459 non-null    object
 15  %.blasts.in.pb              451 non-null    object
 16  flt3-itd                    670 non-null    object
 17  npm1                        669 non-null    object
dtypes: bool(2), float64(1), int64(1), object(14)
memory usage: 85.4+ KB
```

In [30]:
```
clsm_cut.describe()
```

Out[30]:

	patientid	overallsurvival
count	672.000000	607.000000
mean	2088.020833	441.881384
std	973.372734	479.180429
min	17.000000	-1.000000
25%	1450.750000	167.000000
50%	2016.000000	323.000000
75%	2501.500000	555.000000
max	4380.000000	5305.000000

Attribute Information

% Blasts Attributes Numerical Prep

%.blasts.in.bm Attribute:

In [31]:
```
clsm_cut['%.blasts.in.bm'].unique()
```

Out[31]:
```
array(['94', '80', '91', '97', '87', nan, '40', '75', '83', '95', '85',
       '90', '70', '92', '72', '68', '88', '36', '81', '93', '34', '77.5',
       '46', '65', '50', '76', '71', '60', '73', '55', '0.5', '30', '62',
       '18', '82', '28', '41', '64', '84', '21', '51', '17', '49.4', '32',
       '29', '25', '59.3', '66', '20', '52', '54', '22', '10', '12', '13',
       '67', '39', '25.9', '45', '37', '78', '8', '3', '54.8', '74', '96',
       '4', '86.1', '42', '56', '69', '79', '33', '9', '0.4', '51.5',
       '15', '5', '24', '7', '2', '6', '1', '58', '>50', '35', '86',
       '93.2', '0', '27', '89.6', '23', '98', '19', '91.8', '>95', '57',
       '71.5', '78.3', '63', '1.5', '53.74', '59.5', '44', '42.5', '26',
       '3.5', '48', '26.3', '47', '88.5'], dtype=object)
```

In [32]:
```
# > and < will be changed to whole numbers less than or greater than.
clsm_cut['%.blasts.in.bm'] = clsm_cut['%.blasts.in.bm'].replace(['>50'], 51)
clsm_cut['%.blasts.in.bm'] = clsm_cut['%.blasts.in.bm'].replace(['>95'], 96)

clsm_cut['%.blasts.in.bm'].unique()
```

Out[32]:
```
array(['94', '80', '91', '97', '87', nan, '40', '75', '83', '95', '85',
       '90', '70', '92', '72', '68', '88', '36', '81', '93', '34', '77.5',
       '46', '65', '50', '76', '71', '60', '73', '55', '0.5', '30', '62',
       '18', '82', '28', '41', '64', '84', '21', '51', '17', '49.4', '32',
       '29', '25', '59.3', '66', '20', '52', '54', '22', '10', '12', '13',
       '67', '39', '25.9', '45', '37', '78', '8', '3', '54.8', '74', '96',
       '4', '86.1', '42', '56', '69', '79', '33', '9', '0.4', '51.5',
       '15', '5', '24', '7', '2', '6', '1', '58', 51, '35', '86', '93.2',
       '0', '27', '89.6', '23', '98', '19', '91.8', 96, '57', '71.5',
       '78.3', '63', '1.5', '53.74', '59.5', '44', '42.5', '26', '3.5',
       '48', '26.3', '47', '88.5'], dtype=object)
```

%.blasts.in.pb Attribute:

In [33]:
```
clsm_cut['%.blasts.in.pb'].unique()
```

Out[33]:
```
array(['97', '19', '99', '80', nan, '51', '30', '41', '84', '77', '75',
       '63', '60', '96', '66', '45', '93', '9', '82', '15', '33', '0',
       '13', '94', '89', '83', '>90', '78', '72', '59', '32', '6', '29',
       '24', '64', '57', '52', '2.1', '<5', '17', '22', '5', '47', '56',
       '25', '23', '42', '65', '71', '8', '3.5', '66.3', '95', '44', '10',
       '28.6', '18', '58', '67', '40', '92', '54', '1', '2', '20', '28',
       '35', '85', '42.4', '16', '49.1', '14', '88', '46', '7', '0.5',
       '79', '26', '87', '20.4', '68', '48', '5.3', '61', '90', '17.4',
       '57.4', '43.8', '50', '37', '4', '3', '12', '81', '11', '90.5',
       '"rare"', '90.2', '55', 'rare', '39', '31', '86', '47.4', '27.4',
       '39.6', '12.9', '15.4', '9.5', '62', '64.6', '27.8', '69.14',
       '52.2', '91', '67.25', '49', '23.7', '48.6', '98', '74.8', '2.6',
       '43', '29.6', '47.5', '38', '2.5', '25.2', '3.56', '70', '99.2',
       '73', '26.7', '38.5', '7.7', '74', '93.3', '12.1', '11.2', '92.9',
       '98.4', '6.8', '10.5', '53', '3.1', '28.9', '72.9', '40.2', '3.3',
       '42.1', '11.5', '77.8', '3.8', '59.5', '21.7', '53.2'],
      dtype=object)
```

In [34]:
```
#%.Blasts.in.PB attribute has 1 "rare" record with no flt3 nor npm1 input. This will be changed to NAN
clsm_cut['%.blasts.in.pb'] = clsm_cut['%.blasts.in.pb'].replace(['"""rare"""'], np.nan)
clsm_cut['%.blasts.in.pb'] = clsm_cut['%.blasts.in.pb'].replace(['"rare"'], np.nan)
clsm_cut['%.blasts.in.pb'] = clsm_cut['%.blasts.in.pb'].replace(['rare'], np.nan)
# > and < will be changed to whole numbers less than or greater than.
clsm_cut['%.blasts.in.pb'] = clsm_cut['%.blasts.in.pb'].replace(['<5'], 4)
clsm_cut['%.blasts.in.pb'] = clsm_cut['%.blasts.in.pb'].replace(['>90'], 91)

clsm_cut['%.blasts.in.pb'].unique()
```

Out[34]:
```
array(['97', '19', '99', '80', nan, '51', '30', '41', '84', '77', '75',
       '63', '60', '96', '66', '45', '93', '9', '82', '15', '33', '0',
       '13', '94', '89', '83', 91, '78', '72', '59', '32', '6', '29',
       '24', '64', '57', '52', '2.1', 4, '17', '22', '5', '47', '56',
       '25', '23', '42', '65', '71', '8', '3.5', '66.3', '95', '44', '10',
       '28.6', '18', '58', '67', '40', '92', '54', '1', '2', '20', '28',
       '35', '85', '42.4', '16', '49.1', '14', '88', '46', '7', '0.5',
       '79', '26', '87', '20.4', '68', '48', '5.3', '61', '90', '17.4',
       '57.4', '43.8', '50', '37', '4', '3', '12', '81', '11', '90.5',
       '90.2', '55', '39', '31', '86', '47.4', '27.4', '39.6', '12.9',
       '15.4', '9.5', '62', '64.6', '27.8', '69.14', '52.2', '91',
       '67.25', '49', '23.7', '48.6', '98', '74.8', '2.6', '43', '29.6',
       '47.5', '38', '2.5', '25.2', '3.56', '70', '99.2', '73', '26.7',
       '38.5', '7.7', '74', '93.3', '12.1', '11.2', '92.9', '98.4', '6.8',
       '10.5', '53', '3.1', '28.9', '72.9', '40.2', '3.3', '42.1', '11.5',
       '77.8', '3.8', '59.5', '21.7', '53.2'], dtype=object)
```

From Categorical to Numerical

Transform %.blasts.in.bm and %.blasts.in.pb from object to float:

In [35]:
```
clsm_cut['%.blasts.in.bm'] = clsm_cut['%.blasts.in.bm'].astype(float)
clsm_cut['%.blasts.in.pb'] = clsm_cut['%.blasts.in.pb'].astype(float)


clsm_cut.info()
```

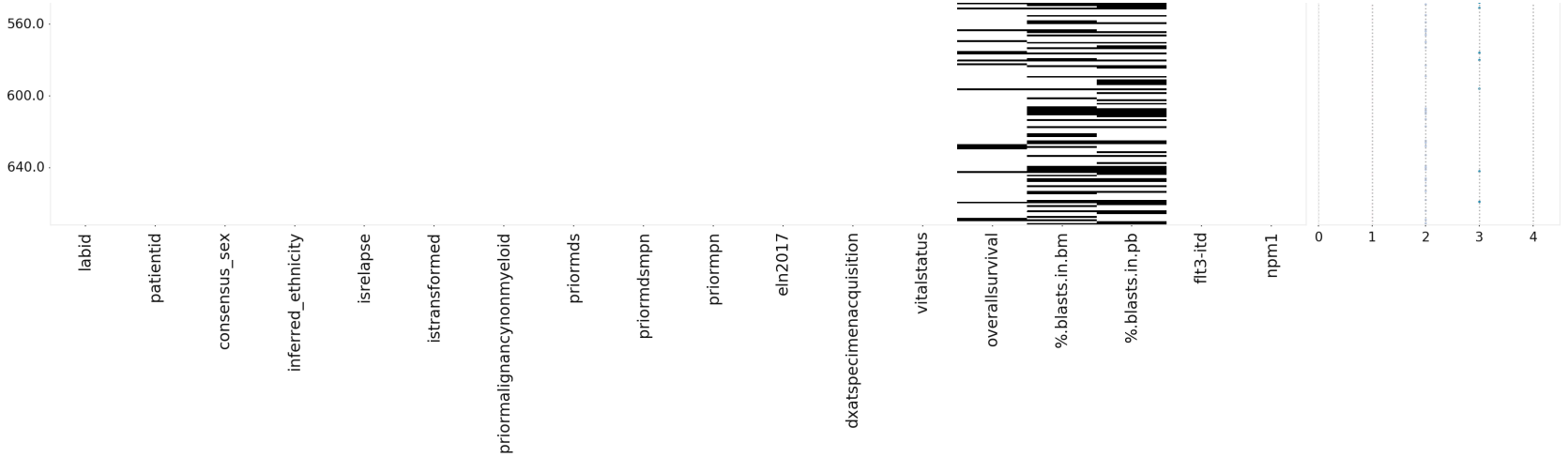
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 672 entries, 0 to 671
Data columns (total 18 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   labid                       672 non-null    object
 1   patientid                   672 non-null    int64
 2   consensus_sex               672 non-null    object
 3   inferred_ethnicity          670 non-null    object
 4   isrelapse                   672 non-null    bool
 5   istransformed               672 non-null    bool
 6   priormalignancynonmyeloid   672 non-null    object
 7   priormds                    672 non-null    object
 8   priormdsmpn                 672 non-null    object
 9   priormpn                    672 non-null    object
 10  eln2017                     672 non-null    object
 11  dxatspecimenacquisition     672 non-null    object
 12  vitalstatus                 672 non-null    object
 13  overallsurvival             607 non-null    float64
 14  %.blasts.in.bm              459 non-null    float64
 15  %.blasts.in.pb              448 non-null    float64
 16  flt3-itd                    670 non-null    object
 17  npm1                        669 non-null    object
dtypes: bool(2), float64(3), int64(1), object(12)
memory usage: 85.4+ KB
```

clsm_cut Identify Missing Values

In [36]:
```
klib.missingval_plot(clsm_cut)
```

Out[36]:
```
GridSpec(6, 6)
```

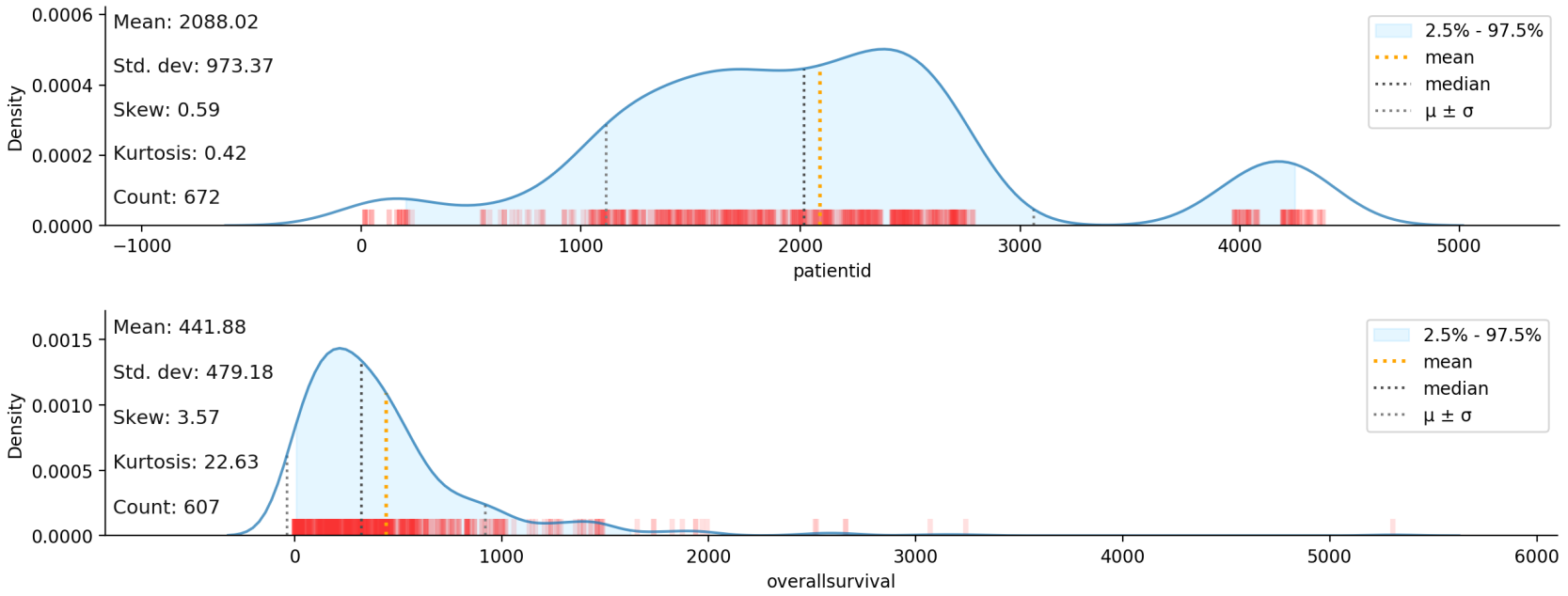
Missing value plot

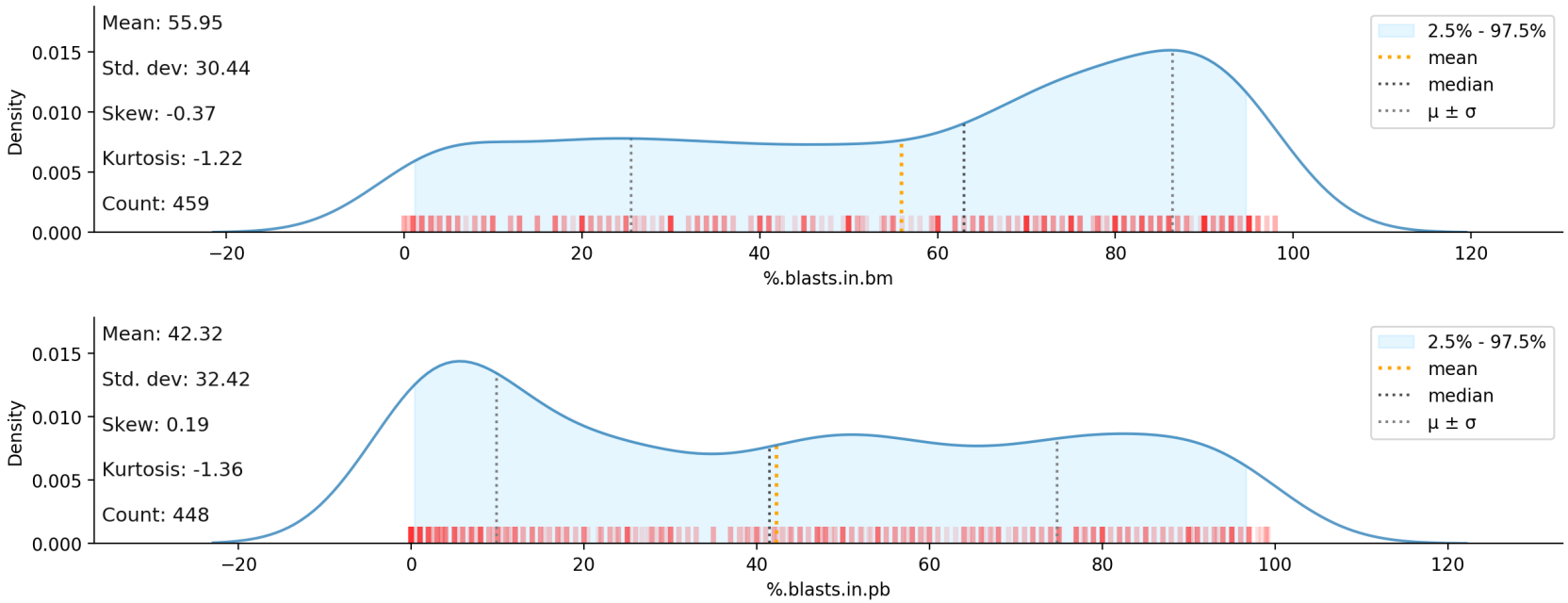


Replace Missing Values

In [37]:
```
klib.dist_plot(clsm_cut)
```

Out[37]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fceaa50a350>
```





In [38]:
```
clsm_cut.describe()
```

Out[38]:

	patientid	overallsurvival	%.blasts.in.bm	%.blasts.in.pb
count	672.000000	607.000000	459.000000	448.000000
mean	2088.020833	441.881384	55.949325	42.316629
std	973.372734	479.180429	30.440925	32.418249
min	17.000000	-1.000000	0.000000	0.000000
25%	1450.750000	167.000000	30.000000	10.000000
50%	2016.000000	323.000000	63.000000	41.500000
75%	2501.500000	555.000000	83.000000	72.000000
max	4380.000000	5305.000000	98.000000	99.200000

In [39]:
```
#From distibution, skewness suggest median is the best representation.
clsm_cut['overallsurvival'] = clsm_cut['overallsurvival'].fillna(clsm_cut['overallsurvival'].median())
clsm_cut['%.blasts.in.bm'] = clsm_cut['%.blasts.in.bm'].fillna(clsm_cut['%.blasts.in.bm'].median())
clsm_cut['%.blasts.in.pb'] = clsm_cut['%.blasts.in.pb'].fillna(clsm_cut['%.blasts.in.pb'].median())
```

In [40]:
```
#Replace categorical NaN with unknown
clsm_cut = clsm_cut.replace(np.nan, 'unknown', regex=True)
```

In [41]:
```
#Determine mode of inferred_ethnicity:
clsm_cut['inferred_ethnicity'].mode()
```

Out[41]:
```
0    White
dtype: object
```

In [42]:
```
#In inferred_ethnicity, replace mode of unknown to white:
clsm_cut['inferred_ethnicity'] = clsm_cut['inferred_ethnicity'].replace(['unknown'], 'white')

