

```
IBM Welcom IBM IBM-Pr IBM-EP Problem Our Sh Weekly CKD PI Admin Learn Config Admin Service prc x +
dataplatform.cloud.ibm.com/analytics/notebooks/v2/8acc9522-900a-4735-99e8-75ec28d093cb/view?projectId=b7a55ca9-047d-4a19-b7c1-c0b6e14278ff&context=cp...
IBM Cloud Pak for Data Search in your workspaces Buy Sowniya T's Account Dallas

Projects / CKD prediction / project

In [165]: #importing libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import os

In [166]: 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='UA6sy0lle-Y8BGtaBfubnZAs0h72H94ui8BqRPvF0k0-',
                              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-q5c4njjaizesn6'
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()
```

IBM Welcom IBM IBM-Pr IBM-EP Problem Our Sh Weekly CKD PI Admin Learn Config Admin Service prc x +

dataplatform.cloud.ibm.com/analytics/notebooks/v2/8acc9522-900a-4735-99e8-75ec28d093cb/view?projectId=b7a55ca9-047d-4a19-b7c1-c0b6e14278ff&context=cp...

IBM Cloud Pak for Data Search in your workspaces Buy Sowniya T's Account Dallas ST

Projects / CKD prediction / project

Out[166]:

	id	age	bp	sg	al	su	rbc	pc	pcc	ba	...	pcv	wc	rc	htn	dm	cad	appet	pe	ane	classification
0	0	48.0	80.0	1.020	1.0	0.0	NaN	normal	notpresent	notpresent	...	44	7800	5.2	yes	yes	no	good	no	no	ckd
1	1	7.0	50.0	1.020	4.0	0.0	NaN	normal	notpresent	notpresent	...	38	6000	NaN	no	no	no	good	no	no	ckd
2	2	62.0	80.0	1.010	2.0	3.0	normal	normal	notpresent	notpresent	...	31	7500	NaN	no	yes	no	poor	no	yes	ckd
3	3	48.0	70.0	1.005	4.0	0.0	normal	abnormal	present	notpresent	...	32	6700	3.9	yes	no	no	poor	yes	yes	ckd
4	4	51.0	80.0	1.010	2.0	0.0	normal	normal	notpresent	notpresent	...	35	7300	4.6	no	no	no	good	no	no	ckd

5 rows × 26 columns

In [167]: data.tail(10)

Out[167]:

	id	age	bp	sg	al	su	rbc	pc	pcc	ba	...	pcv	wc	rc	htn	dm	cad	appet	pe	ane	classification
390	390	52.0	80.0	1.025	0.0	0.0	normal	normal	notpresent	notpresent	...	52	6300	5.3	no	no	no	good	no	no	notckd
391	391	36.0	80.0	1.025	0.0	0.0	normal	normal	notpresent	notpresent	...	44	5800	6.3	no	no	no	good	no	no	notckd
392	392	57.0	80.0	1.020	0.0	0.0	normal	normal	notpresent	notpresent	...	46	6600	5.5	no	no	no	good	no	no	notckd
393	393	43.0	60.0	1.025	0.0	0.0	normal	normal	notpresent	notpresent	...	54	7400	5.4	no	no	no	good	no	no	notckd
394	394	50.0	80.0	1.020	0.0	0.0	normal	normal	notpresent	notpresent	...	45	9500	4.6	no	no	no	good	no	no	notckd
395	395	55.0	80.0	1.020	0.0	0.0	normal	normal	notpresent	notpresent	...	47	6700	4.9	no	no	no	good	no	no	notckd
396	396	42.0	70.0	1.025	0.0	0.0	normal	normal	notpresent	notpresent	...	54	7800	6.2	no	no	no	good	no	no	notckd
397	397	12.0	80.0	1.020	0.0	0.0	normal	normal	notpresent	notpresent	...	49	6600	5.4	no	no	no	good	no	no	notckd
398	398	17.0	60.0	1.025	0.0	0.0	normal	normal	notpresent	notpresent	...	51	7200	5.9	no	no	no	good	no	no	notckd
399	399	58.0	80.0	1.025	0.0	0.0	normal	normal	notpresent	notpresent	...	53	6800	6.1	no	no	no	good	no	no	notckd

