```
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage.
It includes your credentials.
# You might want to remove those credentials before you share the
notebook.
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='0lPA0sFxh7Tft_iFUWVuOhK3teaEfu-vbrJi2dZaeijT',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.us.cloud-object-
storage.appdomain.cloud')

bucket = 'ckdprediction-donotdelete-pr-ah8ipir3oprcay'
object_key = 'chronickidneydisease.csv'

body = cos_client.get_object(Bucket=bucket,Key=object_key)['Body']
# add missing __iter__ method, so pandas accepts body as file-like
object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType(
__iter__, body )

data = pd.read_csv(body)
data.head()
```

	id	age	bp	sg	al	su	rbc	pc	pcc	ba
0	0	48.0	80.0	1.020	1.0	0.0	NaN	normal	notpresent	notpresent
1	1	7.0	50.0	1.020	4.0	0.0	NaN	normal	notpresent	notpresent
2	2	62.0	80.0	1.010	2.0	3.0	normal	normal	notpresent	notpresent
3	3	48.0	70.0	1.005	4.0	0.0	normal	abnormal	present	notpresent
4	4	51.0	80.0	1.010	2.0	0.0	normal	normal	notpresent	notpresent

	...	pcv	wc	rc	htn	dm	cad	appet	pe	ane	classification
0	...	44	7800	5.2	yes	yes	no	good	no	no	ckd
1	...	38	6000	NaN	no	no	no	good	no	no	ckd
2	...	31	7500	NaN	no	yes	no	poor	no	yes	ckd
3	...	32	6700	3.9	yes	no	no	poor	yes	yes	ckd

```
4  ...   35  7300  4.6   no   no   no  good   no   no            ckd

[5 rows x 26 columns]
```

```
import pandas as pd
import numpy as np
from collections import Counter as c
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.metrics import accuracy_score, confusion_matrix
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from sklearn.linear_model import LogisticRegression
import pickle
```

```
data.drop(["id"],axis=1,inplace=True)
data.columns=['age','blood_pressure','specific_gravity','albumin','sug
ar','red_blood_cells','pus_cell','pus_cell_clumps','bacteria',
              'blood glucose
random','blood_urea','serum_creatinine','sodium','potassium','hemoglob
in','packed_cell_volume',

'white_blood_cell_count','red_blood_cell_count','hypertension','diabet
esmellitus','coronary_artery_disease',
               'appetite','pedal_edema','anemia','class']
```

```
data
```

```
      age  blood_pressure  specific_gravity  albumin  sugar
red_blood_cells  \
0    48.0            80.0             1.020      1.0    0.0
NaN
1     7.0            50.0             1.020      4.0    0.0
NaN
2    62.0            80.0             1.010      2.0    3.0
normal
3    48.0            70.0             1.005      4.0    0.0
normal
4    51.0            80.0             1.010      2.0    0.0
normal
..    ...             ...               ...      ...    ...
...
395  55.0            80.0             1.020      0.0    0.0
normal
396  42.0            70.0             1.025      0.0    0.0
normal
397  12.0            80.0             1.020      0.0    0.0
normal
398  17.0            60.0             1.025      0.0    0.0
normal
399  58.0            80.0             1.025      0.0    0.0
```

```
normal

     pus_cell pus_cell_clumps    bacteria  blood glucose
random  ...  \
0      normal      notpresent  notpresent                121.0  ...

1      normal      notpresent  notpresent                  NaN  ...

2      normal      notpresent  notpresent                423.0  ...

3    abnormal         present  notpresent                117.0  ...

4      normal      notpresent  notpresent                106.0  ...

..        ...             ...         ...                  ...  ...

395    normal      notpresent  notpresent                140.0  ...

396    normal      notpresent  notpresent                 75.0  ...

397    normal      notpresent  notpresent                100.0  ...

398    normal      notpresent  notpresent                114.0  ...

399    normal      notpresent  notpresent                131.0  ...


     packed_cell_volume  white_blood_cell_count  red_blood_cell_count
\
0                    44                    7800                   5.2

1                    38                    6000                   NaN

2                    31                    7500                   NaN

3                    32                    6700                   3.9

4                    35                    7300                   4.6

..                  ...                     ...                   ...

395                  47                    6700                   4.9

396                  54                    7800                   6.2

397                  49                    6600                   5.4

398                  51                    7200                   5.9
```

```
399                 53                    6800                    6.1


     hypertension  diabetesmellitus coronary_artery_disease
appetite  \
0             yes               yes                       no     good

1              no                no                       no     good

2              no               yes                       no     poor

3             yes                no                       no     poor

4              no                no                       no     good

..            ...               ...                      ...      ...

395            no                no                       no     good

396            no                no                       no     good

397            no                no                       no     good

398            no                no                       no     good

399            no                no                       no     good


    pedal_edema anemia   class
0            no     no     ckd
1            no     no     ckd
2            no    yes     ckd
3           yes    yes     ckd
4            no     no     ckd
..          ...    ...     ...
395          no     no  notckd
396          no     no  notckd
397          no     no  notckd
398          no     no  notckd
399          no     no  notckd

[400 rows x 25 columns]

catcols=set(data.dtypes[data.dtypes=='O'].index.values)
print(catcols)

{'pedal_edema', 'bacteria', 'red_blood_cells',
'white_blood_cell_count', 'packed_cell_volume', 'anemia',
```

```
'coronary_artery_disease', 'diabetesmellitus', 'red_blood_cell_count',
'class', 'hypertension', 'appetite', 'pus_cell', 'pus_cell_clumps'}
```

```
contcols=set(data.dtypes[data.dtypes!='O'].index.values)
print(contcols)
```

```
{'hemoglobin', 'age', 'serum_creatinine', 'specific_gravity',
'potassium', 'albumin', 'sugar', 'blood_pressure', 'blood_urea',
'blood glucose random', 'sodium'}
```

```
catcols.remove('red_blood_cell_count')
catcols.remove('packed_cell_volume')
catcols.remove('white_blood_cell_count')
catcols.add('specific_gravity')
catcols.add('albumin')
catcols.add('sugar')
print(catcols)
```

```
{'pedal_edema', 'bacteria', 'red_blood_cells', 'anemia',
'coronary_artery_disease', 'specific_gravity', 'diabetesmellitus',
'albumin', 'class', 'sugar', 'hypertension', 'appetite', 'pus_cell',
'pus_cell_clumps'}
```

```
contcols.remove('specific_gravity')
contcols.remove('albumin')
contcols.remove('sugar')
contcols.add('red_blood_cell_count')
contcols.add('packed_cell_volume')
contcols.add('white_blood_cell_count')
print(contcols)
```