clsm_cut['inferred_ethnicity'].unique()
```

Out[42]:
```
array(['White', 'HispNative', 'AdmixedBlack', 'Asian', 'Black',
       'AdmixedAsian', 'white', 'AdmixedWhite', 'AdmixedHispNative'],
      dtype=object)
```

In [43]:
```
#Determine mode of flt3-itd:
clsm_cut['flt3-itd'].mode()
```

Out[43]:
```
0    negative
dtype: object
```

In [44]:
```
#In flt3-itd, replace mode of unknown to negative:
clsm_cut['flt3-itd'] = clsm_cut['flt3-itd'].replace(['unknown'], 'negative')

clsm_cut['flt3-itd'].unique()
```

Out[44]:
```
array(['positive', 'negative'], dtype=object)
```

In [45]:
```
#Determine mode of npm1:
clsm_cut['npm1'].mode()
```

Out[45]:
```
0    negative
dtype: object
```

In [46]:
```
#In npm1, replace mode of unknown to negative:
clsm_cut['npm1'] = clsm_cut['npm1'].replace(['unknown'], 'negative')

clsm_cut['npm1'].unique()
```

Out[46]:
```
array(['positive', 'negative'], dtype=object)
```

In [47]:
```
klib.missingval_plot(clsm_cut)
```

```
No missing values found in the dataset.
```

In [48]:
```
clsm_cut.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 672 entries, 0 to 671
Data columns (total 18 columns):
 #   Column                      Non-Null Count  Dtype  
---  ------                      --------------  -----  
 0   labid                       672 non-null    object 
 1   patientid                   672 non-null    int64  
 2   consensus_sex               672 non-null    object 
 3   inferred_ethnicity          672 non-null    object 
 4   isrelapse                   672 non-null    bool   
 5   istransformed               672 non-null    bool   
 6   priormalignancynonmyeloid   672 non-null    object 
 7   priormds                    672 non-null    object 
 8   priormdsmpn                 672 non-null    object 
 9   priormpn                    672 non-null    object 
 10  eln2017                     672 non-null    object 
 11  dxatspecimenacquisition     672 non-null    object 
 12  vitalstatus                 672 non-null    object 
 13  overallsurvival             672 non-null    float64
 14  %.blasts.in.bm              672 non-null    float64
 15  %.blasts.in.pb              672 non-null    float64
 16  flt3-itd                    672 non-null    object 
 17  npm1                        672 non-null    object 
dtypes: bool(2), float64(3), int64(1), object(12)
memory usage: 85.4+ KB
```

Check for Duplicates

In [49]:
```
clsm_cut = clsm_cut.drop_duplicates(ignore_index=True)
clsm_cut.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 672 entries, 0 to 671
Data columns (total 18 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   labid                       672 non-null    object
 1   patientid                   672 non-null    int64
 2   consensus_sex               672 non-null    object
 3   inferred_ethnicity          672 non-null    object
 4   isrelapse                   672 non-null    bool
 5   istransformed               672 non-null    bool
 6   priormalignancynonmyeloid   672 non-null    object
 7   priormds                    672 non-null    object
 8   priormdsmpn                 672 non-null    object
 9   priormpn                    672 non-null    object
 10  eln2017                     672 non-null    object
 11  dxatspecimenacquisition     672 non-null    object
 12  vitalstatus                 672 non-null    object
 13  overallsurvival             672 non-null    float64
 14  %.blasts.in.bm              672 non-null    float64
 15  %.blasts.in.pb              672 non-null    float64
 16  flt3-itd                    672 non-null    object
 17  npm1                        672 non-null    object
dtypes: bool(2), float64(3), int64(1), object(12)
memory usage: 85.4+ KB
```

Transformation (Final Preparation before Data Modeling)

Create Target Variable

In [50]:
```
clsm_cut['dxatspecimenacquisition'].value_counts()
```

Out[50]:
```
ACUTE MYELOID LEUKAEMIA (AML) AND RELATED PRECURSOR NEOPLASMS    646
MYELODYSPLASTIC SYNDROMES                                         15
MYELODYSPLASTIC/MYELOPROLIFERATIVE NEOPLASMS                       4
ACUTE LEUKAEMIAS OF AMBIGUOUS LINEAGE                             3
MYELOPROLIFERATIVE NEOPLASMS                                      3
MATURE B-CELL NEOPLASMS                                           1
Name: dxatspecimenacquisition, dtype: int64
```

In [51]:
```
#create column for AML detected
clsm_cut['AML_detected'] = ['yes' if x == 'ACUTE MYELOID LEUKAEMIA (AML) AND RELATED PRECURSOR NEOPLASMS'
                            else 'no' for x in clsm_cut['dxatspecimenacquisition']]
```

In [52]:
```
clsm_cut.head()
```

Out[52]:

	labid	patientid	consensus_sex	inferred_ethnicity	isrelapse	istransformed	priormalignancynonmyeloid	priormds	priormdsmpn	p
0	09-00705	163	Male	White	False	False	n	n	n	
1	10-00136	174	Male	White	False	False	n	n	n	
2	10-00172	175	Female	White	False	False	n	n	n	
3	10-00507	45	Female	White	False	False	n	n	n	
4	10-00542	174	Male	White	True	False	n	n	n	

Transform select categorical attributes to numerical:

In [53]:
```
#AML_detected
clsm_cut['AML_detected'].replace(['no', 'yes'],
                         [0, 1], inplace=True)

#npm1
clsm_cut['npm1'].replace(['negative', 'positive'],
                         [0, 1], inplace=True)

#flt3-itd
clsm_cut['flt3-itd'].replace(['negative', 'positive'],
                         [0, 1], inplace=True)

#priormalignancynonmyeloid
clsm_cut['priormalignancynonmyeloid'].replace(['n', 'y'],
                         [0, 1], inplace=True)

#priormds
clsm_cut['priormds'].replace(['y', 'n'],
                         [1, 0], inplace=True)

#priormdsmpn
clsm_cut['priormdsmpn'].replace(['n', 'y'],
                         [0, 1], inplace=True)

#priormpn
clsm_cut['priormpn'].replace(['n', 'y'],
                         [0, 1], inplace=True)

#isrelapse
clsm_cut['isrelapse'].replace(['False', 'True'],
                         [0, 1], inplace=True)

#istransformed
clsm_cut['istransformed'].replace(['True', 'False'],
                         [1, 0], inplace=True)
```

In [54]:
```
clsm_t = pd.DataFrame(clsm_cut[['AML_detected', 'npm1', 'flt3-itd', 'isrelapse', 'istransformed',
                                'priormalignancynonmyeloid', 'priormds', 'priormdsmpn', 'priormpn',
                                '%.blasts.in.pb', '%.blasts.in.bm', 'overallsurvival']])
```

In [55]:
```
#Transform data type:
clsm_t['npm1'] = clsm_cut['npm1'].astype(int)
clsm_t['flt3-itd'] = clsm_cut['flt3-itd'].astype(int)

clsm_t['isrelapse'] = clsm_cut['isrelapse'].astype(int)
clsm_t['istransformed'] = clsm_cut['istransformed'].astype(int)
```

One Hot encoding

In [56]:
```
clsm_t = pd.get_dummies(clsm_t, columns= ['npm1', 'flt3-itd', 'priormalignancynonmyeloid',
                                          'priormds', 'priormdsmpn', 'priormpn', 'isrelapse', 'istransformed'
```

In [57]:
```
clsm_t.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 672 entries, 0 to 671
Data columns (total 20 columns):
 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   AML_detected                  672 non-null    int64
 1   %.blasts.in.pb                672 non-null    float64
 2   %.blasts.in.bm                672 non-null    float64
 3   overallsurvival               672 non-null    float64
 4   npm1_0                        672 non-null    uint8
 5   npm1_1                        672 non-null    uint8
 6   flt3-itd_0                    672 non-null    uint8
 7   flt3-itd_1                    672 non-null    uint8
 8   priormalignancynonmyeloid_0   672 non-null    uint8
 9   priormalignancynonmyeloid_1   672 non-null    uint8
 10  priormds_0                    672 non-null    uint8
 11  priormds_1                    672 non-null    uint8
 12  priormdsmpn_0                 672 non-null    uint8
 13  priormdsmpn_1                 672 non-null    uint8
 14  priormpn_0                    672 non-null    uint8
 15  priormpn_1                    672 non-null    uint8
 16  isrelapse_0                   672 non-null    uint8
 17  isrelapse_1                   672 non-null    uint8
 18  istransformed_0               672 non-null    uint8
 19  istransformed_1               672 non-null    uint8
dtypes: float64(3), int64(1), uint8(16)
memory usage: 31.6 KB
```

In [58]:
```
clsm_t.head()
```

Out[58]:

	AML_detected	%.blasts.in.pb	%.blasts.in.bm	overallsurvival	npm1_0	npm1_1	flt3-itd_0	flt3-itd_1	priormalignancynonmyeloid_0	priormalign
0	1	97.0	94.0	425.0	0	1	0	1	1	
1	1	19.0	80.0	419.0	1	0	0	1	1	
2	1	99.0	91.0	541.0	1	0	0	1	1	
3	1	97.0	97.0	511.0	0	1	0	1	1	
4	1	80.0	87.0	419.0	1	0	0	1	1	

Transform Headers

In [59]:
```
clsm_t = clsm_t.rename(columns={ '%.blasts.in.pb': 'Feature_1', '%.blasts.in.bm': 'Feature_2',
                                 'overallsurvival': 'Feature_3',
                                  'npm1_0': 'Feature_4', 'npm1_1': 'Feature_5',
                                 'flt3-itd_0': 'Feature_6', 'flt3-itd_1': 'Feature_7',
                                  'priormalignancynonmyeloid_0': 'Feature_8', 'priormalignancynonmyeloid_1'
                                  'priormds_0': 'Feature_10', 'priormds_1': 'Feature_11',
                                 'priormdsmpn_0': 'Feature_12', 'priormdsmpn_1': 'Feature_13',
                                  'priormpn_0': 'Feature_14', 'priormpn_1': 'Feature_15',
                                  'isrelapse_0': 'Feature_16', 'isrelapse_1': 'Feature_17',
                                 'istransformed_0': 'Feature_18', 'istransformed_1': 'Feature_19' })
```

In [60]:
```
clsm_t.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 672 entries, 0 to 671
Data columns (total 20 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   AML_detected  672 non-null    int64  
 1   Feature_1     672 non-null    float64
 2   Feature_2     672 non-null    float64
 3   Feature_3     672 non-null    float64
 4   Feature_4     672 non-null    uint8  
 5   Feature_5     672 non-null    uint8  
 6   Feature_6     672 non-null    uint8  
 7   Feature_7     672 non-null    uint8  
 8   Feature_8     672 non-null    uint8  
 9   Feature_9     672 non-null    uint8  
 10  Feature_10    672 non-null    uint8  
 11  Feature_11    672 non-null    uint8  
 12  Feature_12    672 non-null    uint8  
 13  Feature_13    672 non-null    uint8  
 14  Feature_14    672 non-null    uint8  
 15  Feature_15    672 non-null    uint8  
 16  Feature_16    672 non-null    uint8  
 17  Feature_17    672 non-null    uint8  
 18  Feature_18    672 non-null    uint8  
 19  Feature_19    672 non-null    uint8  
dtypes: float64(3), int64(1), uint8(16)
memory usage: 31.6 KB
```

Auto ML

In [61]:
```
df_automl = clsm_t
clsm_t.shape
```

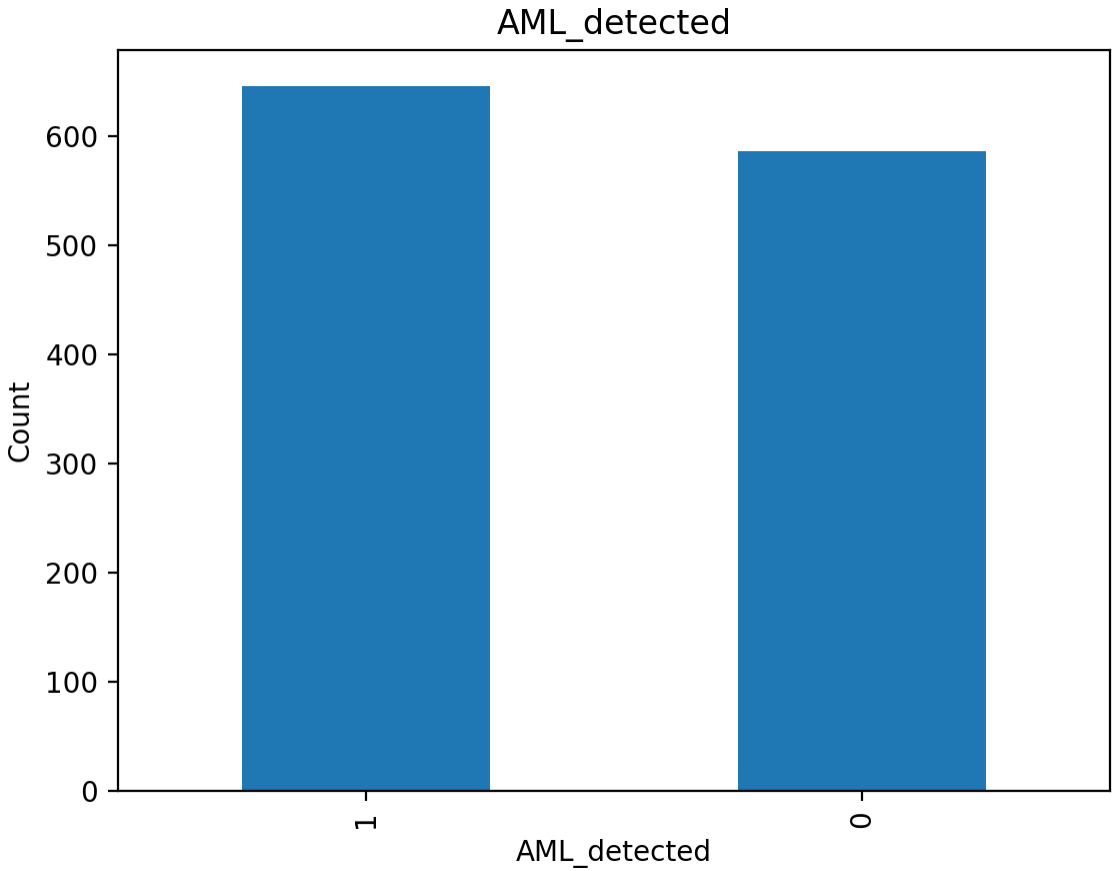
Out[61]:
```
(672, 20)
```

Balance the data set

In [62]:
```
#resampling of training data set
to_resample= df_automl.loc[df_automl["AML_detected"] == 0] #isolate all records of AML_detected
our_resample=to_resample.sample(n=560, replace=True) #sample w/ replacement
df_rebal=pd.concat([df_automl, our_resample]) #combine original training set w/ resampled records
df_rebal["AML_detected"].value_counts()
```

Out[62]:
```
1    646
0    586
Name: AML_detected, dtype: int64
```

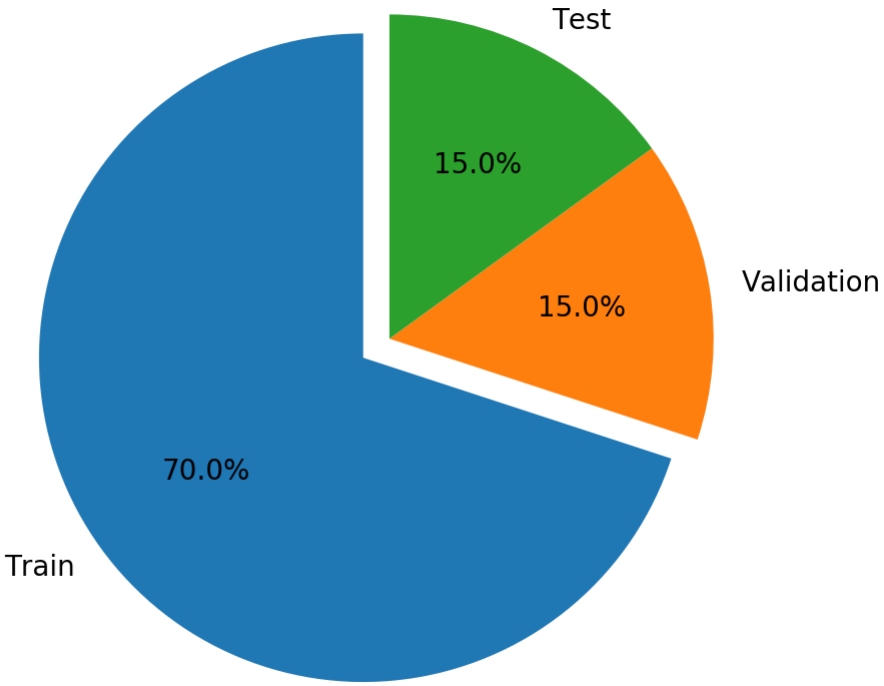
In [63]:
```
import matplotlib.pyplot as plt

df_rebal["AML_detected"].value_counts().plot(kind="bar", title="AML_detected")
plt.xlabel("AML_detected")
plt.ylabel("Count")

plt.show()
```



Split into Train, Validation, and Test Sets

In [64]:
```
from sklearn.model_selection import train_test_split