10 rows × 26 columns

IBM Cloud Pak for Data

Projects / CKD prediction / project

```
In [168]: data.shape
Out[168]: (400, 26)

In [169]: data.columns=['id','age','blood_pressure','specific_gravity','albumin','sugar','red_blood_cells',
                        'pus_cell','pus_cell_clumps','bacteria','blood_glucose_random',
                        'blood_urea','serum_creatinine','sodium','potassium','hemoglobin','packed_cell_volume',
                        'white_blood_cell_count','red_blood_cell_count','hypertension','diabetesmellitus','coronary_artery_disease',
                        'appetite','pedal_edema','anemia','class']

In [170]: data.columns
Out[170]: Index(['id', 'age', 'blood_pressure', 'specific_gravity', 'albumin', 'sugar',
                'red_blood_cells', 'pus_cell', 'pus_cell_clumps', 'bacteria',
                'blood glucose random', 'blood_urea', 'serum_creatinine', 'sodium',
                'potassium', 'hemoglobin', 'packed_cell_volume',
                'white_blood_cell_count', 'red_blood_cell_count', 'hypertension',
                'diabetesmellitus', 'coronary_artery_disease', 'appetite',
                'pedal_edema', 'anemia', 'class'],
                dtype='object')

In [171]: data.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 400 entries, 0 to 399
Data columns (total 26 columns):
#   Column                                Non-Null Count  Dtype
---  ---                                -
0   id                                    400 non-null   int64
1   age                                  391 non-null   float64
2   blood_pressure                       388 non-null   float64
3   specific_gravity                     353 non-null   float64
4   albumin                             354 non-null   float64
5   sugar                               351 non-null   float64
6   red_blood_cells                      248 non-null   object
7   pus_cell                            335 non-null   object
```

IBM Cloud Pak for Data

Projects / CKD prediction / project

```
/   pus_cell                335 non-null   object
8   pus_cell_clumps         396 non-null   object
9   bacteria                 396 non-null   object
10  blood glucose random     356 non-null   float64
11  blood_urea               381 non-null   float64
12  serum_creatinine         383 non-null   float64
13  sodium                   313 non-null   float64
14  potassium                 312 non-null   float64
15  hemoglobin                348 non-null   float64
16  packed_cell_volume        330 non-null   object
17  white_blood_cell_count    295 non-null   object
18  red_blood_cell_count      270 non-null   object
19  hypertension              398 non-null   object
20  diabetesmellitus          398 non-null   object
21  coronary_artery_disease   398 non-null   object
22  appetite                  399 non-null   object
23  pedal_edema               399 non-null   object
24  anemia                    399 non-null   object
25  class                     400 non-null   object
dtypes: float64(11), int64(1), object(14)
memory usage: 81.44 KB

In [172]: data.drop(['id'],axis=1,inplace=True)
data

Out[172]:
```

	age	blood_pressure	specific_gravity	albumin	sugar	red_blood_cells	pus_cell	pus_cell_clumps	bacteria	blood glucose random	...	packed_cell_volume	white_blood_cell_count	red_blood_cell_count	hype
0	48.0	80.0	1.020	1.0	0.0	NaN	normal	notpresent	notpresent	121.0	...	44	7800	5.2	
1	7.0	50.0	1.020	4.0	0.0	NaN	normal	notpresent	notpresent	NaN	...	38	6000	NaN	
2	62.0	80.0	1.010	2.0	3.0	normal	normal	notpresent	notpresent	423.0	...	31	7500	NaN	
3	48.0	70.0	1.005	4.0	0.0	normal	abnormal	present	notpresent	117.0	...	32	6700	3.9	
4	51.0	80.0	1.010	2.0	0.0	normal	normal	notpresent	notpresent	106.0	...	35	7300	4.6	

IBM Cloud Pak for Data

Search in your workspaces

Buy

Sowmiya T's Account

Dallas

ST

Projects / CKD prediction / project

395	55.0	80.0	1.020	0.0	0.0	normal	normal	notpresent	notpresent	140.0	...	47	6700	4.9
396	42.0	70.0	1.025	0.0	0.0	normal	normal	notpresent	notpresent	75.0	...	54	7800	6.2
397	12.0	80.0	1.020	0.0	0.0	normal	normal	notpresent	notpresent	100.0	...	49	6600	5.4
398	17.0	60.0	1.025	0.0	0.0	normal	normal	notpresent	notpresent	114.0	...	51	7200	5.9
399	58.0	80.0	1.025	0.0	0.0	normal	normal	notpresent	notpresent	131.0	...	53	6800	6.1

400 rows x 25 columns

```

In [173]: #fetching categorical column
cat=data.select_dtypes(include=['object']).columns.tolist()
cat

Out[173]: ['red_blood_cells',
'pus_cell',
'pus_cell_clumps',
'bacteria',
'packed_cell_volume',
'white_blood_cell_count',
'red_blood_cell_count',
'hypertension',
'diabetesmellitus',
'coronary_artery_disease',
'appetite',
'pedal_edema',
'anemia',
'class']

In [174]: #removing column which are not categorical
cat.remove('red_blood_cell_count')
cat.remove('packed_cell_volume')
cat.remove('white_blood_cell_count')