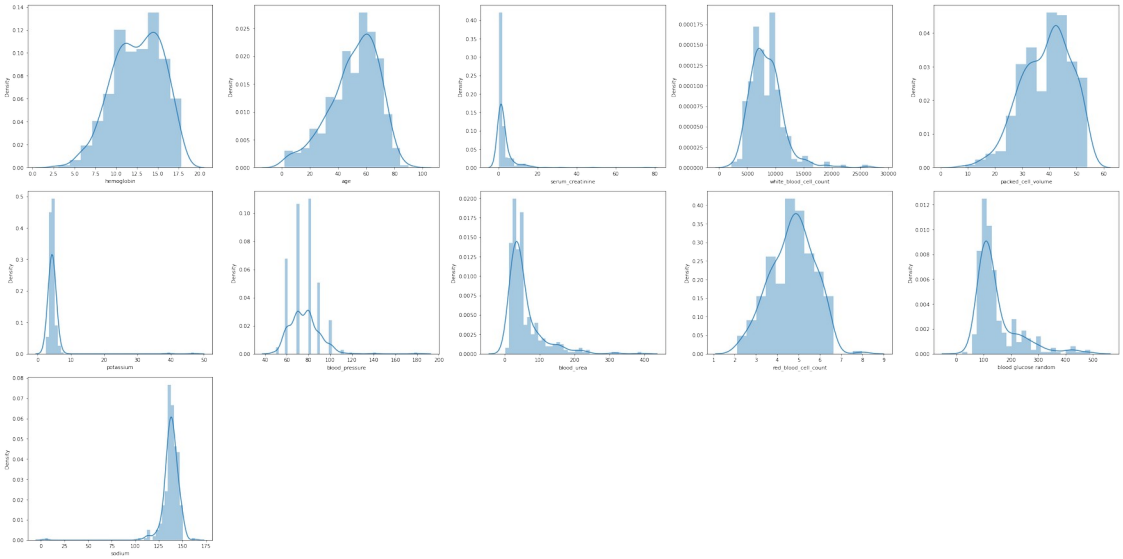
```
{'hemoglobin', 'age', 'serum_creatinine', 'white_blood_cell_count',
'packed_cell_volume', 'potassium', 'blood_pressure', 'blood_urea',
'red_blood_cell_count', 'blood glucose random', 'sodium'}
```

```
#Get unique values from categorical columns

for col in catcols:
    print(f"{col} has {data[col].unique()} values \n")
```

```
pedal_edema has ['no' 'yes' nan] values

bacteria has ['notpresent' 'present' nan] values

red_blood_cells has [nan 'normal' 'abnormal'] values

anemia has ['no' 'yes' nan] values

coronary_artery_disease has ['no' 'yes' '\tno' nan] values

specific_gravity has [1.02  1.01  1.005 1.015   nan 1.025] values
```

```
diabetesmellitus has ['yes' 'no' ' yes' '\tno' '\tyes' nan] values

albumin has [ 1.  4.  2.  3.  0. nan  5.] values

class has ['ckd' 'ckd\t' 'notckd'] values

sugar has [ 0.  3.  4.  1. nan  2.  5.] values

hypertension has ['yes' 'no' nan] values

appetite has ['good' 'poor' nan] values

pus_cell has ['normal' 'abnormal' nan] values

pus_cell_clumps has ['notpresent' 'present' nan] values
```

```
data['class']=data['class'].replace("ckd\t","ckd")
data['coronary_artery_disease']=data.coronary_artery_disease.replace('
\tno','no')
data['diabetesmellitus']=data.diabetesmellitus.replace(to_replace={'\
tno':'no','\tyes':'yes',' yes':'yes'})

data.packed_cell_volume = pd.to_numeric(data.packed_cell_volume,
errors='coerce')
data.white_blood_cell_count =
pd.to_numeric(data.white_blood_cell_count, errors='coerce')
data.red_blood_cell_count = pd.to_numeric(data.red_blood_cell_count,
errors='coerce')

#To handle missing values

plt.figure(figsize = (30,15))
plotnumber =1
for column in contcols:
    if plotnumber <= 14:
        ax = plt.subplot(3, 5, plotnumber)
        sns.distplot(data[column])
        plt.xlabel(column)

    plotnumber += 1

plt.tight_layout()
plt.show()
```



```
data.isna().sum()
```

```
age                        9
blood_pressure            12
specific_gravity          47
albumin                   46
sugar                     49
red_blood_cells          152
pus_cell                  65
pus_cell_clumps            4
bacteria                   4
blood glucose random      44
blood_urea                19
serum_creatinine          17
sodium                    87
potassium                 88
hemoglobin                52
packed_cell_volume        71
white_blood_cell_count   106
red_blood_cell_count     131
hypertension               2
diabetesmellitus           2
coronary_artery_disease    2
appetite                   1
pedal_edema                1
anemia                     1
class                      0
dtype: int64
```

```
data['blood glucose random'].fillna(data['blood glucose
random'].mean(),inplace=True)
data['blood_pressure'].fillna(data['blood_pressure'].mean(),inplace=Tr
ue)
data['blood_urea'].fillna(data['blood_urea'].mean(),inplace=True)
```

```
data['hemoglobin'].fillna(data['hemoglobin'].mean(),inplace=True)
data['packed_cell_volume'].fillna(data['packed_cell_volume'].mean(),in
place=True)
data['potassium'].fillna(data['potassium'].mean(),inplace=True)
data['red_blood_cell_count'].fillna(data['red_blood_cell_count'].mean(
),inplace=True)
data['serum_creatinine'].fillna(data['serum_creatinine'].mean(),inplac
e=True)
data['sodium'].fillna(data['sodium'].mean(),inplace=True)
data['white_blood_cell_count'].fillna(data['white_blood_cell_count'].m
ean(),inplace=True)

data['age'].fillna(data['age'].mode()[0],inplace=True)
data['hypertension'].fillna(data['hypertension'].mode()
[0],inplace=True)
data['pus_cell_clumps'].fillna(data['pus_cell_clumps'].mode()
[0],inplace=True)
data['appetite'].fillna(data['appetite'].mode()[0],inplace=True)
data['albumin'].fillna(data['albumin'].mode()[0],inplace=True)