# Split all data into 70% train and 30% holdout
df_train, df_holdout = train_test_split(df_rebal, test_size=0.30, stratify=df_rebal["AML_detected"])

# Split holdout data into 50% validation and 50% test
df_validation, df_test = train_test_split(df_holdout, test_size=0.50, stratify=df_holdout["AML_detected"])
```

In [65]:
```
# Pie chart, where the slices will be ordered and plotted counter-clockwise:
labels = ["Train", "Validation", "Test"]
sizes = [len(df_train.index), len(df_validation.index), len(df_test.index)]
explode = (0.1, 0, 0)

fig1, ax1 = plt.subplots()

ax1.pie(sizes, explode=explode, labels=labels, autopct="%1.1f%%", startangle=90)

# Equal aspect ratio ensures that pie is drawn as a circle.
ax1.axis("equal")

plt.show()
```

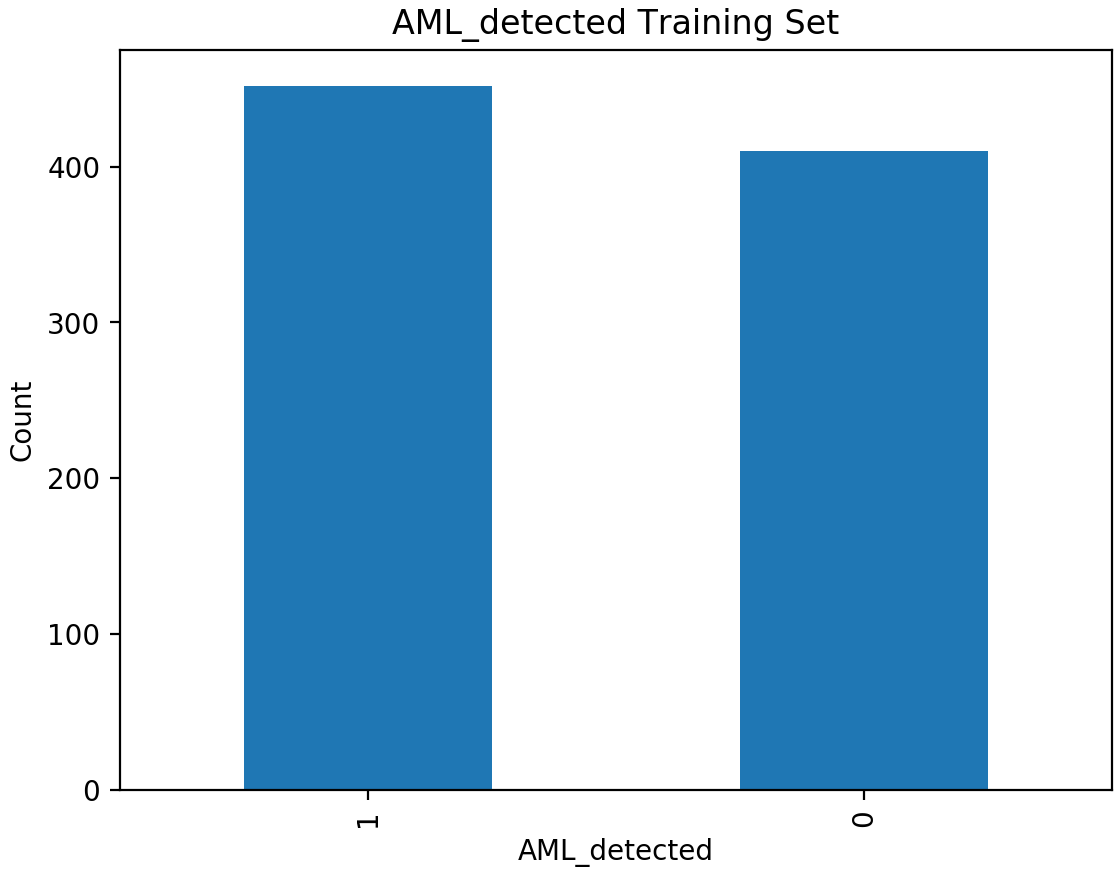


Show 70% Train Data Split

In [66]:
```
df_train.shape
```

Out[66]: (862, 20)

In [67]:
```
df_train["AML_detected"].value_counts().plot(kind="bar", title="AML_detected Training Set")
plt.xlabel("AML_detected")
plt.ylabel("Count")

plt.show()
```

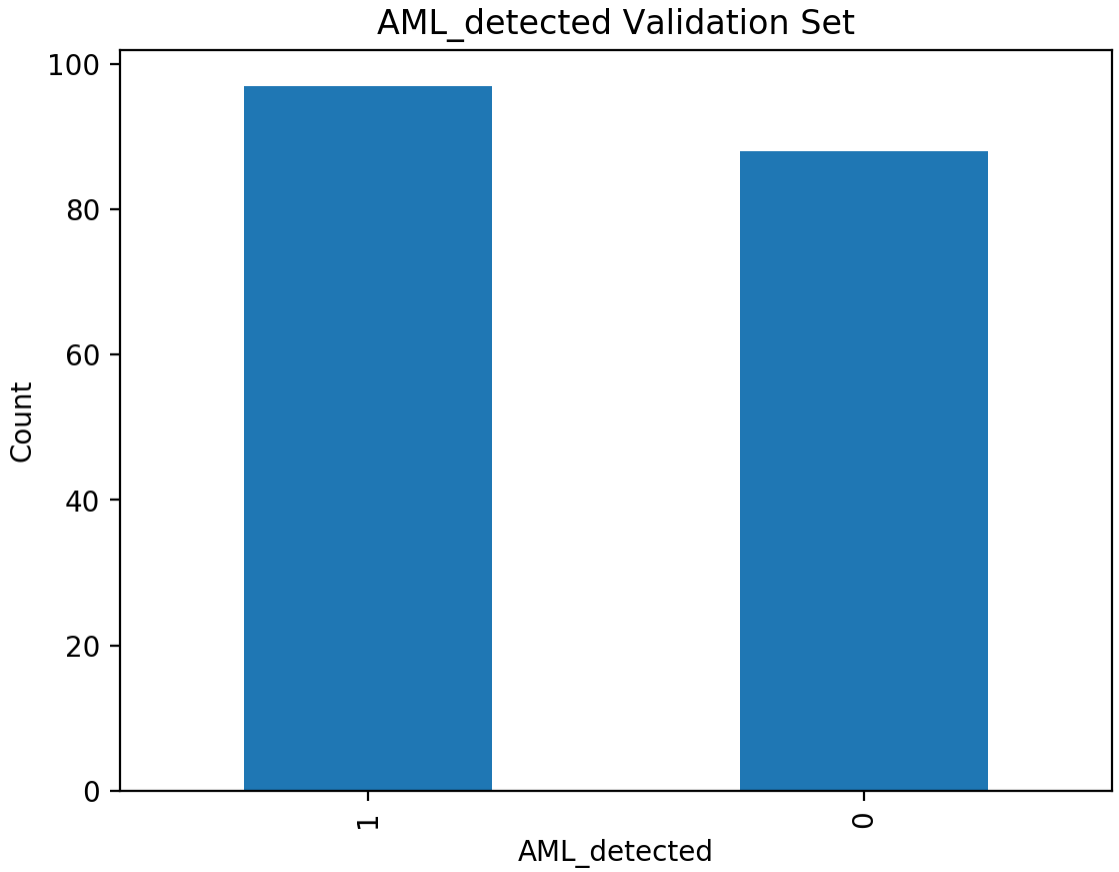


Show 15% Validation Split

In [68]:
```
df_validation.shape
```

Out[68]: (185, 20)

In [69]:
```
df_validation["AML_detected"].value_counts().plot(kind="bar", title="AML_detected Validation Set")
plt.xlabel("AML_detected")
plt.ylabel("Count")

plt.show()
```

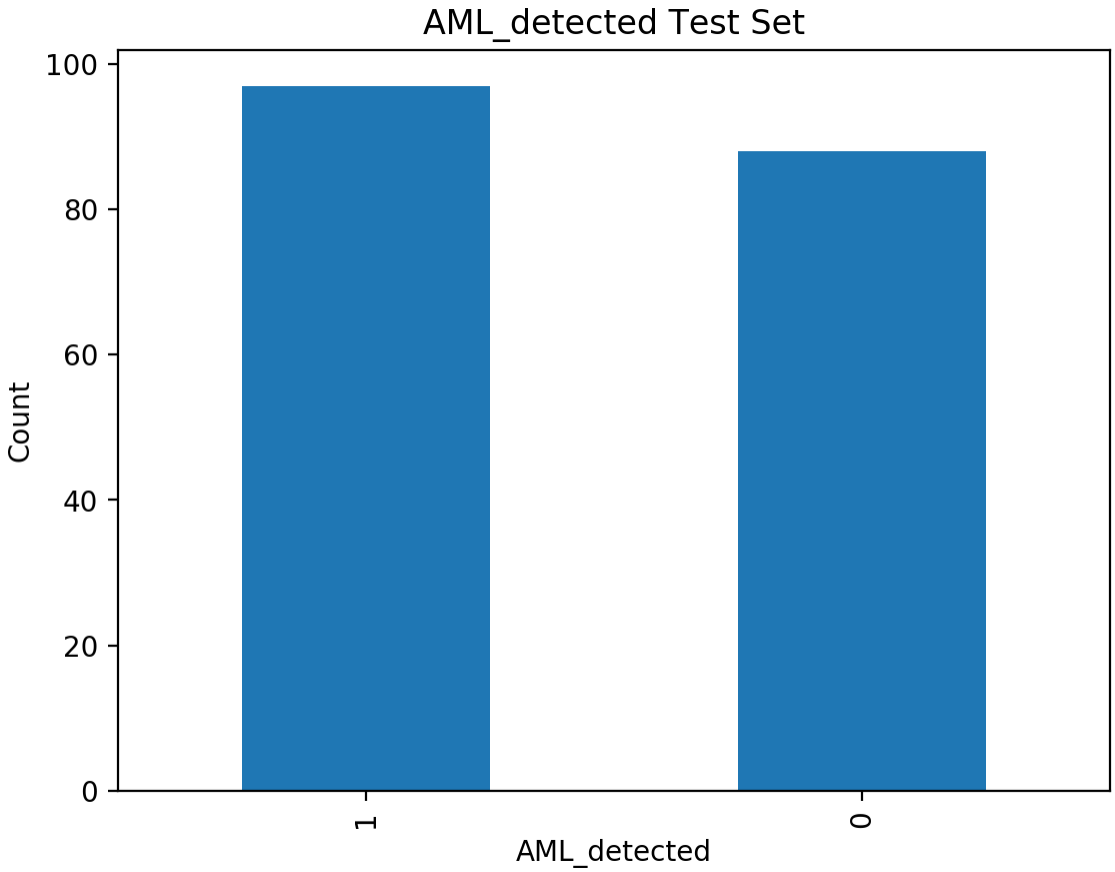


Show 15% Test Data Split

In [70]:
```
df_test.shape
```

Out[70]:
```
(185, 20)
```

In [71]:
```
df_test["AML_detected"].value_counts().plot(kind="bar", title="AML_detected Test Set")
plt.xlabel("AML_detected")
plt.ylabel("Count")

plt.show()
```



Write a Train CSV with Header for Autopilot

In [72]:
```
autopilot_train_path = "./df_autopilot.csv"
df_train.to_csv(autopilot_train_path, index=False, header=True)
```

Upload Train Data to S3 for Autopilot

In [73]:
```
train_s3_prefix = "data"
autopilot_train_s3_uri = sess.upload_data(path=autopilot_train_path, key_prefix=train_s3_prefix)
autopilot_train_s3_uri
```

Out[73]:
```
's3://sagemaker-us-east-1-614093401978/data/df_autopilot.csv'
```

In [74]:
```
!aws s3 ls $autopilot_train_s3_uri
```

```
2023-04-17 03:55:43      42710 df_autopilot.csv
```

Store Variables

In [75]:
```
%store autopilot_train_s3_uri
```

```
Stored 'autopilot_train_s3_uri' (str)
```

In [76]:
```
%store
```

```
Stored variables and their in-db values:
auto_ml_job_name                          -> 'automl-dm-16-20-10-05'
autopilot_endpoint_arn                    -> 'arn:aws:sagemaker:us-east-1:614093401978:endpoint
autopilot_endpoint_name                   -> 'automl-dm-ep-16-22-24-43'
autopilot_model_arn                       -> 'arn:aws:sagemaker:us-east-1:614093401978:model/au
autopilot_model_name                      -> 'automl-dm-model-16-22-20-58'
autopilot_train_s3_uri                    -> 's3://sagemaker-us-east-1-614093401978/data/df_aut
ingest_create_athena_db_passed            -> True
s3_private_path_csv                       -> 's3://sagemaker-us-east-1-614093401978/cell_data'
s3_public_path_clsm                       -> 's3://team4rawdatasets/CSV/Input/OHSU_BeatAML_Clin
s3_public_path_csv                        -> 's3://gdc-beataml1.0-crenolanib-phs001628-2-open/'
s3_public_path_pi                         -> 's3://team4rawdatasets/CSV/Input/OpenCell_ProteinI
setup_dependencies_passed                 -> True
setup_iam_roles_passed                    -> True
setup_instance_check_passed               -> True
setup_s3_bucket_passed                    -> True
```

Train AML Detection

Training Data

In [77]:
```
print(autopilot_train_s3_uri)
```

```
s3://sagemaker-us-east-1-614093401978/data/df_autopilot.csv
```

In [78]:
```
!aws s3 ls $autopilot_train_s3_uri
```

```
2023-04-17 03:55:43      42710 df_autopilot.csv
```

See our prepared traiing data which we use as input for Autopilot

In [79]:
```
!aws s3 cp $autopilot_train_s3_uri ./tmp/
```

```
download: s3://sagemaker-us-east-1-614093401978/data/df_autopilot.csv to tmp/df_autopilot.csv
```

In [80]:
```
import csv

df = pd.read_csv("./tmp/df_autopilot.csv")
df.head()
```

Out[80]:

	AML_detected	Feature_1	Feature_2	Feature_3	Feature_4	Feature_5	Feature_6	Feature_7	Feature_8	Feature_9	Feature_10	Feature_1
0	0	6.0	4.0	286.0	1	0	1	0	1	0	0	
1	0	15.0	63.0	299.0	1	0	1	0	1	0	0	
2	1	41.5	45.0	323.0	1	0	1	0	0	1	1	
3	0	0.0	5.0	189.0	1	0	1	0	1	0	0	
4	1	0.0	51.5	414.0	0	1	1	0	1	0	1	

Setup the S3 location for the Autopilot-Generated Assests

This includes Jupyter Notebooks (analysis), Python Scripts (Feature Engineering), and Trained Models

In [81]:
```
prefix_model_output = "models/autopilot"

model_output_s3_uri = "s3://{}/{}".format(bucket, prefix_model_output)

print(model_output_s3_uri)
```

```
s3://sagemaker-us-east-1-614093401978/models/autopilot
```

In [82]:
```
max_candidates = 3

job_config = {
    "CompletionCriteria": {
        "MaxRuntimePerTrainingJobInSeconds": 900,
        "MaxCandidates": max_candidates,
        "MaxAutoMLJobRuntimeInSeconds": 5400,
    },
}

input_data_config = [
    {
        "DataSource": {"S3DataSource": {"S3DataType": "S3Prefix", "S3Uri": "{}".format(autopilot_train_s3_uri)}},
        "TargetAttributeName": "AML_detected",
    }
]

output_data_config = {"S3OutputPath": "{}".format(model_output_s3_uri)}
```

Check for existing Autopilot jobs

In [83]:
```
existing_jobs_response = sm.list_auto_ml_jobs()
```

In [84]:
```
num_existing_jobs = 0
running_jobs = 0

if "AutoMLJobSummaries" in existing_jobs_response.keys():
    job_list = existing_jobs_response["AutoMLJobSummaries"]
    num_existing_jobs = len(job_list)
    # print('[INFO] You already created {} Autopilot job(s) in this account.'.format(num_existing_jobs))
    for j in job_list:
        if "AutoMLJobStatus" in j.keys():
            if j["AutoMLJobStatus"] == "InProgress":
                running_jobs = running_jobs + 1
    print("[INFO] You have {} Autopilot job(s) currently running << Should be 0 jobs.".format(running_jobs))
else:
    print("[OK] Please continue.")
```