```

IBM Cloud Pak for Data

Search in your workspaces

Buy

Sowmiya T's Account

Dallas

ST

Projects / CKD prediction / project

```

cat.remove('white_blood_cell_count')

In [175]: num=data.select_dtypes(include=['float64']).columns.tolist()#fetch numerical column

In [176]: num.remove('specific_gravity')#remove which are not numerical
num.remove('albumin')
num.remove('sugar')

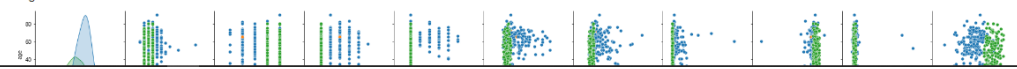
In [177]: #adding column which is numerical
num.append('red_blood_cell_count')
num.append('packed_cell_volume')
num.append('white_blood_cell_count')
num

Out[177]: ['age',
'blood_pressure',
'blood_glucose_random',
'blood_urea',
'serum_creatinine',
'sodium',
'potassium',
'hemoglobin',
'red_blood_cell_count',
'packed_cell_volume',
'white_blood_cell_count']

In [178]: sns.pairplot(data,hue='class')
fig=plt.figure(figsize=(10,5))
fig

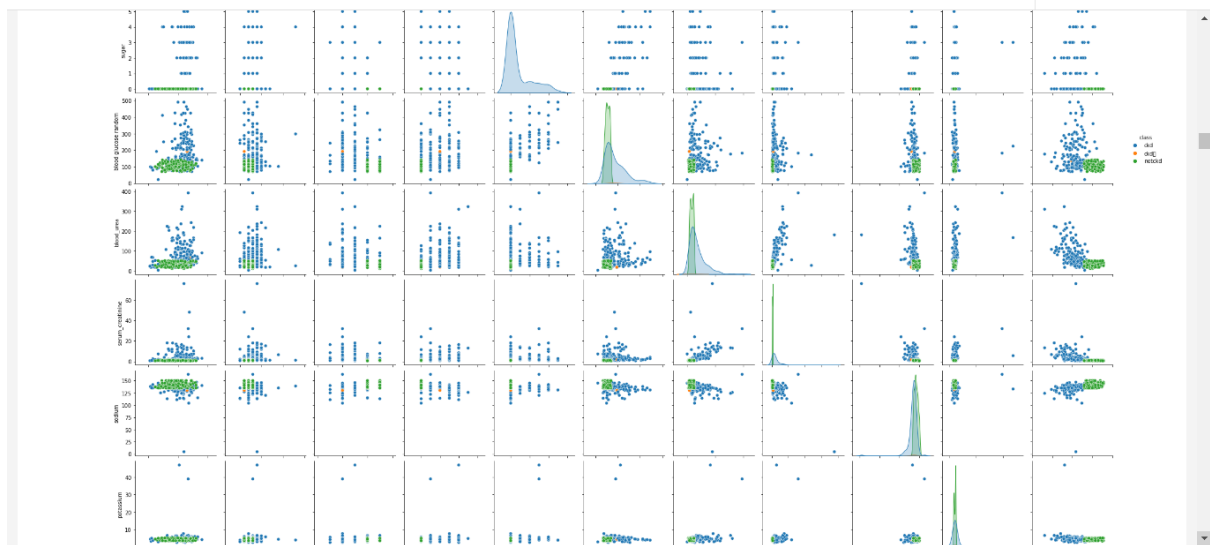
Out[178]: <Figure size 720x360 with 0 Axes>