data['pus_cell'].fillna(data['pus_cell'].mode()[0],inplace=True)
data['red_blood_cells'].fillna(data['red_blood_cells'].mode()
[0],inplace=True)
data['coronary_artery_disease'].fillna(data['coronary_artery_disease']
.mode()[0],inplace=True)
data['bacteria'].fillna(data['bacteria'].mode()[0],inplace=True)
data['anemia'].fillna(data['anemia'].mode()[0],inplace=True)
data['sugar'].fillna(data['sugar'].mode()[0],inplace=True)
data['diabetesmellitus'].fillna(data['diabetesmellitus'].mode()
[0],inplace=True)
data['pedal_edema'].fillna(data['pedal_edema'].mode()[0],inplace=True)
data['specific_gravity'].fillna(data['specific_gravity'].mode()
[0],inplace=True)
LabelEncoder

sklearn.preprocessing._label.LabelEncoder

#Label Encoder

from sklearn.preprocessing import LabelEncoder
for i in catcols:
    ("LABEL ENCODING OF:",i)
    LEi=LabelEncoder()
    print(c(data[i]))
    data[i]=LEi.fit_transform(data[i])
    print(c(data[i]))
    print("*"*100)

Counter({'no': 324, 'yes': 76})
Counter({0: 324, 1: 76})
****************************************************************************
****************************
```

```
Counter({'notpresent': 378, 'present': 22})
Counter({0: 378, 1: 22})
**********************************************************************
******************************
Counter({'normal': 353, 'abnormal': 47})
Counter({1: 353, 0: 47})
**********************************************************************
******************************
Counter({'no': 340, 'yes': 60})
Counter({0: 340, 1: 60})
**********************************************************************
******************************
Counter({'no': 366, 'yes': 34})
Counter({0: 366, 1: 34})
**********************************************************************
******************************
Counter({1.02: 153, 1.01: 84, 1.025: 81, 1.015: 75, 1.005: 7})
Counter({3: 153, 1: 84, 4: 81, 2: 75, 0: 7})
**********************************************************************
******************************
Counter({'no': 263, 'yes': 137})
Counter({0: 263, 1: 137})
**********************************************************************
******************************
Counter({0.0: 245, 1.0: 44, 2.0: 43, 3.0: 43, 4.0: 24, 5.0: 1})
Counter({0: 245, 1: 44, 2: 43, 3: 43, 4: 24, 5: 1})
**********************************************************************
******************************
Counter({'ckd': 250, 'notckd': 150})
Counter({0: 250, 1: 150})
**********************************************************************
******************************
Counter({0.0: 339, 2.0: 18, 3.0: 14, 4.0: 13, 1.0: 13, 5.0: 3})
Counter({0: 339, 2: 18, 3: 14, 4: 13, 1: 13, 5: 3})
**********************************************************************
******************************
Counter({'no': 253, 'yes': 147})
Counter({0: 253, 1: 147})
**********************************************************************
******************************
Counter({'good': 318, 'poor': 82})
Counter({0: 318, 1: 82})
**********************************************************************
******************************
Counter({'normal': 324, 'abnormal': 76})
Counter({1: 324, 0: 76})
**********************************************************************
******************************
Counter({'notpresent': 358, 'present': 42})
Counter({0: 358, 1: 42})
```

```
**************************************************************************
****************************
```

```
#Data split into train and test set
```

```
selcols=['age','blood_urea','blood glucose
random','coronary_artery_disease','anemia','pus_cell','red_blood_cells
',
         'diabetesmellitus','pedal_edema']
```

```
x=pd.DataFrame(data,columns=selcols)
y=pd.DataFrame(data,columns=['class'])
print(x.shape)
print(y.shape)
```

```
(400, 9)
(400, 1)
```