```
[INFO] You have 0 Autopilot job(s) currently running << Should be 0 jobs.
```

Launch Sagemaker Autopilot Job

In [85]:
```
from time import gmtime, strftime, sleep
```

In [86]:
```
%store -r auto_ml_job_name

try:
    auto_ml_job_name
except NameError:
    timestamp_suffix = strftime("%d-%H-%M-%S", gmtime())
    auto_ml_job_name = "automl-dm-" + timestamp_suffix
    print("Created AutoMLJobName: " + auto_ml_job_name)
```

In [87]:
```
print(auto_ml_job_name)
```

```
automl-dm-16-20-10-05
```

In [88]:
```
%store auto_ml_job_name
```

```
Stored 'auto_ml_job_name' (str)
```

In [89]:
```
max_running_jobs = 1

if running_jobs < max_running_jobs:  # Limiting to max. 1 Jobs
    try:
        sm.create_auto_ml_job(
            AutoMLJobName=auto_ml_job_name,
            InputDataConfig=input_data_config,
            OutputDataConfig=output_data_config,
            AutoMLJobConfig=job_config,
            RoleArn=role,
        )
        print("[OK] Autopilot Job {} created.".format(auto_ml_job_name))
        running_jobs = running_jobs + 1
    except:
        print(
            "[INFO] You have already launched an Autopilot job. Please continue see the output of this job.".format(
                running_jobs
            )
        )
else:
    print(
        "[INFO] You have already launched {} Autopilot running job(s). Please continue see the output of the running
            running_jobs
        )
    )
```

```
[INFO] You have already launched an Autopilot job. Please continue see the output of this job.
```

Analyzing Data and Generate Notebooks

In [90]:
```
job_description_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)

while (
    "AutoMLJobStatus" not in job_description_response.keys()
    and "AutoMLJobSecondaryStatus" not in job_description_response.keys()
):
    job_description_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
    print("[INFO] Autopilot Job has not yet started. Please wait. ")
    print(json.dumps(job_description_response, indent=4, sort_keys=True, default=str))
    print("[INFO] Waiting for Autopilot Job to start...")
    sleep(15)

print("[OK] AutoMLJob started.")
```

```
[OK] AutoMLJob started.
```

Review the Sagemaker Processing Jobs

In [91]:
```
from IPython.core.display import display, HTML

display(
    HTML(
        '<b>Review <a target="blank" href="https://console.aws.amazon.com/sagemaker/home?region={}#/processing-jobs/'
            region
        )
    )
)
```

Review Processing Jobs

The next cell will show InProgress for a few minutes

In [92]:
```
%%time

job_status = job_description_response["AutoMLJobStatus"]
job_sec_status = job_description_response["AutoMLJobSecondaryStatus"]

if job_status not in ("Stopped", "Failed"):
    while job_status in ("InProgress") and job_sec_status in ("Starting", "AnalyzingData"):
        job_description_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
        job_status = job_description_response["AutoMLJobStatus"]
        job_sec_status = job_description_response["AutoMLJobSecondaryStatus"]
        print(job_status, job_sec_status)
        sleep(15)
    print("[OK] Data analysis phase completed.\n")

print(json.dumps(job_description_response, indent=4, sort_keys=True, default=str))
```

```
[OK] Data analysis phase completed.

{
    "AutoMLJobArn": "arn:aws:sagemaker:us-east-1:614093401978:automl-job/automl-dm-16-20-10-05",
    "AutoMLJobArtifacts": {
        "CandidateDefinitionNotebookLocation": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-
20-10-05/sagemaker-automl-candidates/automl-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/notebooks/SageM
akerAutopilotCandidateDefinitionNotebook.ipynb",
        "DataExplorationNotebookLocation": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-1
0-05/sagemaker-automl-candidates/automl-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/notebooks/SageMaker
AutopilotDataExplorationNotebook.ipynb"
    },
    "AutoMLJobConfig": {
        "CompletionCriteria": {
            "MaxAutoMLJobRuntimeInSeconds": 5400,
            "MaxCandidates": 3,
            "MaxRuntimePerTrainingJobInSeconds": 900
        }
    },
    "AutoMLJobName": "automl-dm-16-20-10-05",
    "AutoMLJobSecondaryStatus": "Completed",
    "AutoMLJobStatus": "Completed",
    "BestCandidate": {
        "CandidateName": "automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4",
        "CandidateProperties": {
            "CandidateArtifactLocations": {
                "Explainability": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/docu
mentation/explainability/output",
                "ModelInsights": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/docum
entation/model_monitor/output"
            },
            "CandidateMetrics": [
                {
                    "MetricName": "F1",
                    "Set": "Validation",
                    "StandardMetricName": "F1",
                    "Value": 0.995169997215271
                },
                {
                    "MetricName": "LogLoss",
                    "Set": "Validation",
                    "StandardMetricName": "LogLoss",
                    "Value": 0.11816000193357468
                },
```

```
                {
                    "MetricName": "Recall",
                    "Set": "Validation",
                    "StandardMetricName": "Recall",
                    "Value": 1.0
                },
                {
                    "MetricName": "Precision",
                    "Set": "Validation",
                    "StandardMetricName": "Precision",
                    "Value": 0.9904199838638306
                },
                {
                    "MetricName": "AUC",
                    "Set": "Validation",
                    "StandardMetricName": "AUC",
                    "Value": 0.9990599751472473
                },
                {
                    "MetricName": "Accuracy",
                    "Set": "Validation",
                    "StandardMetricName": "Accuracy",
                    "Value": 0.9953600168228149
                },
                {
                    "MetricName": "BalancedAccuracy",
                    "Set": "Validation",
                    "StandardMetricName": "BalancedAccuracy",
                    "Value": 0.9955800175666809
                }
            ]
        },
        "CandidateStatus": "Completed",
        "CandidateSteps": [
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:processing-job/automl-dm-16-20-10-05-d
b-1-2887815310fd4416804d2079485d7d062e1a",
                "CandidateStepName": "automl-dm-16-20-10-05-db-1-2887815310fd4416804d2079485d7d062e1a",
                "CandidateStepType": "AWS::SageMaker::ProcessingJob"
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:training-job/automl-dm-16-20-10-05-dpp
1-1-8411a0fbc81748a9958acf62493120d4d7",
                "CandidateStepName": "automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7",
```

```
                "CandidateStepType": "AWS::SageMaker::TrainingJob"
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:transform-job/automl-dm-16-20-10-05-dp
p1-csv-1-18357715aec34f7eb4c3b75bad6e49",
                "CandidateStepName": "automl-dm-16-20-10-05-dpp1-csv-1-18357715aec34f7eb4c3b75bad6e49",
                "CandidateStepType": "AWS::SageMaker::TransformJob"
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:training-job/automl-dm-16-20-10-054If5
RqjsyAN-001-61c635e4",
                "CandidateStepName": "automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4",
                "CandidateStepType": "AWS::SageMaker::TrainingJob"
            }
        ],
        "CreationTime": "2023-04-16 20:33:00+00:00",
        "EndTime": "2023-04-16 20:38:38+00:00",
        "FinalAutoMLJobObjectiveMetric": {
            "MetricName": "validation:f1_binary",
            "StandardMetricName": "F1",
            "Value": 0.995169997215271
        },
        "InferenceContainers": [
            {
                "Environment": {
                    "AUTOML_TRANSFORM_MODE": "feature-transform",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "application/x-recordio-protobuf",
                    "SAGEMAKER_PROGRAM": "sagemaker_serve",
                    "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-p
rocessor-models/automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz"
            },
            {
                "Environment": {
                    "MAX_CONTENT_LENGTH": "20971520",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
                    "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,probabilities"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.3-1-cpu-py3",
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/tuning
/automl-dm--dpp1-xgb/automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4/output/model.tar.gz"
```

```
            },
            {
                "Environment": {
                    "AUTOML_TRANSFORM_MODE": "inverse-label-transform",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
                    "SAGEMAKER_INFERENCE_INPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,labels,probabilities",
                    "SAGEMAKER_PROGRAM": "sagemaker_serve",
                    "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-p
rocessor-models/automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz"
            }
        ],
        "LastModifiedTime": "2023-04-16 20:39:21.759000+00:00",
        "ObjectiveStatus": "Succeeded"
    },
    "CreationTime": "2023-04-16 20:14:58.074000+00:00",
    "EndTime": "2023-04-16 20:47:31.359000+00:00",
    "GenerateCandidateDefinitionsOnly": false,
    "InputDataConfig": [
        {
            "ChannelType": "training",
            "ContentType": "text/csv;header=present",
            "DataSource": {
                "S3DataSource": {
                    "S3DataType": "S3Prefix",
                    "S3Uri": "s3://sagemaker-us-east-1-614093401978/data/df_autopilot.csv"
                }
            },
            "TargetAttributeName": "AML_detected"
        }
    ],
    "LastModifiedTime": "2023-04-16 20:47:31.403000+00:00",
    "OutputDataConfig": {
        "S3OutputPath": "s3://sagemaker-us-east-1-614093401978/models/autopilot"
    },
    "ResolvedAttributes": {
        "AutoMLJobObjective": {
            "MetricName": "F1"
        },
        "CompletionCriteria": {
```

```
            "MaxAutoMLJobRuntimeInSeconds": 5400,
            "MaxCandidates": 3,
            "MaxRuntimePerTrainingJobInSeconds": 900
        },
        "ProblemType": "BinaryClassification"
    },
    "ResponseMetadata": {
        "HTTPHeaders": {
            "content-length": "5897",
            "content-type": "application/x-amz-json-1.1",
            "date": "Mon, 17 Apr 2023 03:55:47 GMT",
            "x-amzn-requestid": "b1efaf91-fb54-4faa-9fa8-131aa73acd1e"
        },
        "HTTPStatusCode": 200,
        "RequestId": "b1efaf91-fb54-4faa-9fa8-131aa73acd1e",
        "RetryAttempts": 0
    },
    "RoleArn": "arn:aws:iam::614093401978:role/LabRole"
}
CPU times: user 725 µs, sys: 81 µs, total: 806 µs
Wall time: 787 µs
```


View Generated Notebook Samples

In [93]:

```python
job_description_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)

while "AutoMLJobArtifacts" not in job_description_response.keys():
    job_description_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
    print("[INFO] Autopilot Job has not yet generated the artifacts. Please wait. ")
    print(json.dumps(job_description_response, indent=4, sort_keys=True, default=str))
    print("[INFO] Waiting for AutoMLJobArtifacts...")
    sleep(15)

print("[OK] AutoMLJobArtifacts generated.")
```

```
[OK] AutoMLJobArtifacts generated.
```

In [94]:
```
job_description_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)

while "DataExplorationNotebookLocation" not in job_description_response["AutoMLJobArtifacts"].keys():
    job_description_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
    print("[INFO] Autopilot Job has not yet generated the notebooks. Please wait. ")
    print(json.dumps(job_description_response, indent=4, sort_keys=True, default=str))
    print("[INFO] Waiting for DataExplorationNotebookLocation...")
    sleep(15)

print("[OK] DataExplorationNotebookLocation found.")
```

```
[OK] DataExplorationNotebookLocation found.
```

In [95]:
```
generated_resources = job_description_response["AutoMLJobArtifacts"]["DataExplorationNotebookLocation"]
download_path = generated_resources.rsplit("/notebooks/SageMakerAutopilotDataExplorationNotebook.ipynb")[0]
job_id = download_path.rsplit("/", 1)[-1]
```

In [96]:
```
from IPython.core.display import display, HTML

if not job_id:
    print("No AutoMLJobArtifacts found.")
else:
    display(
        HTML(
            '<b>Review <a target="blank" href="https://s3.console.aws.amazon.com/s3/buckets/{}/{}/{}/sagemaker-automl
                bucket, prefix_model_output, auto_ml_job_name, job_id
            )
        )
    )
```

Review S3 Generated Resources

Download Generated Notebooks and code

In [97]:
```
print(download_path)
```

```
s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/sagemaker-automl-candidates/automl-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785
```

In [98]:
```
try:
    !aws s3 cp --recursive $download_path .
except:
    print('Could not download the generated resources. Make sure the path is correct.')
```

```
download: s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/sagemaker-automl-candidates/a
utoml-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/generated_module/MANIFEST.in to generated_module/MANI
FEST.in
download: s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/sagemaker-automl-candidates/a
utoml-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/generated_module/candidate_data_processors/dpp0.py to
generated_module/candidate_data_processors/dpp0.py
download: s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/sagemaker-automl-candidates/a
utoml-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/generated_module/candidate_data_processors/trainer.py
to generated_module/candidate_data_processors/trainer.py
download: s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/sagemaker-automl-candidates/a
utoml-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/generated_module/candidate_data_processors/dpp2.py to
generated_module/candidate_data_processors/dpp2.py
download: s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/sagemaker-automl-candidates/a
utoml-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/generated_module/README.md to generated_module/READM
E.md
download: s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/sagemaker-automl-candidates/a
utoml-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/generated_module/candidate_data_processors/sagemaker_
serve.py to generated_module/candidate_data_processors/sagemaker_serve.py
download: s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/sagemaker-automl-candidates/a
utoml-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/generated_module/setup.py to generated_module/setup.p
y
download: s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/sagemaker-automl-candidates/a
utoml-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/notebooks/SageMakerAutopilotCandidateDefinitionNotebo
ok.ipynb to notebooks/SageMakerAutopilotCandidateDefinitionNotebook.ipynb
download: s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/sagemaker-automl-candidates/a
utoml-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/generated_module/candidate_data_processors/dpp1.py to
generated_module/candidate_data_processors/dpp1.py
download: s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/sagemaker-automl-candidates/a
utoml-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/notebooks/sagemaker_automl/README.md to notebooks/sag
emaker_automl/README.md
download: s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/sagemaker-automl-candidates/a
utoml-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/notebooks/sagemaker_automl/common.py to notebooks/sag
emaker_automl/common.py
download: s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/sagemaker-automl-candidates/a
utoml-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/notebooks/sagemaker_automl/__init__.py to notebooks/s
agemaker_automl/__init__.py
download: s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/sagemaker-automl-candidates/a
utoml-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/notebooks/sagemaker_automl/local_candidate.py to note
books/sagemaker_automl/local_candidate.py
download: s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/sagemaker-automl-candidates/a
utoml-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/notebooks/sagemaker_automl/interactive_runner.py to n
otebooks/sagemaker_automl/interactive_runner.py
download: s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/sagemaker-automl-candidates/a
utoml-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/notebooks/sagemaker_automl/config.py to notebooks/sag
```

```
emaker_automl/config.py
download: s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/sagemaker-automl-candidates/a
utoml-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/notebooks/sagemaker_automl/steps.py to notebooks/sage
maker_automl/steps.py
download: s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/sagemaker-automl-candidates/a
utoml-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/notebooks/SageMakerAutopilotDataExplorationNotebook.i
pynb to notebooks/SageMakerAutopilotDataExplorationNotebook.ipynb
```

Review the generated Resources

In [99]:
```
!ls ./generated_module/candidate_data_processors
```

```
dpp0.py  dpp1.py  dpp2.py  sagemaker_serve.py  trainer.py
```

In [100…
```
!ls ./notebooks
```

```
SageMakerAutopilotCandidateDefinitionNotebook.ipynb  sagemaker_automl
SageMakerAutopilotDataExplorationNotebook.ipynb
```

In [101…
```
from IPython.core.display import display, HTML

display(
    HTML(
        '<b>Review <a target="blank" href="https://console.aws.amazon.com/sagemaker/home?region={}#/jobs/">Training J
            region
        )
    )
)
```

Review Training Jobs

In [102…
```
from IPython.core.display import display, HTML

display(
    HTML(
        '<b>Review <a target="blank" href="https://console.aws.amazon.com/sagemaker/home?region={}#/transform-jobs/">
            region
        )
    )
)
```

Review Batch Transform Jobs

The next cell will show InProgress for a few minutes

In [103…

```
%%time

job_description_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
job_status = job_description_response["AutoMLJobStatus"]
job_sec_status = job_description_response["AutoMLJobSecondaryStatus"]
print(job_status)
print(job_sec_status)
if job_status not in ("Stopped", "Failed"):
    while job_status in ("InProgress") and job_sec_status in ("FeatureEngineering"):
        job_description_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
        job_status = job_description_response["AutoMLJobStatus"]
        job_sec_status = job_description_response["AutoMLJobSecondaryStatus"]
        print(job_status, job_sec_status)
        sleep(15)
    print("[OK] Feature engineering phase completed.\n")

print(json.dumps(job_description_response, indent=4, sort_keys=True, default=str))
```

```
Completed
Completed
[OK] Feature engineering phase completed.