```

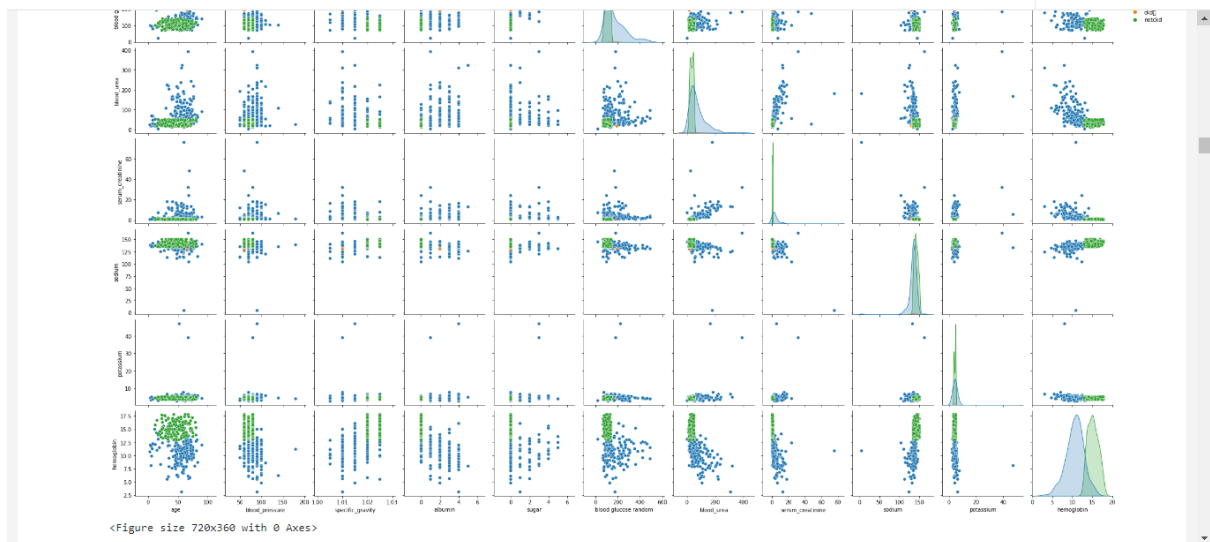


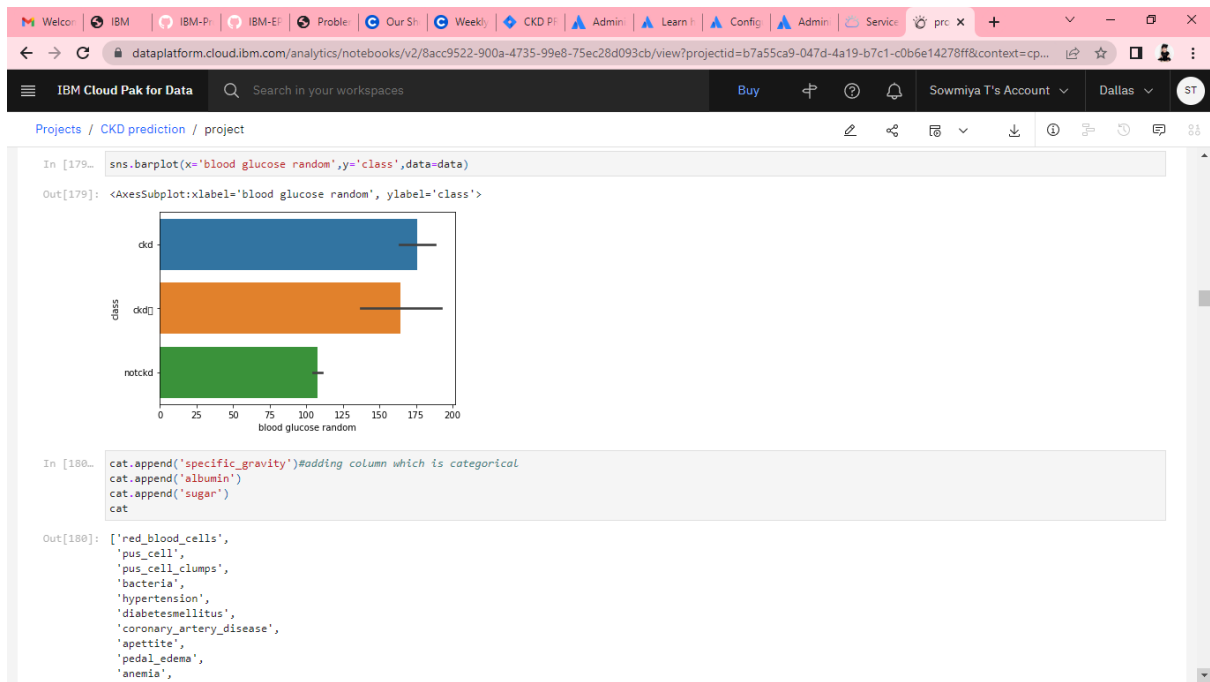


Projects / CKD prediction / project



Projects / CKD prediction / project





IBM Cloud Pak for Data

Projects / CKD prediction / project

```
apetite ,
'pedal_edema',
'anemia',
'class',
'specific_gravity',
'albumin',
'sugar']

In [181]: a=data['coronary_artery_disease'].unique()
#b=data['sugar'].unique()
#c=data['albumin'].unique()
#d=data['specific_gravity'].unique()
#e=data['anemia'].unique()
#f=data['pedal_edema'].unique()#
#g=data['appetite'].unique()
h=data['diabetesmellitus'].unique()
#i=data['bacteria'].unique()
#j=data['hypertension'].unique()#
#k=data['red_blood_cell_count'].unique()
#l=data['pus_cell'].unique()
#m=data['pus_cell_clumps'].unique()

In [182]: a,h
```

Out[182]: (array(['no', 'yes', '\tno', nan], dtype=object), array(['yes', 'no', 'yes', '\tno', '\tyes', nan], dtype=object))

```
In [183]: #rectifying the categorical column classes
data['coronary_artery_disease']=data['coronary_artery_disease'].replace('\tno','no')
data['coronary_artery_disease'].unique()

Out[183]: array(['no', 'yes', nan], dtype=object)

In [186]: data['diabetesmellitus']=data.diabetesmellitus.replace(to_replace={'yes':'yes', '\tyes':'yes', '\tno':'no'})
data['diabetesmellitus'].unique()
```

```
Out[186]: array(['yes', 'no', ' yes', nan], dtype=object)

In [187]: #handling missing value
data.isna().sum()

Out[187]: 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        70
white_blood_cell_count    105
red_blood_cell_count      130
hypertension               2
diabetesmellitus           2
coronary_artery_disease    2
appetite                   1
pedal_edema                1
anemia                     1
class                      0
dtype: int64

In [188]: #before handling the numeric variable which is considered as string should be convert to numerical
data.red_blood_cell_count=pd.to_numeric(data.red_blood_cell_count,errors='coerce')
data.packed_cell_volume=pd.to_numeric(data.packed_cell_volume,errors='coerce')
```

```
In [191]: data.isnull().sum()

Out[191]: age                0
blood_pressure            0
specific_gravity          0
albumin                   0
sugar                     0
red_blood_cells          0
pus_cell                  0
pus_cell_clumps           0
bacteria                   0
blood_glucose_random      0
blood_urea                 0
serum_creatinine          0
sodium                    0
potassium                  0
hemoglobin                 0
packed_cell_volume        0
white_blood_cell_count    0
red_blood_cell_count      0
hypertension               0
diabetesmellitus           0
coronary_artery_disease    0
appetite                   0
pedal_edema                0
anemia                     0
class                      0
dtype: int64

In [ ]:

In [192]: from sklearn.preprocessing import LabelEncoder
for i in cat:
    print('label of encoder= ',i)
    lei=LabelEncoder()
```