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,rando
m_state=2)
print(x_train.shape)
print(y_train.shape)
print(x_test.shape)
print(y_test.shape)
```

```
(320, 9)
(320, 1)
(80, 9)
(80, 1)
```

```
#Model building using Logistic Regression
```

```
from sklearn.linear_model import LogisticRegression
model=LogisticRegression(solver ='lbfgs',max_iter=500)
print('LogisticRegression\n')
model.fit(x_train.values,y_train.values.ravel())
prediction = model.predict(x_test)
from sklearn.metrics import confusion_matrix
print('confusion_matrix')
print(confusion_matrix(prediction,y_test))
print('\n')
print('accuracy_score')
print(accuracy_score(prediction,y_test))
print('\n')
```

```
LogisticRegression

confusion_matrix
[[49  0]
 [ 5 26]]
```

```
accuracy_score
0.9375
```

```
/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/sklearn/
base.py:443: UserWarning: X has feature names, but LogisticRegression
was fitted without feature names
  warnings.warn(
```

```
#Using RandomForestClassifier

from sklearn.ensemble import RandomForestClassifier
model = RandomForestClassifier()
model.fit(x_train , y_train)
prediction = model.predict(x_test)
from sklearn.metrics import confusion_matrix
print('RandomForest\n')
print('confusion_matrix')
print(confusion_matrix(prediction,y_test))
print('\n')
print('accuracy_score')
print(accuracy_score(prediction,y_test))
print('\n')
```

```
RandomForest

confusion_matrix
[[52  2]
 [ 2 24]]


accuracy_score
0.95
```

```
/tmp/wsuser/ipykernel_269/2865502932.py:3: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change
the shape of y to (n_samples,), for example using ravel().
  model.fit(x_train , y_train)
```

IBM-Deployment