{
    "AutoMLJobArn": "arn:aws:sagemaker:us-east-1:614093401978:automl-job/automl-dm-16-20-10-05",
    "AutoMLJobArtifacts": {
        "CandidateDefinitionNotebookLocation": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-
20-10-05/sagemaker-automl-candidates/automl-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/notebooks/SageM
akerAutopilotCandidateDefinitionNotebook.ipynb",
        "DataExplorationNotebookLocation": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-1
0-05/sagemaker-automl-candidates/automl-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/notebooks/SageMaker
AutopilotDataExplorationNotebook.ipynb"
    },
    "AutoMLJobConfig": {
        "CompletionCriteria": {
            "MaxAutoMLJobRuntimeInSeconds": 5400,
            "MaxCandidates": 3,
            "MaxRuntimePerTrainingJobInSeconds": 900
        }
    },
    "AutoMLJobName": "automl-dm-16-20-10-05",
    "AutoMLJobSecondaryStatus": "Completed",
    "AutoMLJobStatus": "Completed",
    "BestCandidate": {
        "CandidateName": "automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4",
        "CandidateProperties": {
            "CandidateArtifactLocations": {
                "Explainability": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/docu
mentation/explainability/output",
                "ModelInsights": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/docum
entation/model_monitor/output"
            },
            "CandidateMetrics": [
                {
                    "MetricName": "F1",
                    "Set": "Validation",
                    "StandardMetricName": "F1",
                    "Value": 0.995169997215271
                },
                {
                    "MetricName": "LogLoss",
                    "Set": "Validation",
                    "StandardMetricName": "LogLoss",
```

```
                    "Value": 0.11816000193357468
                },
                {
                    "MetricName": "Recall",
                    "Set": "Validation",
                    "StandardMetricName": "Recall",
                    "Value": 1.0
                },
                {
                    "MetricName": "Precision",
                    "Set": "Validation",
                    "StandardMetricName": "Precision",
                    "Value": 0.9904199838638306
                },
                {
                    "MetricName": "AUC",
                    "Set": "Validation",
                    "StandardMetricName": "AUC",
                    "Value": 0.9990599751472473
                },
                {
                    "MetricName": "Accuracy",
                    "Set": "Validation",
                    "StandardMetricName": "Accuracy",
                    "Value": 0.9953600168228149
                },
                {
                    "MetricName": "BalancedAccuracy",
                    "Set": "Validation",
                    "StandardMetricName": "BalancedAccuracy",
                    "Value": 0.9955800175666809
                }
            ]
        },
        "CandidateStatus": "Completed",
        "CandidateSteps": [
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:processing-job/automl-dm-16-20-10-05-d
b-1-2887815310fd4416804d2079485d7d062e1a",
                "CandidateStepName": "automl-dm-16-20-10-05-db-1-2887815310fd4416804d2079485d7d062e1a",
                "CandidateStepType": "AWS::SageMaker::ProcessingJob"
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:training-job/automl-dm-16-20-10-05-dpp
```

```
1-1-8411a0fbc81748a9958acf62493120d4d7",
                "CandidateStepName": "automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7",
                "CandidateStepType": "AWS::SageMaker::TrainingJob"
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:transform-job/automl-dm-16-20-10-05-dp
p1-csv-1-18357715aec34f7eb4c3b75bad6e49",
                "CandidateStepName": "automl-dm-16-20-10-05-dpp1-csv-1-18357715aec34f7eb4c3b75bad6e49",
                "CandidateStepType": "AWS::SageMaker::TransformJob"
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:training-job/automl-dm-16-20-10-054If5
RqjsyAN-001-61c635e4",
                "CandidateStepName": "automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4",
                "CandidateStepType": "AWS::SageMaker::TrainingJob"
            }
        ],
        "CreationTime": "2023-04-16 20:33:00+00:00",
        "EndTime": "2023-04-16 20:38:38+00:00",
        "FinalAutoMLJobObjectiveMetric": {
            "MetricName": "validation:f1_binary",
            "StandardMetricName": "F1",
            "Value": 0.995169997215271
        },
        "InferenceContainers": [
            {
                "Environment": {
                    "AUTOML_TRANSFORM_MODE": "feature-transform",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "application/x-recordio-protobuf",
                    "SAGEMAKER_PROGRAM": "sagemaker_serve",
                    "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-p
rocessor-models/automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz"
            },
            {
                "Environment": {
                    "MAX_CONTENT_LENGTH": "20971520",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
                    "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,probabilities"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.3-1-cpu-py3",
```

```
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/tuning
/automl-dm--dpp1-xgb/automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4/output/model.tar.gz"
            },
            {
                "Environment": {
                    "AUTOML_TRANSFORM_MODE": "inverse-label-transform",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
                    "SAGEMAKER_INFERENCE_INPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,labels,probabilities",
                    "SAGEMAKER_PROGRAM": "sagemaker_serve",
                    "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-p
rocessor-models/automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz"
            }
        ],
        "LastModifiedTime": "2023-04-16 20:39:21.759000+00:00",
        "ObjectiveStatus": "Succeeded"
    },
    "CreationTime": "2023-04-16 20:14:58.074000+00:00",
    "EndTime": "2023-04-16 20:47:31.359000+00:00",
    "GenerateCandidateDefinitionsOnly": false,
    "InputDataConfig": [
        {
            "ChannelType": "training",
            "ContentType": "text/csv;header=present",
            "DataSource": {
                "S3DataSource": {
                    "S3DataType": "S3Prefix",
                    "S3Uri": "s3://sagemaker-us-east-1-614093401978/data/df_autopilot.csv"
                }
            },
            "TargetAttributeName": "AML_detected"
        }
    ],
    "LastModifiedTime": "2023-04-16 20:47:31.403000+00:00",
    "OutputDataConfig": {
        "S3OutputPath": "s3://sagemaker-us-east-1-614093401978/models/autopilot"
    },
    "ResolvedAttributes": {
        "AutoMLJobObjective": {
            "MetricName": "F1"
```

```
        },
        "CompletionCriteria": {
            "MaxAutoMLJobRuntimeInSeconds": 5400,
            "MaxCandidates": 3,
            "MaxRuntimePerTrainingJobInSeconds": 900
        },
        "ProblemType": "BinaryClassification"
    },
    "ResponseMetadata": {
        "HTTPHeaders": {
            "content-length": "5897",
            "content-type": "application/x-amz-json-1.1",
            "date": "Mon, 17 Apr 2023 03:55:49 GMT",
            "x-amzn-requestid": "ce5a80c9-5e76-4b3c-85ac-430065d52082"
        },
        "HTTPStatusCode": 200,
        "RequestId": "ce5a80c9-5e76-4b3c-85ac-430065d52082",
        "RetryAttempts": 0
    },
    "RoleArn": "arn:aws:iam::614093401978:role/LabRole"
}
CPU times: user 6.96 ms, sys: 0 ns, total: 6.96 ms
Wall time: 105 ms
```


Model Training and Tuning

In [104…

```
from IPython.core.display import display, HTML

display(
    HTML(
        '<b>Review <a target="blank" href="https://console.aws.amazon.com/sagemaker/home?region={}#/hyper-tuning-jobs
            region
        )
    )
)
```

Review Hyperparameter Tuning Jobs

In [105…

```
from IPython.core.display import display, HTML

display(
    HTML(
        '<b>Review <a target="blank" href="https://console.aws.amazon.com/sagemaker/home?region={}#/jobs/">Training J
            region
        )
    )
)
```

Review Training Jobs

The next cell will show InProgress for a few minutes

In [106…

```
%%time

job_description_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
job_status = job_description_response["AutoMLJobStatus"]
job_sec_status = job_description_response["AutoMLJobSecondaryStatus"]
print(job_status)
print(job_sec_status)
if job_status not in ("Stopped", "Failed"):
    while job_status in ("InProgress") and job_sec_status in ("ModelTuning"):
        job_description_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
        job_status = job_description_response["AutoMLJobStatus"]
        job_sec_status = job_description_response["AutoMLJobSecondaryStatus"]
        print(job_status, job_sec_status)
        sleep(15)
    print("[OK] Model tuning phase completed.\n")

print(json.dumps(job_description_response, indent=4, sort_keys=True, default=str))
```

```
Completed
Completed
[OK] Model tuning phase completed.

{
    "AutoMLJobArn": "arn:aws:sagemaker:us-east-1:614093401978:automl-job/automl-dm-16-20-10-05",
    "AutoMLJobArtifacts": {
        "CandidateDefinitionNotebookLocation": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-
20-10-05/sagemaker-automl-candidates/automl-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/notebooks/SageM
akerAutopilotCandidateDefinitionNotebook.ipynb",
        "DataExplorationNotebookLocation": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-1
0-05/sagemaker-automl-candidates/automl-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/notebooks/SageMaker
AutopilotDataExplorationNotebook.ipynb"
    },
    "AutoMLJobConfig": {
        "CompletionCriteria": {
            "MaxAutoMLJobRuntimeInSeconds": 5400,
            "MaxCandidates": 3,
            "MaxRuntimePerTrainingJobInSeconds": 900
        }
    },
    "AutoMLJobName": "automl-dm-16-20-10-05",
    "AutoMLJobSecondaryStatus": "Completed",
    "AutoMLJobStatus": "Completed",
    "BestCandidate": {
        "CandidateName": "automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4",
        "CandidateProperties": {
            "CandidateArtifactLocations": {
                "Explainability": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/docu
mentation/explainability/output",
                "ModelInsights": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/docum
entation/model_monitor/output"
            },
            "CandidateMetrics": [
                {
                    "MetricName": "F1",
                    "Set": "Validation",
                    "StandardMetricName": "F1",
                    "Value": 0.995169997215271
                },
                {
                    "MetricName": "LogLoss",
                    "Set": "Validation",
                    "StandardMetricName": "LogLoss",
```

```
                    "Value": 0.11816000193357468
                },
                {
                    "MetricName": "Recall",
                    "Set": "Validation",
                    "StandardMetricName": "Recall",
                    "Value": 1.0
                },
                {
                    "MetricName": "Precision",
                    "Set": "Validation",
                    "StandardMetricName": "Precision",
                    "Value": 0.9904199838638306
                },
                {
                    "MetricName": "AUC",
                    "Set": "Validation",
                    "StandardMetricName": "AUC",
                    "Value": 0.9990599751472473
                },
                {
                    "MetricName": "Accuracy",
                    "Set": "Validation",
                    "StandardMetricName": "Accuracy",
                    "Value": 0.9953600168228149
                },
                {
                    "MetricName": "BalancedAccuracy",
                    "Set": "Validation",
                    "StandardMetricName": "BalancedAccuracy",
                    "Value": 0.9955800175666809
                }
            ]
        },
        "CandidateStatus": "Completed",
        "CandidateSteps": [
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:processing-job/automl-dm-16-20-10-05-d
b-1-2887815310fd4416804d2079485d7d062e1a",
                "CandidateStepName": "automl-dm-16-20-10-05-db-1-2887815310fd4416804d2079485d7d062e1a",
                "CandidateStepType": "AWS::SageMaker::ProcessingJob"
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:training-job/automl-dm-16-20-10-05-dpp
```

```
1-1-8411a0fbc81748a9958acf62493120d4d7",
                "CandidateStepName": "automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7",
                "CandidateStepType": "AWS::SageMaker::TrainingJob"
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:transform-job/automl-dm-16-20-10-05-dp
p1-csv-1-18357715aec34f7eb4c3b75bad6e49",
                "CandidateStepName": "automl-dm-16-20-10-05-dpp1-csv-1-18357715aec34f7eb4c3b75bad6e49",
                "CandidateStepType": "AWS::SageMaker::TransformJob"
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:training-job/automl-dm-16-20-10-054If5
RqjsyAN-001-61c635e4",
                "CandidateStepName": "automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4",
                "CandidateStepType": "AWS::SageMaker::TrainingJob"
            }
        ],
        "CreationTime": "2023-04-16 20:33:00+00:00",
        "EndTime": "2023-04-16 20:38:38+00:00",
        "FinalAutoMLJobObjectiveMetric": {
            "MetricName": "validation:f1_binary",
            "StandardMetricName": "F1",
            "Value": 0.995169997215271
        },
        "InferenceContainers": [
            {
                "Environment": {
                    "AUTOML_TRANSFORM_MODE": "feature-transform",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "application/x-recordio-protobuf",
                    "SAGEMAKER_PROGRAM": "sagemaker_serve",
                    "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-p
rocessor-models/automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz"
            },
            {
                "Environment": {
                    "MAX_CONTENT_LENGTH": "20971520",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
                    "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,probabilities"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.3-1-cpu-py3",
```

```
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/tuning
/automl-dm--dpp1-xgb/automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4/output/model.tar.gz"
            },
            {
                "Environment": {
                    "AUTOML_TRANSFORM_MODE": "inverse-label-transform",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
                    "SAGEMAKER_INFERENCE_INPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,labels,probabilities",
                    "SAGEMAKER_PROGRAM": "sagemaker_serve",
                    "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-p
rocessor-models/automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz"
            }
        ],
        "LastModifiedTime": "2023-04-16 20:39:21.759000+00:00",
        "ObjectiveStatus": "Succeeded"
    },
    "CreationTime": "2023-04-16 20:14:58.074000+00:00",
    "EndTime": "2023-04-16 20:47:31.359000+00:00",
    "GenerateCandidateDefinitionsOnly": false,
    "InputDataConfig": [
        {
            "ChannelType": "training",
            "ContentType": "text/csv;header=present",
            "DataSource": {
                "S3DataSource": {
                    "S3DataType": "S3Prefix",
                    "S3Uri": "s3://sagemaker-us-east-1-614093401978/data/df_autopilot.csv"
                }
            },
            "TargetAttributeName": "AML_detected"
        }
    ],
    "LastModifiedTime": "2023-04-16 20:47:31.403000+00:00",
    "OutputDataConfig": {
        "S3OutputPath": "s3://sagemaker-us-east-1-614093401978/models/autopilot"
    },
    "ResolvedAttributes": {
        "AutoMLJobObjective": {
            "MetricName": "F1"
```

```
        },
        "CompletionCriteria": {
            "MaxAutoMLJobRuntimeInSeconds": 5400,
            "MaxCandidates": 3,
            "MaxRuntimePerTrainingJobInSeconds": 900
        },
        "ProblemType": "BinaryClassification"
    },
    "ResponseMetadata": {
        "HTTPHeaders": {
            "content-length": "5897",
            "content-type": "application/x-amz-json-1.1",
            "date": "Mon, 17 Apr 2023 03:55:49 GMT",
            "x-amzn-requestid": "83313c47-458e-44dc-8f62-7c4d38216386"
        },
        "HTTPStatusCode": 200,
        "RequestId": "83313c47-458e-44dc-8f62-7c4d38216386",
        "RetryAttempts": 0
    },
    "RoleArn": "arn:aws:iam::614093401978:role/LabRole"
}
CPU times: user 4.86 ms, sys: 96 µs, total: 4.96 ms
Wall time: 112 ms
```

In [107…

```
%%time

job_description_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
job_status = job_description_response["AutoMLJobStatus"]
print(job_status)
if job_status not in ("Stopped", "Failed"):
    while job_status not in ("Completed"):
        job_description_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
        job_status = job_description_response["AutoMLJobStatus"]
        print(job_status)
        sleep(10)
    print("[OK] Autopilot Job completed.\n")
else:
    print(job_status)
```

```
Completed
[OK] Autopilot Job completed.

CPU times: user 4.51 ms, sys: 516 µs, total: 5.03 ms
Wall time: 102 ms
```

Viewing all Candidates

In [108…

```
candidates_response = sm.list_candidates_for_auto_ml_job(
    AutoMLJobName=auto_ml_job_name, SortBy="FinalObjectiveMetricValue"
)
```

In [109…

```
while "Candidates" not in candidates_response.keys():
    candidates_response = sm.list_candidates_for_auto_ml_job(
        AutoMLJobName=auto_ml_job_name, SortBy="FinalObjectiveMetricValue"
    )
    print("[INFO] Autopilot Job is generating the Candidates. Please wait.")
    print(json.dumps(candidates_response, indent=4, sort_keys=True, default=str))
    sleep(10)

candidates = candidates_response["Candidates"]
print("[OK] Candidates generated.")
```

```
[OK] Candidates generated.
```

In [110…

```
print(candidates[0].keys())
```

```
dict_keys(['CandidateName', 'FinalAutoMLJobObjectiveMetric', 'ObjectiveStatus', 'CandidateSteps', 'CandidateStatus',
'InferenceContainers', 'CreationTime', 'EndTime', 'LastModifiedTime', 'CandidateProperties'])
```

In [111…

```
while "CandidateName" not in candidates[0]:
    candidates_response = sm.list_candidates_for_auto_ml_job(
        AutoMLJobName=auto_ml_job_name, SortBy="FinalObjectiveMetricValue"
    )
    candidates = candidates_response["Candidates"]
    print("[INFO] Autopilot Job is generating CandidateName. Please wait. ")
    print(json.dumps(candidates, indent=4, sort_keys=True, default=str))
    sleep(10)

print("[OK] CandidateName generated.")
```

```
[OK] CandidateName generated.
```

In [112…

```
while "FinalAutoMLJobObjectiveMetric" not in candidates[0]:
    candidates_response = sm.list_candidates_for_auto_ml_job(
        AutoMLJobName=auto_ml_job_name, SortBy="FinalObjectiveMetricValue"
    )
    candidates = candidates_response["Candidates"]
    print("[INFO] Autopilot Job is generating FinalAutoMLJobObjectiveMetric. Please wait. ")
    print(json.dumps(candidates, indent=4, sort_keys=True, default=str))
    sleep(10)

print("[OK] FinalAutoMLJobObjectiveMetric generated.")
```

```
[OK] FinalAutoMLJobObjectiveMetric generated.
```

In [113…

```
print(json.dumps(candidates, indent=4, sort_keys=True, default=str))
```

```
[
    {
        "CandidateName": "automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4",
        "CandidateProperties": {
            "CandidateArtifactLocations": {
                "Explainability": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/docu
mentation/explainability/output",
                "ModelInsights": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/docum
entation/model_monitor/output"
            },
            "CandidateMetrics": [
                {
                    "MetricName": "F1",
                    "Set": "Validation",
                    "StandardMetricName": "F1",
                    "Value": 0.995169997215271
                },
                {
                    "MetricName": "LogLoss",
                    "Set": "Validation",
                    "StandardMetricName": "LogLoss",
                    "Value": 0.11816000193357468
                },
                {
                    "MetricName": "Recall",
                    "Set": "Validation",
                    "StandardMetricName": "Recall",
                    "Value": 1.0
                },
                {
                    "MetricName": "Precision",
                    "Set": "Validation",
                    "StandardMetricName": "Precision",
                    "Value": 0.9904199838638306
                },
                {
                    "MetricName": "AUC",
                    "Set": "Validation",
                    "StandardMetricName": "AUC",
                    "Value": 0.9990599751472473
                },
                {
                    "MetricName": "Accuracy",
                    "Set": "Validation",
```