Projects / CKD prediction / project

```
In [192]: from sklearn.preprocessing import LabelEncoder
for i in cat:
    print('label of encoder= ',i)
    lei=LabelEncoder()
    print(data[i])
    data[i]=lei.fit_transform(data[i])
    print(data[i])
    print('*'*100)
```

```
label of encoder= red_blood_cells
0      normal
1      normal
2      normal
3      normal
4      normal
...
395     normal
396     normal
397     normal
398     normal
399     normal
Name: red_blood_cells, Length: 400, dtype: object
0      1
1      1
2      1
3      1
4      1
..
395     1
396     1
397     1
398     1
399     1
```

Projects / CKD prediction / project

```
Name: red_blood_cells, Length: 400, dtype: int64
*****
label of encoder= pus_cell
0      normal
1      normal
2      normal
3      abnormal
4      normal
...
395     normal
396     normal
397     normal
398     normal
399     normal
Name: pus_cell, Length: 400, dtype: object
0      1
1      1
2      1
3      0
4      1
..
395     1
396     1
397     1
398     1
399     1
Name: pus_cell, Length: 400, dtype: int64
*****
label of encoder= pus_cell_clumps
0      notpresent
1      notpresent
2      notpresent
3      present
4      notpresent
...
395     notpresent
```



Projects / CKD prediction / project

```
Name: pus_cell_clumps, Length: 400, dtype: object
0      0
1      0
2      0
3      1
4      0
..
395    0
396    0
397    0
398    0
399    0
Name: pus_cell_clumps, Length: 400, dtype: int64
*****
label of encoder=  bacteria
0      notpresent
1      notpresent
2      notpresent
3      notpresent
4      notpresent
...
395    notpresent
396    notpresent
397    notpresent
398    notpresent
399    notpresent
Name: bacteria, Length: 400, dtype: object
0      0
1      0
2      0
3      0
4      0
..
395    0
396    0
397    0
```

Projects / CKD prediction / project

```
Name: bacteria, Length: 400, dtype: int64
*****
label of encoder=  hypertension
0      yes
1      no
2      no
3      yes
4      no
...
395    no
396    no
397    no
398    no
399    no
Name: hypertension, Length: 400, dtype: object
0      1
1      0
2      0
3      1
4      0
..
395    0
396    0
397    0
398    0
399    0
Name: hypertension, Length: 400, dtype: int64
*****
label of encoder=  diabetesmellitus
0      yes
1      no
2      yes
3      no
4      no
...
```

Projects / CKD prediction / project

```
397 no
398 no
399 no
Name: diabetesmellitus, Length: 400, dtype: object
0 2
1 1
2 2
3 1
4 1
..
395 1
396 1
397 1
398 1
399 1
Name: diabetesmellitus, Length: 400, dtype: int64
*****
label of encoder= coronary_artery_disease
0 no
1 no
2 no
3 no
4 no
..
395 no
396 no
397 no
398 no
399 no
Name: coronary_artery_disease, Length: 400, dtype: object
0 0
1 0
2 0
3 0
```

Projects / CKD prediction / project

```
399 0
Name: coronary_artery_disease, Length: 400, dtype: int64
*****
label of encoder= appetite
0 good
1 good
2 poor
3 poor
4 good
...
395 good
396 good
397 good
398 good
399 good
Name: appetite, Length: 400, dtype: object
0 0
1 0
2 1
3 1
4 0
..
395 0
396 0
397 0
398 0
399 0
Name: appetite, Length: 400, dtype: int64
*****
label of encoder= pedal_edema
0 no
1 no
2 no
3 yes
4 no
...
```

IBM Cloud Pak for Data

Projects / CKD prediction / project

```
label of encoder= albumin
0 1.0
1 4.0
2 2.0
3 4.0
4 2.0
...
395 0.0
396 0.0
397 0.0
398 0.0
399 0.0
Name: albumin, Length: 400, dtype: float64
0 1
1 4
2 2
3 4
4 2
...
395 0
396 0
397 0
398 0
399 0
Name: albumin, Length: 400, dtype: int64
label of encoder= sugar
0 0.0
1 0.0
2 3.0
3 0.0
4 0.0
...
395 0.0
396 0.0
```