```
!pip install -U ibm-watson-machine-learning
```

```
Requirement already satisfied: ibm-watson-machine-learning in
/opt/conda/envs/Python-3.9/lib/python3.9/site-packages (1.0.257)
Requirement already satisfied: urllib3 in
/opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-
watson-machine-learning) (1.26.7)
Requirement already satisfied: certifi in
/opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-
watson-machine-learning) (2022.9.24)
Requirement already satisfied: requests in /opt/conda/envs/Python-
3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning)
(2.26.0)
Requirement already satisfied: pandas<1.5.0,>=0.24.2 in
/opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-
watson-machine-learning) (1.3.4)
Requirement already satisfied: tabulate in /opt/conda/envs/Python-
3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning)
(0.8.9)
Requirement already satisfied: lomond in
/opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-
watson-machine-learning) (0.3.3)
Requirement already satisfied: ibm-cos-sdk==2.11.* in
/opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-
watson-machine-learning) (2.11.0)
Requirement already satisfied: packaging in /opt/conda/envs/Python-
3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning)
(21.3)
Requirement already satisfied: importlib-metadata in
/opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-
watson-machine-learning) (4.8.2)
Requirement already satisfied: ibm-cos-sdk-s3transfer==2.11.0 in
/opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-
sdk==2.11.*->ibm-watson-machine-learning) (2.11.0)
Requirement already satisfied: ibm-cos-sdk-core==2.11.0 in
/opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-
sdk==2.11.*->ibm-watson-machine-learning) (2.11.0)
Requirement already satisfied: jmespath<1.0.0,>=0.7.1 in
/opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-
sdk==2.11.*->ibm-watson-machine-learning) (0.10.0)
Requirement already satisfied: python-dateutil<3.0.0,>=2.1 in
/opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-
sdk-core==2.11.0->ibm-cos-sdk==2.11.*->ibm-watson-machine-learning)
(2.8.2)
Requirement already satisfied: pytz>=2017.3 in /opt/conda/envs/Python-
3.9/lib/python3.9/site-packages (from pandas<1.5.0,>=0.24.2->ibm-
watson-machine-learning) (2021.3)
Requirement already satisfied: numpy>=1.17.3 in
/opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from
pandas<1.5.0,>=0.24.2->ibm-watson-machine-learning) (1.20.3)
Requirement already satisfied: six>=1.5 in /opt/conda/envs/Python-
3.9/lib/python3.9/site-packages (from python-dateutil<3.0.0,>=2.1-
```

```
>ibm-cos-sdk-core==2.11.0->ibm-cos-sdk==2.11.*->ibm-watson-machine-
learning) (1.15.0)
Requirement already satisfied: idna<4,>=2.5 in /opt/conda/envs/Python-
3.9/lib/python3.9/site-packages (from requests->ibm-watson-machine-
learning) (3.3)
Requirement already satisfied: charset-normalizer~=2.0.0 in
/opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from requests-
>ibm-watson-machine-learning) (2.0.4)
Requirement already satisfied: zipp>=0.5 in /opt/conda/envs/Python-
3.9/lib/python3.9/site-packages (from importlib-metadata->ibm-watson-
machine-learning) (3.6.0)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in
/opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from
packaging->ibm-watson-machine-learning) (3.0.4)
```

```
from ibm_watson_machine_learning import APIClient
import json
```

Authenticate and set space

```
wml_credentials={
    "apikey":"nLqYPcmmMXzouXWU9VTlIupFXMTPBw7NVDrx71IyAIY-",
    "url":"https://us-south.ml.cloud.ibm.com"
}
```

```
wml_client=APIClient(wml_credentials)
```

```
wml_client.spaces.list()
```

```
Note: 'limit' is not provided. Only first 50 records will be displayed
if the number of records exceed 50
------------------------------------  -------------
------------------------
ID                                    NAME           CREATED
425c3a67-aed6-4812-a256-b34098a7660c  ckdprediction  2022-11-
21T15:01:29.078Z
------------------------------------  -------------
------------------------
```