```
                    "StandardMetricName": "Accuracy",
                    "Value": 0.9953600168228149
                },
                {
                    "MetricName": "BalancedAccuracy",
                    "Set": "Validation",
                    "StandardMetricName": "BalancedAccuracy",
                    "Value": 0.9955800175666809
                }
            ]
        },
        "CandidateStatus": "Completed",
        "CandidateSteps": [
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:processing-job/automl-dm-16-20-10-05-d
b-1-2887815310fd4416804d2079485d7d062e1a",
                "CandidateStepName": "automl-dm-16-20-10-05-db-1-2887815310fd4416804d2079485d7d062e1a",
                "CandidateStepType": "AWS::SageMaker::ProcessingJob"
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:training-job/automl-dm-16-20-10-05-dpp
1-1-8411a0fbc81748a9958acf62493120d4d7",
                "CandidateStepName": "automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7",
                "CandidateStepType": "AWS::SageMaker::TrainingJob"
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:transform-job/automl-dm-16-20-10-05-dp
p1-csv-1-18357715aec34f7eb4c3b75bad6e49",
                "CandidateStepName": "automl-dm-16-20-10-05-dpp1-csv-1-18357715aec34f7eb4c3b75bad6e49",
                "CandidateStepType": "AWS::SageMaker::TransformJob"
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:training-job/automl-dm-16-20-10-054If5
RqjsyAN-001-61c635e4",
                "CandidateStepName": "automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4",
                "CandidateStepType": "AWS::SageMaker::TrainingJob"
            }
        ],
        "CreationTime": "2023-04-16 20:33:00+00:00",
        "EndTime": "2023-04-16 20:38:38+00:00",
        "FinalAutoMLJobObjectiveMetric": {
            "MetricName": "validation:f1_binary",
            "StandardMetricName": "F1",
            "Value": 0.995169997215271
```

```
        },
        "InferenceContainers": [
            {
                "Environment": {
                    "AUTOML_TRANSFORM_MODE": "feature-transform",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "application/x-recordio-protobuf",
                    "SAGEMAKER_PROGRAM": "sagemaker_serve",
                    "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-p
rocessor-models/automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz"
            },
            {
                "Environment": {
                    "MAX_CONTENT_LENGTH": "20971520",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
                    "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,probabilities"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.3-1-cpu-py3",
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/tuning
/automl-dm--dpp1-xgb/automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4/output/model.tar.gz"
            },
            {
                "Environment": {
                    "AUTOML_TRANSFORM_MODE": "inverse-label-transform",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
                    "SAGEMAKER_INFERENCE_INPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,labels,probabilities",
                    "SAGEMAKER_PROGRAM": "sagemaker_serve",
                    "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-p
rocessor-models/automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz"
            }
        ],
        "LastModifiedTime": "2023-04-16 20:39:21.759000+00:00",
        "ObjectiveStatus": "Succeeded"
    },
    {
        "CandidateName": "automl-dm-16-20-10-054If5RqjsyAN-003-910bc585",
```

```
"CandidateProperties": {
    "CandidateMetrics": [
        {
            "MetricName": "F1",
            "Set": "Validation",
            "StandardMetricName": "F1",
            "Value": 0.9795699715614319
        },
        {
            "MetricName": "LogLoss",
            "Set": "Validation",
            "StandardMetricName": "LogLoss",
            "Value": 0.11089000105857849
        },
        {
            "MetricName": "Recall",
            "Set": "Validation",
            "StandardMetricName": "Recall",
            "Value": 0.9756100177764893
        },
        {
            "MetricName": "Precision",
            "Set": "Validation",
            "StandardMetricName": "Precision",
            "Value": 0.9837700128555298
        },
        {
            "MetricName": "AUC",
            "Set": "Validation",
            "StandardMetricName": "AUC",
            "Value": 0.998420000076294
        },
        {
            "MetricName": "Accuracy",
            "Set": "Validation",
            "StandardMetricName": "Accuracy",
            "Value": 0.9806600213050842
        },
        {
            "MetricName": "BalancedAccuracy",
            "Set": "Validation",
            "StandardMetricName": "BalancedAccuracy",
            "Value": 0.9804199934005737
        }
```

```
            ]
        },
        "CandidateStatus": "Completed",
        "CandidateSteps": [
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:processing-job/automl-dm-16-20-10-05-d
b-1-2887815310fd4416804d2079485d7d062e1a",
                "CandidateStepName": "automl-dm-16-20-10-05-db-1-2887815310fd4416804d2079485d7d062e1a",
                "CandidateStepType": "AWS::SageMaker::ProcessingJob"
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:training-job/automl-dm-16-20-10-05-dpp
1-1-8411a0fbc81748a9958acf62493120d4d7",
                "CandidateStepName": "automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7",
                "CandidateStepType": "AWS::SageMaker::TrainingJob"
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:transform-job/automl-dm-16-20-10-05-dp
p1-csv-1-18357715aec34f7eb4c3b75bad6e49",
                "CandidateStepName": "automl-dm-16-20-10-05-dpp1-csv-1-18357715aec34f7eb4c3b75bad6e49",
                "CandidateStepType": "AWS::SageMaker::TransformJob"
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:training-job/automl-dm-16-20-10-054If5
RqjsyAN-003-910bc585",
                "CandidateStepName": "automl-dm-16-20-10-054If5RqjsyAN-003-910bc585",
                "CandidateStepType": "AWS::SageMaker::TrainingJob"
            }
        ],
        "CreationTime": "2023-04-16 20:33:21+00:00",
        "EndTime": "2023-04-16 20:36:31+00:00",
        "FinalAutoMLJobObjectiveMetric": {
            "MetricName": "validation:f1_binary",
            "StandardMetricName": "F1",
            "Value": 0.9795699715614319
        },
        "InferenceContainers": [
            {
                "Environment": {
                    "AUTOML_TRANSFORM_MODE": "feature-transform",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "application/x-recordio-protobuf",
                    "SAGEMAKER_PROGRAM": "sagemaker_serve",
                    "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
                },
```

```
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-p
rocessor-models/automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz"
            },
            {
                "Environment": {
                    "MAX_CONTENT_LENGTH": "20971520",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
                    "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,probabilities"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.3-1-cpu-py3",
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/tuning
/automl-dm--dpp1-xgb/automl-dm-16-20-10-054If5RqjsyAN-003-910bc585/output/model.tar.gz"
            },
            {
                "Environment": {
                    "AUTOML_TRANSFORM_MODE": "inverse-label-transform",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
                    "SAGEMAKER_INFERENCE_INPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,labels,probabilities",
                    "SAGEMAKER_PROGRAM": "sagemaker_serve",
                    "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-p
rocessor-models/automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz"
            }
        ],
        "LastModifiedTime": "2023-04-16 20:39:21.685000+00:00",
        "ObjectiveStatus": "Succeeded"
    },
    {
        "CandidateName": "automl-dm-16-20-10-054If5RqjsyAN-002-ebe18ef1",
        "CandidateProperties": {
            "CandidateMetrics": [
                {
                    "MetricName": "F1",
                    "Set": "Validation",
                    "StandardMetricName": "F1",
                    "Value": 0.9746400117874146
                },
                {
```

```
                    "MetricName": "LogLoss",
                    "Set": "Validation",
                    "StandardMetricName": "LogLoss",
                    "Value": 0.1131799966096878
                },
                {
                    "MetricName": "Recall",
                    "Set": "Validation",
                    "StandardMetricName": "Recall",
                    "Value": 0.9666699767112732
                },
                {
                    "MetricName": "Precision",
                    "Set": "Validation",
                    "StandardMetricName": "Precision",
                    "Value": 0.9835600256919861
                },
                {
                    "MetricName": "AUC",
                    "Set": "Validation",
                    "StandardMetricName": "AUC",
                    "Value": 0.9985899925231934
                },
                {
                    "MetricName": "Accuracy",
                    "Set": "Validation",
                    "StandardMetricName": "Accuracy",
                    "Value": 0.9764099717140198
                },
                {
                    "MetricName": "BalancedAccuracy",
                    "Set": "Validation",
                    "StandardMetricName": "BalancedAccuracy",
                    "Value": 0.9759500026702881
                }
            ]
        },
        "CandidateStatus": "Completed",
        "CandidateSteps": [
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:processing-job/automl-dm-16-20-10-05-d
b-1-2887815310fd4416804d2079485d7d062e1a",
                "CandidateStepName": "automl-dm-16-20-10-05-db-1-2887815310fd4416804d2079485d7d062e1a",
                "CandidateStepType": "AWS::SageMaker::ProcessingJob"
```

```
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:training-job/automl-dm-16-20-10-05-dpp
0-1-cbc8bbf88b664a35a963ac2c65fe94899c",
                "CandidateStepName": "automl-dm-16-20-10-05-dpp0-1-cbc8bbf88b664a35a963ac2c65fe94899c",
                "CandidateStepType": "AWS::SageMaker::TrainingJob"
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:transform-job/automl-dm-16-20-10-05-dp
p0-csv-1-0d6756d0637d421dabef7d3c55af5a",
                "CandidateStepName": "automl-dm-16-20-10-05-dpp0-csv-1-0d6756d0637d421dabef7d3c55af5a",
                "CandidateStepType": "AWS::SageMaker::TransformJob"
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:training-job/automl-dm-16-20-10-054If5
RqjsyAN-002-ebe18ef1",
                "CandidateStepName": "automl-dm-16-20-10-054If5RqjsyAN-002-ebe18ef1",
                "CandidateStepType": "AWS::SageMaker::TrainingJob"
            }
        ],
        "CreationTime": "2023-04-16 20:33:01+00:00",
        "EndTime": "2023-04-16 20:35:19+00:00",
        "FinalAutoMLJobObjectiveMetric": {
            "MetricName": "validation:f1_binary",
            "StandardMetricName": "F1",
            "Value": 0.9746400117874146
        },
        "InferenceContainers": [
            {
                "Environment": {
                    "AUTOML_TRANSFORM_MODE": "feature-transform",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "application/x-recordio-protobuf",
                    "SAGEMAKER_PROGRAM": "sagemaker_serve",
                    "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-p
rocessor-models/automl-dm-16-20-10-05-dpp0-1-cbc8bbf88b664a35a963ac2c65fe94899c/output/model.tar.gz"
            },
            {
                "Environment": {
                    "MAX_CONTENT_LENGTH": "20971520",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
                    "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
```

```
                    "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,probabilities"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.3-1-cpu-py3",
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/tuning
/automl-dm--dpp0-xgb/automl-dm-16-20-10-054If5RqjsyAN-002-ebe18ef1/output/model.tar.gz"
            },
            {
                "Environment": {
                    "AUTOML_TRANSFORM_MODE": "inverse-label-transform",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
                    "SAGEMAKER_INFERENCE_INPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,labels,probabilities",
                    "SAGEMAKER_PROGRAM": "sagemaker_serve",
                    "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-p
rocessor-models/automl-dm-16-20-10-05-dpp0-1-cbc8bbf88b664a35a963ac2c65fe94899c/output/model.tar.gz"
            }
        ],
        "LastModifiedTime": "2023-04-16 20:39:21.686000+00:00",
        "ObjectiveStatus": "Succeeded"
    }
]
```


In [114…

```
for index, candidate in enumerate(candidates):
    print(
        str(index)
        + "  "
        + candidate["CandidateName"]
        + "  "
        + str(candidate["FinalAutoMLJobObjectiveMetric"]["Value"])
    )
```

```
0  automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4  0.995169997215271
1  automl-dm-16-20-10-054If5RqjsyAN-003-910bc585  0.9795699715614319
2  automl-dm-16-20-10-054If5RqjsyAN-002-ebe18ef1  0.9746400117874146
```

Inspect Trials using Experiments API

In [115…

```
from sagemaker.analytics import ExperimentAnalytics, TrainingJobAnalytics

exp = ExperimentAnalytics(
    sagemaker_session=sess,
    experiment_name=auto_ml_job_name + "-aws-auto-ml-job",
)

df = exp.dataframe()
print(df)
```

```
                                TrialComponentName  \
0  automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4-...
1  automl-dm-16-20-10-054If5RqjsyAN-003-910bc585-...
2  automl-dm-16-20-10-054If5RqjsyAN-002-ebe18ef1-...
3  automl-dm-16-20-10-05-dpp1-csv-1-18357715aec34...
4  automl-dm-16-20-10-05-dpp0-csv-1-0d6756d0637d4...
5  automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a99...
6  automl-dm-16-20-10-05-dpp0-1-cbc8bbf88b664a35a...
7  automl-dm-16-20-10-05-db-1-2887815310fd4416804...

                                     DisplayName  \
0  automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4-...
1  automl-dm-16-20-10-054If5RqjsyAN-003-910bc585-...
2  automl-dm-16-20-10-054If5RqjsyAN-002-ebe18ef1-...
3  automl-dm-16-20-10-05-dpp1-csv-1-18357715aec34...
4  automl-dm-16-20-10-05-dpp0-csv-1-0d6756d0637d4...
5  automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a99...
6  automl-dm-16-20-10-05-dpp0-1-cbc8bbf88b664a35a...
7  automl-dm-16-20-10-05-db-1-2887815310fd4416804...

                                       SourceArn  \
0  arn:aws:sagemaker:us-east-1:614093401978:train...
1  arn:aws:sagemaker:us-east-1:614093401978:train...
2  arn:aws:sagemaker:us-east-1:614093401978:train...
3  arn:aws:sagemaker:us-east-1:614093401978:trans...
4  arn:aws:sagemaker:us-east-1:614093401978:trans...
5  arn:aws:sagemaker:us-east-1:614093401978:train...
6  arn:aws:sagemaker:us-east-1:614093401978:train...
7  arn:aws:sagemaker:us-east-1:614093401978:proce...

                                SageMaker.ImageUri  SageMaker.InstanceCount  \
0  683313688378.dkr.ecr.us-east-1.amazonaws.com/s...                      1.0
1  683313688378.dkr.ecr.us-east-1.amazonaws.com/s...                      1.0
2  683313688378.dkr.ecr.us-east-1.amazonaws.com/s...                      1.0
3                                                NaN                      1.0
4                                                NaN                      1.0
5  683313688378.dkr.ecr.us-east-1.amazonaws.com/s...                      1.0
6  683313688378.dkr.ecr.us-east-1.amazonaws.com/s...                      1.0
7                                                NaN                      1.0

  SageMaker.InstanceType  SageMaker.VolumeSizeInGB  _kfold  _num_cv_round  \
0        ml.m5.12xlarge                      50.0     5.0            3.0
1        ml.m5.12xlarge                      50.0     5.0            3.0
2        ml.m5.12xlarge                      50.0     5.0            3.0
```

```
3          ml.m5.4xlarge                       NaN     NaN            NaN
4          ml.m5.4xlarge                       NaN     NaN            NaN
5         ml.m5.12xlarge                      50.0     NaN            NaN
6         ml.m5.12xlarge                      50.0     NaN            NaN
7          ml.m5.2xlarge                     250.0     NaN            NaN

  _tuning_objective_metric  ...  enable_validation_split  input_channel_mode  \
0    validation:f1_binary  ...                      NaN                 NaN
1    validation:f1_binary  ...                      NaN                 NaN
2    validation:f1_binary  ...                      NaN                 NaN
3                     NaN  ...                      NaN                 NaN
4                     NaN  ...                      NaN                 NaN
5                     NaN  ...                      NaN                 NaN
6                     NaN  ...                      NaN                 NaN
7                     NaN  ...                     true                Pipe

               job_name     label_col  max_dataset_size  \
0                   NaN           NaN               NaN
1                   NaN           NaN               NaN
2                   NaN           NaN               NaN
3                   NaN           NaN               NaN
4                   NaN           NaN               NaN
5                   NaN           NaN               NaN
6                   NaN           NaN               NaN
7  automl-dm-16-20-10-05  AML_detected               100

   max_subsampled_dataset_size  SageMaker.ImageUri - MediaType  \
0                          NaN                             NaN
1                          NaN                             NaN
2                          NaN                             NaN
3                          NaN                             NaN
4                          NaN                             NaN
5                          NaN                             NaN
6                          NaN                             NaN
7                            5                             NaN

                        SageMaker.ImageUri - Value  ds - MediaType  \
0                                              NaN             NaN
1                                              NaN             NaN
2                                              NaN             NaN
3                                              NaN             NaN
4                                              NaN             NaN
5                                              NaN             NaN
6                                              NaN             NaN
```

```
7  120479346908.dkr.ecr.us-east-1.amazonaws.com/d...            NaN

                                          ds - Value
0                                                NaN
1                                                NaN
2                                                NaN
3                                                NaN
4                                                NaN
5                                                NaN
6                                                NaN
7  s3://sagemaker-us-east-1-614093401978/models/a...