In [193]: data.corr().T

Out[193]:

	age	blood_pressure	specific_gravity	albumin	sugar	red_blood_cells	pus_cell	pus_cell_clumps	bacteria	blood_glucose_random	packed_cell_volume	white_blood_cell_count
age	1.000000	0.146830	-0.161959	0.087907	0.185221	-0.019305	-0.102595	0.154774	0.040733	0.213779	-0.213961	0.118071
blood_pressure	0.146830	1.000000	-0.164422	0.122541	0.190218	-0.151369	-0.156856	0.059560	0.112173	0.149100	-0.286473	0.039911
specific_gravity	-0.161959	-0.164422	1.000000	-0.479962	-0.292053	0.253894	0.365353	-0.306426	-0.231704	-0.317893	0.491157	-0.238711
albumin	0.087907	0.122541	-0.479962	1.000000	0.287751	-0.394844	-0.561713	0.417868	0.377935	0.310481	-0.470819	0.214211
sugar	0.185221	0.190218	-0.292053	0.287751	1.000000	-0.092940	-0.190062	0.168091	0.119399	0.629809	-0.180658	0.161111
red_blood_cells	-0.019305	-0.151369	0.253894	-0.394844	-0.092940	1.000000	0.377394	-0.102948	-0.184402	-0.153076	0.278263	-0.021111
pus_cell	-0.102595	-0.156856	0.365353	-0.561713	-0.190062	0.377394	1.000000	-0.520118	-0.330401	-0.262259	0.417339	-0.107711
pus_cell_clumps	0.154774	0.059560	-0.306426	0.417868	0.168091	-0.102948	-0.520118	1.000000	0.275082	0.197593	-0.297980	0.158211
bacteria	0.040733	0.112173	-0.231704	0.377935	0.119399	-0.184402	-0.330401	0.275082	1.000000	0.085940	-0.188624	0.095711

30°C Mostly s... ENG 16:40

IBM Cloud Pak for Data

Search in your workspaces

Buy

Sowmiya T's Account

Dallas

ST

Projects / CKD prediction / project

age	1.000000	0.146830	-0.161959	0.087907	0.185221	-0.019305	-0.102595	0.154774	0.040733	0.213779	...	-0.213961	0.118007
blood_pressure	0.146830	1.000000	-0.164422	0.122541	0.190218	-0.151369	-0.156856	0.059560	0.112173	0.149100	...	-0.286473	0.039917
specific_gravity	-0.161959	-0.164422	1.000000	-0.479962	-0.292053	0.253894	0.365353	-0.306426	-0.231704	-0.317893	...	0.491157	-0.238717
albumin	0.087907	0.122541	-0.479962	1.000000	0.287751	-0.394844	-0.561713	0.417868	0.377935	0.310481	...	-0.470819	0.214211
sugar	0.185221	0.190218	-0.292053	0.287751	1.000000	-0.092940	-0.190062	0.168091	0.119399	0.629809	...	-0.180658	0.161111
red_blood_cells	-0.019305	-0.151369	0.253894	-0.394844	-0.092940	1.000000	0.377394	-0.102948	-0.184402	-0.153076	...	0.278263	-0.021111
pus_cell	-0.102595	-0.156856	0.365353	-0.561713	-0.190062	0.377394	1.000000	-0.520118	-0.330401	-0.262259	...	0.417339	-0.107717
pus_cell_clumps	0.154774	0.059560	-0.306426	0.417868	0.168091	-0.102948	-0.520118	1.000000	0.275082	0.197593	...	-0.297980	0.158217
bacteria	0.040733	0.112173	-0.231704	0.377935	0.119399	-0.184402	-0.330401	0.275082	1.000000	0.085940	...	-0.188624	0.095717
blood_glucose_random	0.213779	0.149100	-0.317893	0.310481	0.629809	-0.153076	-0.262259	0.197593	0.085940	1.000000	...	-0.259831	0.137111
blood_urea	0.190669	0.183970	-0.249370	0.346935	0.126043	-0.236322	-0.344048	0.184415	0.158444	0.127489	...	-0.517605	0.053611
serum_creatinine	0.128292	0.144359	-0.176146	0.160252	0.094565	-0.138394	-0.157896	0.049940	0.050830	0.082242	...	-0.332099	0.017917
sodium	-0.086552	-0.103220	0.217473	-0.228076	-0.053452	0.140572	0.173323	-0.142135	-0.081733	-0.154393	...	0.342082	-0.001717
potassium	0.049315	0.066648	-0.063324	0.111614	0.180067	0.018192	-0.158750	-0.006316	-0.002688	0.056695	...	-0.116381	-0.062011
hemoglobin	-0.182737	-0.279535	0.492143	-0.474211	-0.156876	0.280990	0.411500	-0.275763	-0.204954	-0.269149	...	0.837611	-0.202111
packed_cell_volume	-0.213961	-0.286473	0.491157	-0.470819	-0.180658	0.278263	0.417339	-0.297980	-0.188624	-0.259831	...	1.000000	-0.193717
white_blood_cell_count	0.118077	0.039957	-0.238735	0.214295	0.161132	-0.021104	-0.107753	0.158224	0.095702	0.137186	...	-0.193722	1.000000
red_blood_cell_count	-0.202451	-0.209598	0.371683	-0.368210	-0.149470	0.166161	0.365479	-0.238772	-0.186593	-0.194327	...	0.642322	-0.094411
hypertension	0.396749	0.270447	-0.323643	0.406057	0.254268	-0.140538	-0.291719	0.195623	0.089046	0.369849	...	-0.565524	0.135511
diabetesmellitus	0.357048	0.231573	-0.352804	0.312978	0.431277	-0.148374	-0.204596	0.167585	0.081995	0.504587	...	-0.443637	0.170311
coronary_artery_disease	0.228872	0.086618	-0.135814	0.200957	0.229301	-0.111493	-0.172295	0.188029	0.162395	0.212433	...	-0.297873	0.012911
appetite	0.154683	0.177601	-0.230975	0.303145	0.069216	-0.160868	-0.274985	0.189688	0.149126	0.176132	...	-0.374184	0.165911
pedal_edema	0.095099	0.054083	-0.253803	0.411080	0.116442	-0.199285	-0.350227	0.104356	0.134732	0.101502	...	-0.382286	0.172611
anemia	0.056215	0.194962	-0.184155	0.229556	0.042464	-0.107625	-0.260566	0.175861	0.052208	0.126854	...	-0.513150	0.043311
class	-0.226120	-0.293477	0.657810	-0.531885	-0.296919	0.284911	0.371574	-0.259008	-0.188370	-0.400775	...	0.654348	-0.291811