```
SPACE_ID="425c3a67-aed6-4812-a256-b34098a7660c"
```

```
wml_client.set.default_space(SPACE_ID)
```

```
'SUCCESS'
```

```
wml_client.software_specifications.list(500)
```

```
------------------------------  ------------------------------------
----
NAME                            ASSET_ID
TYPE
default_py3.6                   0062b8c9-8b7d-44a0-a9b9-46c416adcbd9
```

base
kernel-spark3.2-scala2.12 020d69ce-7ac1-5e68-ac1a-31189867356a
base
pytorch-onnx_1.3-py3.7-edt 069ea134-3346-5748-b513-49120e15d288
base
scikit-learn_0.20-py3.6 09c5a1d0-9c1e-4473-a344-eb7b665ff687
base
spark-mllib_3.0-scala_2.12 09f4cff0-90a7-5899-b9ed-1ef348aebdee
base
pytorch-onnx_rt22.1-py3.9 0b848dd4-e681-5599-be41-b5f6fccc6471
base
ai-function_0.1-py3.6 0cdb0f1e-5376-4f4d-92dd-da3b69aa9bda
base
shiny-r3.6 0e6e79df-875e-4f24-8ae9-62dcc2148306
base
tensorflow_2.4-py3.7-horovod 1092590a-307d-563d-9b62-4eb7d64b3f22
base
pytorch_1.1-py3.6 10ac12d6-6b30-4ccd-8392-3e922c096a92
base
tensorflow_1.15-py3.6-ddl 111e41b3-de2d-5422-a4d6-bf776828c4b7
base
autoai-kb_rt22.2-py3.10 125b6d9a-5b1f-5e8d-972a-b251688ccf40
base
runtime-22.1-py3.9 12b83a17-24d8-5082-900f-0ab31fbfd3cb
base
scikit-learn_0.22-py3.6 154010fa-5b3b-4ac1-82af-4d5ee5abbc85
base
default_r3.6 1b70aec3-ab34-4b87-8aa0-a4a3c8296a36
base
pytorch-onnx_1.3-py3.6 1bc6029a-cc97-56da-b8e0-39c3880dbbe7
base
kernel-spark3.3-r3.6 1c9e5454-f216-59dd-a20e-474a5cdf5988
base
pytorch-onnx_rt22.1-py3.9-edt 1d362186-7ad5-5b59-8b6c-9d0880bde37f
base
tensorflow_2.1-py3.6 1eb25b84-d6ed-5dde-b6a5-3fbdf1665666
base
spark-mllib_3.2 20047f72-0a98-58c7-9ff5-a77b012eb8f5
base
tensorflow_2.4-py3.8-horovod 217c16f6-178f-56bf-824a-b19f20564c49
base
runtime-22.1-py3.9-cuda 26215f05-08c3-5a41-a1b0-da66306ce658
base
do_py3.8 295addb5-9ef9-547e-9bf4-92ae3563e720
base
autoai-ts_3.8-py3.8 2aa0c932-798f-5ae9-abd6-15e0c2402fb5
base
tensorflow_1.15-py3.6 2b73a275-7cbf-420b-a912-eae7f436e0bc
base
kernel-spark3.3-py3.9 2b7961e2-e3b1-5a8c-a491-482c8368839a