[8 rows x 126 columns]
```

Explore the Best Candidate

In [116…
```
best_candidate_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
```

In [117…
```
print(best_candidate_response.keys())
```

```
dict_keys(['AutoMLJobName', 'AutoMLJobArn', 'InputDataConfig', 'OutputDataConfig', 'RoleArn', 'AutoMLJobConfig', 'Cr
eationTime', 'EndTime', 'LastModifiedTime', 'BestCandidate', 'AutoMLJobStatus', 'AutoMLJobSecondaryStatus', 'Generat
eCandidateDefinitionsOnly', 'AutoMLJobArtifacts', 'ResolvedAttributes', 'ResponseMetadata'])
```

In [118…
```
while "BestCandidate" not in best_candidate_response:
    best_candidate_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
    print("[INFO] Autopilot Job is generating BestCandidate. Please wait. ")
    print(json.dumps(best_candidate_response, indent=4, sort_keys=True, default=str))
    sleep(10)

best_candidate = best_candidate_response["BestCandidate"]
print("[OK] BestCandidate generated.")
```

```
[OK] BestCandidate generated.
```

In [119…
```
print(json.dumps(best_candidate_response, indent=4, sort_keys=True, default=str))
```

```
{
    "AutoMLJobArn": "arn:aws:sagemaker:us-east-1:614093401978:automl-job/automl-dm-16-20-10-05",
    "AutoMLJobArtifacts": {
        "CandidateDefinitionNotebookLocation": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-
20-10-05/sagemaker-automl-candidates/automl-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/notebooks/SageM
akerAutopilotCandidateDefinitionNotebook.ipynb",
        "DataExplorationNotebookLocation": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-1
0-05/sagemaker-automl-candidates/automl-dm-16-20-10-05-pr-1-4b26f436ffcc4edaace14844f354501a5785/notebooks/SageMaker
AutopilotDataExplorationNotebook.ipynb"
    },
    "AutoMLJobConfig": {
        "CompletionCriteria": {
            "MaxAutoMLJobRuntimeInSeconds": 5400,
            "MaxCandidates": 3,
            "MaxRuntimePerTrainingJobInSeconds": 900
        }
    },
    "AutoMLJobName": "automl-dm-16-20-10-05",
    "AutoMLJobSecondaryStatus": "Completed",
    "AutoMLJobStatus": "Completed",
    "BestCandidate": {
        "CandidateName": "automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4",
        "CandidateProperties": {
            "CandidateArtifactLocations": {
                "Explainability": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/docu
mentation/explainability/output",
                "ModelInsights": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/docum
entation/model_monitor/output"
            },
            "CandidateMetrics": [
                {
                    "MetricName": "F1",
                    "Set": "Validation",
                    "StandardMetricName": "F1",
                    "Value": 0.995169997215271
                },
                {
                    "MetricName": "LogLoss",
                    "Set": "Validation",
                    "StandardMetricName": "LogLoss",
                    "Value": 0.11816000193357468
                },
                {
                    "MetricName": "Recall",
```

```
                    "Set": "Validation",
                    "StandardMetricName": "Recall",
                    "Value": 1.0
                },
                {
                    "MetricName": "Precision",
                    "Set": "Validation",
                    "StandardMetricName": "Precision",
                    "Value": 0.9904199838638306
                },
                {
                    "MetricName": "AUC",
                    "Set": "Validation",
                    "StandardMetricName": "AUC",
                    "Value": 0.9990599751472473
                },
                {
                    "MetricName": "Accuracy",
                    "Set": "Validation",
                    "StandardMetricName": "Accuracy",
                    "Value": 0.9953600168228149
                },
                {
                    "MetricName": "BalancedAccuracy",
                    "Set": "Validation",
                    "StandardMetricName": "BalancedAccuracy",
                    "Value": 0.9955800175666809
                }
            ]
        },
        "CandidateStatus": "Completed",
        "CandidateSteps": [
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:processing-job/automl-dm-16-20-10-05-d
b-1-2887815310fd4416804d2079485d7d062e1a",
                "CandidateStepName": "automl-dm-16-20-10-05-db-1-2887815310fd4416804d2079485d7d062e1a",
                "CandidateStepType": "AWS::SageMaker::ProcessingJob"
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:training-job/automl-dm-16-20-10-05-dpp
1-1-8411a0fbc81748a9958acf62493120d4d7",
                "CandidateStepName": "automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7",
                "CandidateStepType": "AWS::SageMaker::TrainingJob"
            },
```

```
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:transform-job/automl-dm-16-20-10-05-dp
p1-csv-1-18357715aec34f7eb4c3b75bad6e49",
                "CandidateStepName": "automl-dm-16-20-10-05-dpp1-csv-1-18357715aec34f7eb4c3b75bad6e49",
                "CandidateStepType": "AWS::SageMaker::TransformJob"
            },
            {
                "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:training-job/automl-dm-16-20-10-054If5
RqjsyAN-001-61c635e4",
                "CandidateStepName": "automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4",
                "CandidateStepType": "AWS::SageMaker::TrainingJob"
            }
        ],
        "CreationTime": "2023-04-16 20:33:00+00:00",
        "EndTime": "2023-04-16 20:38:38+00:00",
        "FinalAutoMLJobObjectiveMetric": {
            "MetricName": "validation:f1_binary",
            "StandardMetricName": "F1",
            "Value": 0.995169997215271
        },
        "InferenceContainers": [
            {
                "Environment": {
                    "AUTOML_TRANSFORM_MODE": "feature-transform",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "application/x-recordio-protobuf",
                    "SAGEMAKER_PROGRAM": "sagemaker_serve",
                    "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-p
rocessor-models/automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz"
            },
            {
                "Environment": {
                    "MAX_CONTENT_LENGTH": "20971520",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
                    "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,probabilities"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.3-1-cpu-py3",
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/tuning
/automl-dm--dpp1-xgb/automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4/output/model.tar.gz"
            },
            {
```

```
                "Environment": {
                    "AUTOML_TRANSFORM_MODE": "inverse-label-transform",
                    "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
                    "SAGEMAKER_INFERENCE_INPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
                    "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,labels,probabilities",
                    "SAGEMAKER_PROGRAM": "sagemaker_serve",
                    "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
                },
                "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
                "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-p
rocessor-models/automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz"
            }
        ],
        "LastModifiedTime": "2023-04-16 20:39:21.759000+00:00",
        "ObjectiveStatus": "Succeeded"
    },
    "CreationTime": "2023-04-16 20:14:58.074000+00:00",
    "EndTime": "2023-04-16 20:47:31.359000+00:00",
    "GenerateCandidateDefinitionsOnly": false,
    "InputDataConfig": [
        {
            "ChannelType": "training",
            "ContentType": "text/csv;header=present",
            "DataSource": {
                "S3DataSource": {
                    "S3DataType": "S3Prefix",
                    "S3Uri": "s3://sagemaker-us-east-1-614093401978/data/df_autopilot.csv"
                }
            },
            "TargetAttributeName": "AML_detected"
        }
    ],
    "LastModifiedTime": "2023-04-16 20:47:31.403000+00:00",
    "OutputDataConfig": {
        "S3OutputPath": "s3://sagemaker-us-east-1-614093401978/models/autopilot"
    },
    "ResolvedAttributes": {
        "AutoMLJobObjective": {
            "MetricName": "F1"
        },
        "CompletionCriteria": {
            "MaxAutoMLJobRuntimeInSeconds": 5400,
            "MaxCandidates": 3,
```

```
            "MaxRuntimePerTrainingJobInSeconds": 900
        },
        "ProblemType": "BinaryClassification"
    },
    "ResponseMetadata": {
        "HTTPHeaders": {
            "content-length": "5897",
            "content-type": "application/x-amz-json-1.1",
            "date": "Mon, 17 Apr 2023 03:55:50 GMT",
            "x-amzn-requestid": "d08f68fe-7de6-4222-9709-b84f6db538d3"
        },
        "HTTPStatusCode": 200,
        "RequestId": "d08f68fe-7de6-4222-9709-b84f6db538d3",
        "RetryAttempts": 0
    },
    "RoleArn": "arn:aws:iam::614093401978:role/LabRole"
}
```


In [120…

```
print(best_candidate.keys())
```

```
dict_keys(['CandidateName', 'FinalAutoMLJobObjectiveMetric', 'ObjectiveStatus', 'CandidateSteps', 'CandidateStatus',
'InferenceContainers', 'CreationTime', 'EndTime', 'LastModifiedTime', 'CandidateProperties'])
```

In [121…

```
while "CandidateName" not in best_candidate:
    best_candidate_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
    best_candidate = best_candidate_response["BestCandidate"]
    print("[INFO] Autopilot Job is generating BestCandidate CandidateName. Please wait. ")
    print(json.dumps(best_candidate, indent=4, sort_keys=True, default=str))
    sleep(10)

print("[OK] BestCandidate CandidateName generated.")
```

```
[OK] BestCandidate CandidateName generated.
```

In [122…

```
while "FinalAutoMLJobObjectiveMetric" not in best_candidate:
    best_candidate_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
    best_candidate = best_candidate_response["BestCandidate"]
    print("[INFO] Autopilot Job is generating BestCandidate FinalAutoMLJobObjectiveMetric. Please wait. ")
    print(json.dumps(best_candidate, indent=4, sort_keys=True, default=str))
    sleep(10)

print("[OK] BestCandidate FinalAutoMLJobObjectiveMetric generated.")
```

```
[OK] BestCandidate FinalAutoMLJobObjectiveMetric generated.
```

In [123…

```
best_candidate_identifier = best_candidate["CandidateName"]
print("Candidate name: " + best_candidate_identifier)
print("Metric name: " + best_candidate["FinalAutoMLJobObjectiveMetric"]["MetricName"])
print("Metric value: " + str(best_candidate["FinalAutoMLJobObjectiveMetric"]["Value"]))
```

```
Candidate name: automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4
Metric name: validation:f1_binary
Metric value: 0.995169997215271
```

In [124…

```
print(json.dumps(best_candidate, indent=4, sort_keys=True, default=str))
```

```
{
    "CandidateName": "automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4",
    "CandidateProperties": {
        "CandidateArtifactLocations": {
            "Explainability": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/document
ation/explainability/output",
            "ModelInsights": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/documenta
tion/model_monitor/output"
        },
        "CandidateMetrics": [
            {
                "MetricName": "F1",
                "Set": "Validation",
                "StandardMetricName": "F1",
                "Value": 0.995169997215271
            },
            {
                "MetricName": "LogLoss",
                "Set": "Validation",
                "StandardMetricName": "LogLoss",
                "Value": 0.11816000193357468
            },
            {
                "MetricName": "Recall",
                "Set": "Validation",
                "StandardMetricName": "Recall",
                "Value": 1.0
            },
            {
                "MetricName": "Precision",
                "Set": "Validation",
                "StandardMetricName": "Precision",
                "Value": 0.9904199838638306
            },
            {
                "MetricName": "AUC",
                "Set": "Validation",
                "StandardMetricName": "AUC",
                "Value": 0.9990599751472473
            },
            {
                "MetricName": "Accuracy",
                "Set": "Validation",
                "StandardMetricName": "Accuracy",
```

```
                "Value": 0.9953600168228149
            },
            {
                "MetricName": "BalancedAccuracy",
                "Set": "Validation",
                "StandardMetricName": "BalancedAccuracy",
                "Value": 0.9955800175666809
            }
        ]
    },
    "CandidateStatus": "Completed",
    "CandidateSteps": [
        {
            "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:processing-job/automl-dm-16-20-10-05-db-1-
2887815310fd4416804d2079485d7d062e1a",
            "CandidateStepName": "automl-dm-16-20-10-05-db-1-2887815310fd4416804d2079485d7d062e1a",
            "CandidateStepType": "AWS::SageMaker::ProcessingJob"
        },
        {
            "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:training-job/automl-dm-16-20-10-05-dpp1-1-
8411a0fbc81748a9958acf62493120d4d7",
            "CandidateStepName": "automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7",
            "CandidateStepType": "AWS::SageMaker::TrainingJob"
        },
        {
            "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:transform-job/automl-dm-16-20-10-05-dpp1-c
sv-1-18357715aec34f7eb4c3b75bad6e49",
            "CandidateStepName": "automl-dm-16-20-10-05-dpp1-csv-1-18357715aec34f7eb4c3b75bad6e49",
            "CandidateStepType": "AWS::SageMaker::TransformJob"
        },
        {
            "CandidateStepArn": "arn:aws:sagemaker:us-east-1:614093401978:training-job/automl-dm-16-20-10-054If5Rqjs
yAN-001-61c635e4",
            "CandidateStepName": "automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4",
            "CandidateStepType": "AWS::SageMaker::TrainingJob"
        }
    ],
    "CreationTime": "2023-04-16 20:33:00+00:00",
    "EndTime": "2023-04-16 20:38:38+00:00",
    "FinalAutoMLJobObjectiveMetric": {
        "MetricName": "validation:f1_binary",
        "StandardMetricName": "F1",
        "Value": 0.995169997215271
    },
```

```
    "InferenceContainers": [
        {
            "Environment": {
                "AUTOML_TRANSFORM_MODE": "feature-transform",
                "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "application/x-recordio-protobuf",
                "SAGEMAKER_PROGRAM": "sagemaker_serve",
                "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
            },
            "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
            "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-proce
ssor-models/automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz"
        },
        {
            "Environment": {
                "MAX_CONTENT_LENGTH": "20971520",
                "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
                "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
                "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,probabilities"
            },
            "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.3-1-cpu-py3",
            "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/tuning/aut
oml-dm--dpp1-xgb/automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4/output/model.tar.gz"
        },
        {
            "Environment": {
                "AUTOML_TRANSFORM_MODE": "inverse-label-transform",
                "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
                "SAGEMAKER_INFERENCE_INPUT": "predicted_label",
                "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
                "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,labels,probabilities",
                "SAGEMAKER_PROGRAM": "sagemaker_serve",
                "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
            },
            "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
            "ModelDataUrl": "s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-proce
ssor-models/automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz"
        }
    ],
    "LastModifiedTime": "2023-04-16 20:39:21.759000+00:00",
    "ObjectiveStatus": "Succeeded"
}
```

View individual Autopilot jobs

In [125…

```
while "CandidateSteps" not in best_candidate:
    best_candidate_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
    best_candidate = best_candidate_response["BestCandidate"]
    print("[INFO] Autopilot Job is generating BestCandidate CandidateSteps. Please wait. ")
    print(json.dumps(best_candidate, indent=4, sort_keys=True, default=str))
    sleep(10)

best_candidate = best_candidate_response["BestCandidate"]
print("[OK] BestCandidate CandidateSteps generated.")
```

```
[OK] BestCandidate CandidateSteps generated.
```

In [126…

```
while "CandidateStepType" not in best_candidate["CandidateSteps"][0]:
    best_candidate_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
    best_candidate = best_candidate_response["BestCandidate"]
    print("[INFO] Autopilot Job is generating BestCandidate CandidateSteps CandidateStepType. Please wait. ")
    print(json.dumps(best_candidate, indent=4, sort_keys=True, default=str))
    sleep(10)

best_candidate = best_candidate_response["BestCandidate"]
print("[OK] BestCandidate CandidateSteps CandidateStepType generated.")
```

```
[OK] BestCandidate CandidateSteps CandidateStepType generated.
```

In [127…

```
while "CandidateStepName" not in best_candidate["CandidateSteps"][0]:
    best_candidate_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
    best_candidate = best_candidate_response["BestCandidate"]
    print("[INFO] Autopilot Job is generating BestCandidate CandidateSteps CandidateStepName. Please wait. ")
    print(json.dumps(best_candidate, indent=4, sort_keys=True, default=str))
    sleep(10)

best_candidate = best_candidate_response["BestCandidate"]
print("[OK] BestCandidate CandidateSteps CandidateStepName generated.")
```

```
[OK] BestCandidate CandidateSteps CandidateStepName generated.
```

In [128…

```
best_candidate
```

Out[128]:

```
{'CandidateName': 'automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4',
 'FinalAutoMLJobObjectiveMetric': {'MetricName': 'validation:f1_binary',
  'Value': 0.995169997215271,
  'StandardMetricName': 'F1'},
 'ObjectiveStatus': 'Succeeded',
 'CandidateSteps': [{'CandidateStepType': 'AWS::SageMaker::ProcessingJob',
   'CandidateStepArn': 'arn:aws:sagemaker:us-east-1:614093401978:processing-job/automl-dm-16-20-10-05-db-1-2887815310fd4416804d2079485d7d062e1a',
   'CandidateStepName': 'automl-dm-16-20-10-05-db-1-2887815310fd4416804d2079485d7d062e1a'},
  {'CandidateStepType': 'AWS::SageMaker::TrainingJob',
   'CandidateStepArn': 'arn:aws:sagemaker:us-east-1:614093401978:training-job/automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7',
   'CandidateStepName': 'automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7'},
  {'CandidateStepType': 'AWS::SageMaker::TransformJob',
   'CandidateStepArn': 'arn:aws:sagemaker:us-east-1:614093401978:transform-job/automl-dm-16-20-10-05-dpp1-csv-1-18357715aec34f7eb4c3b75bad6e49',
   'CandidateStepName': 'automl-dm-16-20-10-05-dpp1-csv-1-18357715aec34f7eb4c3b75bad6e49'},
  {'CandidateStepType': 'AWS::SageMaker::TrainingJob',
   'CandidateStepArn': 'arn:aws:sagemaker:us-east-1:614093401978:training-job/automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4',
   'CandidateStepName': 'automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4'}],
 'CandidateStatus': 'Completed',
 'InferenceContainers': [{'Image': '683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3',
   'ModelDataUrl': 's3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-processor-models/automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz',
   'Environment': {'AUTOML_TRANSFORM_MODE': 'feature-transform',
    'SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT': 'application/x-recordio-protobuf',
    'SAGEMAKER_PROGRAM': 'sagemaker_serve',
    'SAGEMAKER_SUBMIT_DIRECTORY': '/opt/ml/model/code'}},
  {'Image': '683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.3-1-cpu-py3',
   'ModelDataUrl': 's3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/tuning/automl-dm--dpp1-xgb/automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4/output/model.tar.gz',
   'Environment': {'MAX_CONTENT_LENGTH': '20971520',
    'SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT': 'text/csv',
    'SAGEMAKER_INFERENCE_OUTPUT': 'predicted_label',
    'SAGEMAKER_INFERENCE_SUPPORTED': 'predicted_label,probability,probabilities'}},
  {'Image': '683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3',
   'ModelDataUrl': 's3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-processor-models/automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz',
   'Environment': {'AUTOML_TRANSFORM_MODE': 'inverse-label-transform',
    'SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT': 'text/csv',
    'SAGEMAKER_INFERENCE_INPUT': 'predicted_label',
    'SAGEMAKER_INFERENCE_OUTPUT': 'predicted_label',
```

```
    'SAGEMAKER_INFERENCE_SUPPORTED': 'predicted_label,probability,labels,probabilities',
    'SAGEMAKER_PROGRAM': 'sagemaker_serve',
    'SAGEMAKER_SUBMIT_DIRECTORY': '/opt/ml/model/code'}}],
 'CreationTime': datetime.datetime(2023, 4, 16, 20, 33, tzinfo=tzlocal()),
 'EndTime': datetime.datetime(2023, 4, 16, 20, 38, 38, tzinfo=tzlocal()),
 'LastModifiedTime': datetime.datetime(2023, 4, 16, 20, 39, 21, 759000, tzinfo=tzlocal()),
 'CandidateProperties': {'CandidateArtifactLocations': {'Explainability': 's3://sagemaker-us-east-1-614093401978/mod
els/autopilot/automl-dm-16-20-10-05/documentation/explainability/output',
   'ModelInsights': 's3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/documentation/mode
l_monitor/output'},
  'CandidateMetrics': [{'MetricName': 'F1',
    'Value': 0.995169997215271,
    'Set': 'Validation',
    'StandardMetricName': 'F1'},
   {'MetricName': 'LogLoss',
    'Value': 0.11816000193357468,
    'Set': 'Validation',
    'StandardMetricName': 'LogLoss'},
   {'MetricName': 'Recall',
    'Value': 1.0,
    'Set': 'Validation',
    'StandardMetricName': 'Recall'},
   {'MetricName': 'Precision',
    'Value': 0.9904199838638306,
    'Set': 'Validation',
    'StandardMetricName': 'Precision'},
   {'MetricName': 'AUC',
    'Value': 0.9990599751472473,
    'Set': 'Validation',
    'StandardMetricName': 'AUC'},
   {'MetricName': 'Accuracy',
    'Value': 0.9953600168228149,
    'Set': 'Validation',
    'StandardMetricName': 'Accuracy'},
   {'MetricName': 'BalancedAccuracy',
    'Value': 0.9955800175666809,
    'Set': 'Validation',
    'StandardMetricName': 'BalancedAccuracy'}]}}
```

In [129…

```
steps = []
for step in best_candidate["CandidateSteps"]:
    print("Candidate Step Type: {}".format(step["CandidateStepType"]))
    print("Candidate Step Name: {}".format(step["CandidateStepName"]))
    steps.append(step["CandidateStepName"])
```

```
Candidate Step Type: AWS::SageMaker::ProcessingJob
Candidate Step Name: automl-dm-16-20-10-05-db-1-2887815310fd4416804d2079485d7d062e1a
Candidate Step Type: AWS::SageMaker::TrainingJob
Candidate Step Name: automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7
Candidate Step Type: AWS::SageMaker::TransformJob
Candidate Step Name: automl-dm-16-20-10-05-dpp1-csv-1-18357715aec34f7eb4c3b75bad6e49
Candidate Step Type: AWS::SageMaker::TrainingJob
Candidate Step Name: automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4
```

In [130…

```
from IPython.core.display import display, HTML

display(
    HTML(
        '<b>Review Best Candidate <a target="blank" href="https://console.aws.amazon.com/sagemaker/home?region={}#/pr
            region, steps[0]
        )
    )
)
```

Review Best Candidate Processing Job

In [131…

```
from IPython.core.display import display, HTML

display(
    HTML(
        '<b>Review Best Candidate <a target="blank" href="https://console.aws.amazon.com/sagemaker/home?region={}#/jo
            region, steps[1]
        )
    )
)
```

Review Best Candidate Training Job

In [132…

```
from IPython.core.display import display, HTML

display(
    HTML(
        '<b>Review Best Candidate <a target="blank" href="https://console.aws.amazon.com/sagemaker/home?region={}#/tr
            region, steps[2]
        )
    )
)
```

Review Best Candidate Transform Job

In [133…

```
from IPython.core.display import display, HTML

display(
    HTML(
        '<b>Review Best Candidate <a target="blank" href="https://console.aws.amazon.com/sagemaker/home?region={}#/jo
            region, steps[3]
        )
    )
)
```

Review Best Candidate Training Job (Tuning)

In [134…

```
from IPython.core.display import display, HTML

display(
    HTML(
        '<b>Review All <a target="blank" href="https://console.aws.amazon.com/sagemaker/home?region={}#/processing-jo
            region
        )
    )
)
```

Review All Processing Jobs

Review all output in S3

In [135…

```
from IPython.core.display import display, HTML

display(
    HTML(
        '<b>Review All <a target="blank" href="https://s3.console.aws.amazon.com/s3/buckets/{}?region={}&prefix=model
            bucket, region, auto_ml_job_name
        )
    )
)
```

Review All Output in S3

See the containers and models within the inference pipeline

In [136…

```
while "InferenceContainers" not in best_candidate:
    best_candidate_response = sm.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
    best_candidate = best_candidate_response["BestCandidate"]
    print("[INFO] Autopilot Job is generating BestCandidate InferenceContainers. Please wait. ")
    print(json.dumps(best_candidate, indent=4, sort_keys=True, default=str))
    sleep(10)

print("[OK] BestCandidate InferenceContainers generated.")
```

```
[OK] BestCandidate InferenceContainers generated.
```

In [137…

```
best_candidate_containers = best_candidate["InferenceContainers"]
```

In [138…

```
for container in best_candidate_containers:
    print(container["Image"])
    print(container["ModelDataUrl"])
    print("======================")
```

```
683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3
s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-processor-models/automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz
======================
683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.3-1-cpu-py3
s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/tuning/automl-dm--dpp1-xgb/automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4/output/model.tar.gz
======================
683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3
s3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-processor-models/automl-dm-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz
======================
```

Update Containers to show predicted label and confidence score

In [139…

```python
for container in best_candidate_containers:
    print(container["Environment"])
    print("======================")
```

```
{'AUTOML_TRANSFORM_MODE': 'feature-transform', 'SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT': 'application/x-recordio-protobuf', 'SAGEMAKER_PROGRAM': 'sagemaker_serve', 'SAGEMAKER_SUBMIT_DIRECTORY': '/opt/ml/model/code'}
======================
{'MAX_CONTENT_LENGTH': '20971520', 'SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT': 'text/csv', 'SAGEMAKER_INFERENCE_OUTPUT': 'predicted_label', 'SAGEMAKER_INFERENCE_SUPPORTED': 'predicted_label,probability,probabilities'}
======================
{'AUTOML_TRANSFORM_MODE': 'inverse-label-transform', 'SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT': 'text/csv', 'SAGEMAKER_INFERENCE_INPUT': 'predicted_label', 'SAGEMAKER_INFERENCE_OUTPUT': 'predicted_label', 'SAGEMAKER_INFERENCE_SUPPORTED': 'predicted_label,probability,labels,probabilities', 'SAGEMAKER_PROGRAM': 'sagemaker_serve', 'SAGEMAKER_SUBMIT_DIRECTORY': '/opt/ml/model/code'}
======================
```

In [140…

```python
best_candidate_containers[1]["Environment"].update({"SAGEMAKER_INFERENCE_OUTPUT": "predicted_label, probability"})
best_candidate_containers[2]["Environment"].update({"SAGEMAKER_INFERENCE_INPUT": "predicted_label, probability"})
best_candidate_containers[2]["Environment"].update({"SAGEMAKER_INFERENCE_OUTPUT": "predicted_label, probability"})
```

In [141…

```python
for container in best_candidate_containers:
    print(container["Environment"])
    print("======================")
```

```
{'AUTOML_TRANSFORM_MODE': 'feature-transform', 'SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT': 'application/x-recordio-proto
buf', 'SAGEMAKER_PROGRAM': 'sagemaker_serve', 'SAGEMAKER_SUBMIT_DIRECTORY': '/opt/ml/model/code'}
======================
{'MAX_CONTENT_LENGTH': '20971520', 'SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT': 'text/csv', 'SAGEMAKER_INFERENCE_OUTPUT':
'predicted_label, probability', 'SAGEMAKER_INFERENCE_SUPPORTED': 'predicted_label,probability,probabilities'}
======================
{'AUTOML_TRANSFORM_MODE': 'inverse-label-transform', 'SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT': 'text/csv', 'SAGEMAKER_
INFERENCE_INPUT': 'predicted_label, probability', 'SAGEMAKER_INFERENCE_OUTPUT': 'predicted_label, probability', 'SAG
EMAKER_INFERENCE_SUPPORTED': 'predicted_label,probability,labels,probabilities', 'SAGEMAKER_PROGRAM': 'sagemaker_ser
ve', 'SAGEMAKER_SUBMIT_DIRECTORY': '/opt/ml/model/code'}
======================
```

Autopilot chose XGBoost as best candidate

In [142…

```
print(best_candidate["InferenceContainers"])
```

```
[{'Image': '683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3', 'ModelDataUrl': 's
3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-processor-models/automl-dm-16-20-10
-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz', 'Environment': {'AUTOML_TRANSFORM_MODE': 'featur
e-transform', 'SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT': 'application/x-recordio-protobuf', 'SAGEMAKER_PROGRAM': 'sagem
aker_serve', 'SAGEMAKER_SUBMIT_DIRECTORY': '/opt/ml/model/code'}}, {'Image': '683313688378.dkr.ecr.us-east-1.amazona
ws.com/sagemaker-xgboost:1.3-1-cpu-py3', 'ModelDataUrl': 's3://sagemaker-us-east-1-614093401978/models/autopilot/aut
oml-dm-16-20-10-05/tuning/automl-dm--dpp1-xgb/automl-dm-16-20-10-054If5RqjsyAN-001-61c635e4/output/model.tar.gz', 'E
nvironment': {'MAX_CONTENT_LENGTH': '20971520', 'SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT': 'text/csv', 'SAGEMAKER_INFER
ENCE_OUTPUT': 'predicted_label, probability', 'SAGEMAKER_INFERENCE_SUPPORTED': 'predicted_label,probability,probabil
ities'}}, {'Image': '683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3', 'ModelDat
aUrl': 's3://sagemaker-us-east-1-614093401978/models/autopilot/automl-dm-16-20-10-05/data-processor-models/automl-dm
-16-20-10-05-dpp1-1-8411a0fbc81748a9958acf62493120d4d7/output/model.tar.gz', 'Environment': {'AUTOML_TRANSFORM_MODE
': 'inverse-label-transform', 'SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT': 'text/csv', 'SAGEMAKER_INFERENCE_INPUT': 'pred
icted_label, probability', 'SAGEMAKER_INFERENCE_OUTPUT': 'predicted_label, probability', 'SAGEMAKER_INFERENCE_SUPPOR
TED': 'predicted_label,probability,labels,probabilities', 'SAGEMAKER_PROGRAM': 'sagemaker_serve', 'SAGEMAKER_SUBMIT_
DIRECTORY': '/opt/ml/model/code'}}]
```

Deploy Model as a REST Endpoint

In [143…

```
%store -r autopilot_model_name
```

In [144…
```
try:
    autopilot_model_name
except NameError:
    timestamp_suffix = strftime("%d-%H-%M-%S", gmtime())
    autopilot_model_name = "automl-dm-model-" + timestamp_suffix
    print("[OK] Created Autopilot Model Name: " + autopilot_model_name)
```

In [145…
```
%store autopilot_model_name
```

```
Stored 'autopilot_model_name' (str)
```

In [146…
```
%store -r autopilot_model_arn
```

In [147…
```
try:
    autopilot_model_arn
except NameError:
    create_model_response = sm.create_model(
        Containers=best_candidate["InferenceContainers"], ModelName=autopilot_model_name, ExecutionRoleArn=role
    )
    autopilot_model_arn = create_model_response["ModelArn"]
    print("[OK] Created Autopilot Model: {}".format(autopilot_model_arn))
```

In [148…
```
%store autopilot_model_arn
```

```
Stored 'autopilot_model_arn' (str)
```

Define EndpointConfig Name

In [149…
```
timestamp_suffix = strftime("%d-%H-%M-%S", gmtime())
epc_name = "automl-dm-epc-" + timestamp_suffix

print(epc_name)
```

```
automl-dm-epc-17-03-55-53
```

Define REST Endpoint Name fo rAutopilot Model

In [150…
```
%store -r autopilot_endpoint_name
```

In [151…
```
timestamp_suffix = strftime("%d-%H-%M-%S", gmtime())

try:
    autopilot_endpoint_name
except NameError:
    autopilot_endpoint_name = "automl-dm-ep-" + timestamp_suffix
    print("[OK] Created Autopilot Endpoint Name {}: ".format(autopilot_endpoint_name))
```

In [152…
```
variant_name = "automl-dm-variant-" + timestamp_suffix
print("[OK] Created Endpoint Variant Name {}: ".format(variant_name))
```

```
[OK] Created Endpoint Variant Name automl-dm-variant-17-03-55-53:
```

In [153…
```
%store autopilot_endpoint_name
```

```
Stored 'autopilot_endpoint_name' (str)
```

In [154…
```
ep_config = sm.create_endpoint_config(
    EndpointConfigName=epc_name,
    ProductionVariants=[
        {
            "InstanceType": "ml.m5.large",
            "InitialInstanceCount": 1,
            "ModelName": autopilot_model_name,
            "VariantName": variant_name,
        }
    ],
)
```

In [155…
```
%store -r autopilot_endpoint_arn
```

In [156…
```
try:
    autopilot_endpoint_arn
except NameError:
    create_endpoint_response = sm.create_endpoint(EndpointName=autopilot_endpoint_name, EndpointConfigName=epc_name)
    autopilot_endpoint_arn = create_endpoint_response["EndpointArn"]
    print(autopilot_endpoint_arn)
```

In [157…
```
%store autopilot_endpoint_arn
```

```
Stored 'autopilot_endpoint_arn' (str)
```

In [158…

```
from IPython.core.display import display, HTML

display(
    HTML(
        '<b>Review <a target="blank" href="https://console.aws.amazon.com/sagemaker/home?region={}#/endpoints/{}">Sag
            region, autopilot_endpoint_name
        )
    )
)
```

Review SageMaker REST Endpoint

Store Variables

In [159…

```
%store
```

```
Stored variables and their in-db values:
auto_ml_job_name                          -> 'automl-dm-16-20-10-05'
autopilot_endpoint_arn                    -> 'arn:aws:sagemaker:us-east-1:614093401978:endpoint
autopilot_endpoint_name                   -> 'automl-dm-ep-16-22-24-43'
autopilot_model_arn                       -> 'arn:aws:sagemaker:us-east-1:614093401978:model/au
autopilot_model_name                      -> 'automl-dm-model-16-22-20-58'
autopilot_train_s3_uri                    -> 's3://sagemaker-us-east-1-614093401978/data/df_aut
ingest_create_athena_db_passed            -> True
s3_private_path_csv                       -> 's3://sagemaker-us-east-1-614093401978/cell_data'
s3_public_path_clsm                       -> 's3://team4rawdatasets/CSV/Input/OHSU_BeatAML_Clin
s3_public_path_csv                        -> 's3://gdc-beataml1.0-crenolanib-phs001628-2-open/'
s3_public_path_pi                         -> 's3://team4rawdatasets/CSV/Input/OpenCell_ProteinI
setup_dependencies_passed                 -> True
setup_iam_roles_passed                    -> True
setup_instance_check_passed               -> True
setup_s3_bucket_passed                    -> True
```

Release Resources

In [160…

```
%%html

<p><b>Shutting down your kernel for this notebook to release resources.</b></p>
<button class="sm-command-button" data-commandlinker-command="kernelmenu:shutdown" style="display:none;">Shutdown Ker

<script>
try {
    els = document.getElementsByClassName("sm-command-button");
    els[0].click();
}
catch(err) {
    // NoOp
}
</script>
```

Shutting down your kernel for this notebook to release resources.

In [161…

```
%%javascript

try {
    Jupyter.notebook.save_checkpoint();
    Jupyter.notebook.session.delete();
}
catch(err) {
    // NoOp
}
```