25 rows x 25 columns

In [194]

```
selcols=['red_blood_cells','pus_cell',
'diabetesmellitus', 'coronary_artery_disease','blood_urea','pedal_edema','anemia',
'blood_glucose_random']
```

In [195]

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

(400, 8)
(400, 1)
```

Projects / CKD prediction / project

```
(400, 8)
(400, 1)

In [196]: 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,random_state=2)
print(x_train.shape)
print(y_train.shape)
print(x_test.shape)
print(y_test.shape)

(320, 8)
(320, 1)
(80, 8)
(80, 1)

In [202]: from sklearn.linear_model import LogisticRegression
lgr=LogisticRegression()
lgr.fit(x_train,y_train)

Out[202]: LogisticRegression()

In [204]: y_pred=lgr.predict(x_test)
y_pred1=lgr.predict([[140,45,0,0,0,0,0]])
print(y_pred1)

[2]

In [206]: from sklearn.metrics import accuracy_score
acc=accuracy_score(y_test,y_pred)
acc

Out[206]: 0.9125

In [208]: from sklearn.metrics import confusion_matrix
conf_mat=confusion_matrix(y_test,y_pred)
```

Projects / CKD prediction / project

```
[ 0, 0, 20]]

In [209]: !pip install 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: 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: tabulate in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (0.8.9)
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: urllib3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (1.26.7)
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==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: lmond 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-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: cipp>=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)

In [340]: from ibm_watson_machine_learning import APIClient
WML_credentials = {
    "url": "https://us-south.ml.cloud.ibm.com",
    "apikey": "43YHw_v0705kAA7icE3TH_MUakuEou7Bucku1s17a0"
```

Projects / CKD prediction / project

```
Requirement already satisfied: pyyaml==5.4.0 in /opt/conda/envs/python3.9/lib/python3.9/site-packages (from packaging==20.0)
In [340]: from ibm_watson_machine_learning import APIClient
        WML_credentials = {
            "url": "https://us-south.ml.cloud.ibm.com",
            "apikey": "t3XLHV_v0705KAATirEJTN_NUakyEouJPUchv1s17aaQ"
        }
        WML_Client = APIClient(WML_credentials)

In [343]: def guid_from_space_name(WML_Client, space_name):
        space = WML_Client.spaces.get_details()
        #print(space)
        return(next(item for item in space['resources'] if item['entity']['name'] == space_name)['metadata']['id'])

In [344]: space_uid = guid_from_space_name(WML_Client, 'models')
        print("space UID = " + space_uid)

        space UID = 4550afb7-4737-4406-809b-1491c247234f

In [345]: WML_Client.set_default_space(space_uid)

Out[345]: 'SUCCESS'

In [346]: WML_Client.software_specifications.list()

-----
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       09f4cffe-90a7-5899-b9ed-1ef348aebdee base
pytorch-onnx_rt22.1-py3.9        0b848dd4-e681-5599-be41-b5f6fccc6471 base
ai-function_0.1-py3.6            0cdeb01e-5376-4f4d-92dd-da3b69aa9bda base
shiny-r3.6                       0e6e79df-875e-4f24-8ae9-62dc2148306 base
shiny-r3.6                       0e6e79df-875e-4f24-8ae9-62dc2148306 base
```

Projects / CKD prediction / project