```
base
pytorch_1.2-py3.6                2c8ef57d-2687-4b7d-acce-01f94976dac1
base
spark-mllib_2.3                  2e51f700-bca0-4b0d-88dc-5c6791338875
base
pytorch-onnx_1.1-py3.6-edt       32983cea-3f32-4400-8965-dde874a8d67e
base
spark-mllib_3.0-py37             36507ebe-8770-55ba-ab2a-eafe787600e9
base
spark-mllib_2.4                  390d21f8-e58b-4fac-9c55-d7ceda621326
base
autoai-ts_rt22.2-py3.10          396b2e83-0953-5b86-9a55-7ce1628a406f
base
xgboost_0.82-py3.6               39e31acd-5f30-41dc-ae44-60233c80306e
base
pytorch-onnx_1.2-py3.6-edt       40589d0e-7019-4e28-8daa-fb03b6f4fe12
base
pytorch-onnx_rt22.2-py3.10       40e73f55-783a-5535-b3fa-0c8b94291431
base
default_r36py38                  41c247d3-45f8-5a71-b065-8580229facf0
base
autoai-ts_rt22.1-py3.9           4269d26e-07ba-5d40-8f66-2d495b0c71f7
base
autoai-obm_3.0                   42b92e18-d9ab-567f-988a-4240ba1ed5f7
base
pmml-3.0_4.3                     493bcb95-16f1-5bc5-bee8-81b8af80e9c7
base
spark-mllib_2.4-r_3.6            49403dff-92e9-4c87-a3d7-a42d0021c095
base
xgboost_0.90-py3.6               4ff8d6c2-1343-4c18-85e1-689c965304d3
base
pytorch-onnx_1.1-py3.6           50f95b2a-bc16-43bb-bc94-b0bed208c60b
base
autoai-ts_3.9-py3.8              52c57136-80fa-572e-8728-a5e7cbb42cde
base
spark-mllib_2.4-scala_2.11       55a70f99-7320-4be5-9fb9-9edb5a443af5
base
spark-mllib_3.0                  5c1b0ca2-4977-5c2e-9439-ffd44ea8ffe9
base
autoai-obm_2.0                   5c2e37fa-80b8-5e77-840f-d912469614ee
base
spss-modeler_18.1                5c3cad7e-507f-4b2a-a9a3-ab53a21dee8b
base
cuda-py3.8                       5d3232bf-c86b-5df4-a2cd-7bb870a1cd4e
base
runtime-22.2-py3.10-xc           5e8cddff-db4a-5a6a-b8aa-2d4af9864dab
base
autoai-kb_3.1-py3.7              632d4b22-10aa-5180-88f0-f52dfb6444d7
base
pytorch-onnx_1.7-py3.8           634d3cdc-b562-5bf9-a2d4-ea90a478456b
```

base
spark-mllib_2.3-r_3.6 6586b9e3-ccd6-4f92-900f-0f8cb2bd6f0c
base
tensorflow_2.4-py3.7 65e171d7-72d1-55d9-8ebb-f813d620c9bb
base
spss-modeler_18.2 687eddc9-028a-4117-b9dd-e57b36f1efa5
base
pytorch-onnx_1.2-py3.6 692a6a4d-2c4d-45ff-a1ed-b167ee55469a
base
spark-mllib_2.3-scala_2.11 7963efe5-bbec-417e-92cf-0574e21b4e8d
base
spark-mllib_2.4-py37 7abc992b-b685-532b-a122-a396a3cdbaab
base
caffe_1.0-py3.6 7bb3dbe2-da6e-4145-918d-b6d84aa93b6b
base
pytorch-onnx_1.7-py3.7 812c6631-42b7-5613-982b-02098e6c909c
base
cuda-py3.6 82c79ece-4d12-40e6-8787-a7b9e0f62770
base
tensorflow_1.15-py3.6-horovod 8964680e-d5e4-5bb8-919b-8342c6c0dfd8
base
hybrid_0.1 8c1a58c6-62b5-4dc4-987a-df751c2756b6
base
pytorch-onnx_1.3-py3.7 8d5d8a87-a912-54cf-81ec-3914adaa988d
base
caffe-ibm_1.0-py3.6 8d863266-7927-4d1e-97d7-56a7f4c0a19b
base
runtime-22.2-py3.10-cuda 8ef391e4-ef58-5d46-b078-a82c211c1058
base
spss-modeler_17.1 902d0051-84bd-4af6-ab6b-8f6aa6fdeabb
base
do_12.10 9100fd72-8159-4eb9-8a0b-a87e12eefa36
base
do_py3.7 9447fa8b-2051-4d24-9eef-5acb0e3c59f8
base
spark-mllib_3.0-r_3.6 94bb6052-c837-589d-83f1-f4142f219e32
base
cuda-py3.7-opence 94e9652b-7f2d-59d5-ba5a-23a414ea488f
base
nlp-py3.8 96e60351-99d4-5a1c-9cc0-473ac1b5a864
base
cuda-py3.7 9a44990c-1aa1-4c7d-baf8-c4099011741c
base
hybrid_0.2 9b3f9040-9cee-4ead-8d7a-780600f542f7
base
spark-mllib_3.0-py38 9f7a8fc1-4d3c-5e65-ab90-41fa8de2d418
base
autoai-kb_3.3-py3.7 a545cca3-02df-5c61-9e88-998b09dc79af
base
spark-mllib_3.0-py39 a6082a27-5acc-5163-b02c-6b96916eb5e0

base
runtime-22.1-py3.9-do   a7e7dbf1-1d03-5544-994d-e5ec845ce99a
base
default_py3.8   ab9e1b80-f2ce-592c-a7d2-4f2344f77194
base
tensorflow_rt22.1-py3.9   acd9c798-6974-5d2f-a657-ce06e986df4d
base
kernel-spark3.2-py3.9   ad7033ee-794e-58cf-812e-a95f4b64b207
base
autoai-obm_2.0 with Spark 3.0   af10f35f-69fa-5d66-9bf5-acb58434263a
base
runtime-22.2-py3.10   b56101f1-309d-549b-a849-eaa63f77b2fb
base
default_py3.7_opence   c2057dd4-f42c-5f77-a02f-72bdbd3282c9
base
tensorflow_2.1-py3.7   c4032338-2a40-500a-beef-b01ab2667e27
base
do_py3.7_opence   cc8f8976-b74a-551a-bb66-6377f8d865b4
base
spark-mllib_3.3   d11f2434-4fc7-58b7-8a62-755da64fdaf8
base
autoai-kb_3.0-py3.6   d139f196-e04b-5d8b-9140-9a10ca1fa91a
base
spark-mllib_3.0-py36   d82546d5-dd78-5fbb-9131-2ec309bc56ed
base
autoai-kb_3.4-py3.8   da9b39c3-758c-5a4f-9cfd-457dd4d8c395
base
kernel-spark3.2-r3.6   db2fe4d6-d641-5d05-9972-73c654c60e0a
base
autoai-kb_rt22.1-py3.9   db6afe93-665f-5910-b117-d879897404d9
base
tensorflow_rt22.1-py3.9-horovod dda170cc-ca67-5da7-9b7a-cf84c6987fae
base
autoai-ts_1.0-py3.7   deef04f0-0c42-5147-9711-89f9904299db
base
tensorflow_2.1-py3.7-horovod   e384fce5-fdd1-53f8-bc71-11326c9c635f
base
default_py3.7   e4429883-c883-42b6-87a8-f419d64088cd
base
do_22.1   e51999ba-6452-5f1f-8287-17228b88b652
base
autoai-obm_3.2   eae86aab-da30-5229-a6a6-1d0d4e368983
base
runtime-22.2-r4.2   ec0a3d28-08f7-556c-9674-ca7c2dba30bd
base
tensorflow_rt22.2-py3.10   f65bd165-f057-55de-b5cb-f97cf2c0f393
base
do_20.1   f686cdd9-7904-5f9d-a732-01b0d6b10dc5
base
pytorch-onnx_rt22.2-py3.10-edt   f8a05d07-e7cd-57bb-a10b-23f1d4b837ac