```
pytorch-onnx_rt22.1-py3.9        0b848dd4-e681-5599-be41-b5f6fccc6471 base
ai-function_0.1-py3.6            0cdeb01e-5376-4f4d-92dd-da3b69aa9bda base
shiny-r3.6                       0e6e79df-875e-4f24-8ae9-62dc2148306 base
tensorflow_2.4-py3.7-horovod    1092590a-3076-563d-9b62-4eb7964b3f22 base
pytorch_1.1-py3.6              10ac11d6-6b30-4ccd-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-0ab31f6fd3cb 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-9d080b0de37f 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-abd6-19e0c2402f55 base
tensorflow_1.15-py3.6          2b73a275-7c6f-420b-a912-aae7f436e0bc base
kernel-spark3.3-py3.9          2b7961e2-e3b1-5a8c-a491-482c8368839a base
pytorch_1.2-py3.6              2c8ef57d-2687-4b7d-acce-01f9497d6ac1 base
spark-mllib_2.3                 2e51f708-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-ae44-60233c00306e 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-980a-4240ba1ed577 base
pmlml-3.0_4.3                 4393cb95-16f1-5bc5-bee8-81b0af00e5c7 base
spark-mllib_2.4-r_3.6          494030ff-92a9-4c87-a3d7-a42d0021c095 base
xgboost_0.90-py3.6             4ff8d6c2-1343-4c18-85e1-689c965304d3 base
```

Projects / CKD prediction / project

```
autoai-omv_2.0 5c2e37fa-8008-b5e7-840f-d9124b9b14ee base
spss-modeler_18.1 5c3cad7e-507f-4b2a-a9a3-ab53a21dee8b base
cuda-py3.8 5d3232bf-c86b-5d4a-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
Note: Only first 50 records were displayed. To display more use 'limit' parameter.
```

```
In [347]: software_spec_uid = WML_Client.software_specifications.get_uid_by_name("default_py3.7")
software_spec_uid
```

```
Out[347]: 'e4429883-c883-42b6-87a8-f419d64088cd'
```

```
In [348]: MODEL_NAME = 'ckd-B7'
DEPLOYMENT_NAME = 'ckd1-B7'
DEMO_MODEL = 1gr
```

```
In [349]: software_spec_uid = WML_Client.software_specifications.get_id_by_name('runtime-22.1-py3.9')
software_spec_uid
```

```
Out[349]: '12b83a17-24d8-5082-900f-0ab31fbfd3cb'
```

```
In [350]: 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_uid
}
```

```
In [351]: model_details = WML_Client.repository.store_model(
model=DEMO_MODEL,
meta_props=model_props,
training_data=x_train,
training_target=y_train
)
```

Projects / CKD prediction / project

```
In [352]: model_details
```

```
Out[352]: {'entity': {'hybrid_pipeline_software_specs': [],
'label_column': 'class',
'schemas': {'input': [{'fields': [{'name': 'red_blood_cells',
'type': 'int64'},
{'name': 'pus_cell', 'type': 'int64'},
{'name': 'diabetesmellitus', 'type': 'int64'},
{'name': 'coronary_artery_disease', 'type': 'int64'},
{'name': 'blood_urea', 'type': 'float64'},
{'name': 'pedal_edema', 'type': 'int64'},
{'name': 'anemia', 'type': 'int64'},
{'name': 'blood glucose random', 'type': 'float64'}],
'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': '2022-11-17T18:40:17.239Z',
'id': 'da15b054-f93d-42b0-aea7-5fb671faab81',
'modified_at': '2022-11-17T18:40:20.263Z',
'name': 'ckd-B7',
'owner': 'IBMId-665002NKON',
'resource_key': 'b7d606f5-1b66-40af-a137-f2a589ef76db',
'space_id': '4550afb7-4737-4406-809b-1491c247234f',
'system': {'warnings': []}}}
```

```
In [356]: model_id = WML_Client.repository.get_model_id(model_details)
model_id
```

```
Out[356]: 'da15b054-f93d-42b0-aea7-5fb671faab81'
```

```
In [357]: import pickle
pickle.dump(model_details, open('CKD.pkl', 'wb'))
```

IBM Cloud Pak for Data

Search in your workspaces

Buy

Sowmiya T's Account

Dallas

ST

Projects / CKD prediction / project

```
'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': '2022-11-17T18:40:17.239Z',
  'id': 'da15b054-f93d-42b0-aea7-5fb671faab81',
  'modified_at': '2022-11-17T18:40:20.263Z',
  'name': 'ckd-B7',
  'owner': 'IBMId-665002HKON',
  'resource_key': 'b7d686f5-1b66-40af-a137-f2a589ef76db',
  'space_id': '4550afb7-4737-4406-809b-1491c247234f'},
'system': {'warnings': []}}
```

In [356]:  
model\_id = MWL\_Client.repository.get\_model\_id(model\_details)  
model\_id

Out[356]: 'da15b054-f93d-42b0-aea7-5fb671faab81'

In [357]:  
import pickle  
pickle.dump(lgr,open('CKD.pk1','wb'))

In [ ]:

30°C Mostly s...

ENG 1643