```
base
scikit-learn_0.19-py3.6          f963fa9d-4bb7-5652-9c5d-8d9289ef6ad9
base
tensorflow_2.4-py3.8             fe185c44-9a99-5425-986b-59bd1d2eda46
base
------------------------------  ------------------------------------
----
```

```
#Save and Deploy the model
```

```
import sklearn
sklearn.__version__
```

```
'1.0.2'
```

```
MODEL_NAME='ckdprediction'
DEPLOYMENT_NAME='ckdprediction'
DEMO_MODEL=model
```

```
#set python version
software_spec_uid=wml_client.software_specifications.get_id_by_name('r
untime-22.1-py3.9')
```

```
#setup model meta
model_props={
    wml_client.repository.ModelMetaNames.NAME:MODEL_NAME,
    wml_client.repository.ModelMetaNames.TYPE:'scikit-learn_1.0',

wml_client.repository.ModelMetaNames.SOFTWARE_SPEC_UID:software_spec_u
id
}
```

```
#save model
model_details=wml_client.repository.store_model(
    model=DEMO_MODEL,
    meta_props=model_props,
    training_data=x_train,
    training_target=y_train
)
```

```
model_details
```

```
{'entity': {'hybrid_pipeline_software_specs': [],
  'label_column': 'class',
  'schemas': {'input': [{'fields': [{'name': 'age', 'type':
'float64'},
      {'name': 'blood_urea', 'type': 'float64'},
      {'name': 'blood glucose random', 'type': 'float64'},
      {'name': 'coronary_artery_disease', 'type': 'int64'},
      {'name': 'anemia', 'type': 'int64'},
      {'name': 'pus_cell', 'type': 'int64'},
      {'name': 'red_blood_cells', 'type': 'int64'},
```

```
      {'name': 'diabetesmellitus', 'type': 'int64'},
      {'name': 'pedal_edema', 'type': 'int64'}],
     'id': '1',
     'type': 'struct'}],
   'output': []},
  'software_spec': {'id': '12b83a17-24d8-5082-900f-0ab31fbfd3cb',
   'name': 'runtime-22.1-py3.9'},
  'type': 'scikit-learn_1.0'},
 'metadata': {'created_at': '2022-11-21T15:37:40.781Z',
  'id': '468402ce-7780-4529-8179-6ffdde51025a',
  'modified_at': '2022-11-21T15:37:43.086Z',
  'name': 'ckdprediction',
  'owner': 'IBMid-6610045B99',
  'resource_key': '5eb8b156-1c9e-49c9-90d5-872e6b3ba5cd',
  'space_id': '425c3a67-aed6-4812-a256-b34098a7660c'},
 'system': {'warnings': []}}
```

```
model_id=wml_client.repository.get_model_id(model_details)
model_id
```

```
'468402ce-7780-4529-8179-6ffdde51025a'
```

```
#set meta
deployment_props={

wml_client.deployments.ConfigurationMetaNames.NAME:DEPLOYMENT_NAME,
    wml_client.deployments.ConfigurationMetaNames.ONLINE:{}
}
```

```
#Deploy
deployment=wml_client.deployments.create(
    artifact_uid=model_id,
    meta_props=deployment_props
)
```

```
#######################################################################
#################
```

```
Synchronous deployment creation for uid: '468402ce-7780-4529-8179-
6ffdde51025a' started
```

```
#######################################################################
#################
```

```
initializing
Note: online_url is deprecated and will be removed in a future
release. Use serving_urls instead.
```

```
ready


------------------------------------------------------------------------
-------------------------
Successfully finished deployment creation, deployment_uid='b8bec8b9-
3690-484d-ac4b-111a7da8657f'
------------------------------------------------------------------------
-------------------------
```