

# ALL OUTPUTS:

```
[2] data = pd.read_csv('/content/Thyroid.csv')

data.head()
```

	age	sex	on_thyroxine	query_on_thyroxine	on_antithyroid_meds	sick	pregnant	thyroid_surgery	I131_treatment	query_h
0	29	F	f	f	f	f	f	f	f	f
1	29	F	f	f	f	f	f	f	f	f
2	41	F	f	f	f	f	f	f	f	f
3	36	F	f	f	f	f	f	f	f	f
4	32	F	f	f	f	f	f	f	f	f

5 rows x 31 columns

```
data.isnull().sum()
```

age	0
sex	307
on_thyroxine	0
query_on_thyroxine	0
on_antithyroid_meds	0
sick	0
pregnant	0
thyroid_surgery	0
I131_treatment	0
query_hypothyroid	0
query_hyperthyroid	0
lithium	0
goitre	0
tumor	0
hypopituitary	0
psych	0
TSH_measured	0
TSH	842
T3_measured	0
T3	2604
TT4_measured	0
TT4	442
T4U_measured	0
T4U	809
FTI_measured	0
FTI	882
TBG_measured	0
TBG	8823
referral_source	0
target	0
patient_id	0
dtype: int64	

```
[6] data.drop(['TSH_measured', 'T3_measured', 'TT4_measured', 'T4U_measured', 'FTI_measured', 'TBG_measured', 'referral_source', 'patient_id'], axis=1, inplace=True)
```

```
[7] diagnoses = {'A': 'hyperthyroid conditions',
               'B': 'hyperthyroid condition',
               'C': 'hyperthyroid condition',
               ...}
```

```
[8] data.dropna(subset=['target'], inplace=True)
```

```
[9] data['target'].value_counts()
```

hyperthyroid condition	614
general health	436
binding protein	376
replacement therapy	336
miscellaneous	281
hyperthyroid conditions	147
antithyroid treatment	19
antithyroid treatment	14

Name: target, dtype: int64

```
data[data.age>100]
```

	age	sex	on_thyroxine	query_on_thyroxine	on_antithyroid_meds	sick	pregnant	thyroid_surgery	I131_treatment	query_h
4	32	F	f	f	f	f	f	f	f	f
18	63	F	t	f	f	f	t	f	f	f
32	41	M	f	f	f	f	f	f	f	f
33	71	F	t	f	f	f	f	f	f	f
39	55	F	t	f	f	f	f	f	f	f
...	...	...	...	...	...	...	...	...	...	...
9153	64	M	f	f	f	f	f	f	f	f
9157	60	M	f	f	f	t	f	f	f	f

0 rows x 29 columns

```
[11] x=data.iloc[:,0:-1]
      y=data.iloc[:,1]
```

	age	sex	on_thyroxine	query_on_thyroxine	on_antithyroid_meds	sick	pregnant	thyroid_surgery	I131_treatment	query_h
4	32	F	f	f	f	f	f	f	f	f
18	63	F	t	f	f	f	t	f	f	f
32	41	M	f	f	f	f	f	f	f	f
33	71	F	t	f	f	f	f	f	f	f
39	55	F	t	f	f	f	f	f	f	f
...	...	...	...	...	...	...	...	...	...	...
9153	64	M	f	f	f	f	f	f	f	f
9157	60	M	f	f	f	t	f	f	f	f

```
[13] x['sex'].unique()
array(['F', 'M', nan], dtype=object)

[14] x['sex'].replace(np.nan, 'F', inplace=True)

[15] x['sex'].value_counts()
F    1687
M     536
Name: sex, dtype: int64
```

```
[17] x.info()
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2223 entries, 4 to 9169
Data columns (total 22 columns):
#   Column                Non-Null Count  Dtype
---  -
0   age                    2223 non-null  float64
1   sex                    2223 non-null  object
2   on_thyroxine           2223 non-null  object
3   query_on_thyroxine     2223 non-null  object
4   on_antithyroid_meds    2223 non-null  object
5   sick                   2223 non-null  object
6   pregnant               2223 non-null  object
7   thyroid_surgery        2223 non-null  object
8   I131_treatment         2223 non-null  object
9   query_hypothyroid      2223 non-null  object
```

2223 rows x 22 columns

	age	sex	on_thyroxine	query_on_thyroxine	on_antithyroid_meds	sick	pregnant	thyroid_surgery	I131_treatment	query_hypothyroid	goitre	tumor	hypothyroidism
4	32.0	0	0	0	0	0	0	0	0	0	0	0	0
18	63.0	0	1	0	0	1	0	0	0	0	0	0	0
32	41.0	1	0	0	0	0	0	0	0	0	0	0	0
33	71.0	0	1	0	0	0	0	0	0	0	0	0	0
39	55.0	0	1	0	0	0	0	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
9153	64.0	1	0	0	0	0	0	0	0	0	0	0	0
9157	60.0	1	0	0	0	1	0	0	0	0	0	0	0
9158	64.0	1	0	0	0	0	0	0	0	0	0	0	0
9162	36.0	0	0	0	0	0	0	0	0	0	0	0	0
9169	69.0	1	0	0	0	0	0	0	0	0	0	0	0

```
[43] x=data.iloc[:,0:-1]
      y=data.iloc[:, -1]

[61] x
      age  sex  on_thyroxine  query_on_thyroxine  on_antithyroid_meds  sick  pregnant  thyroid_surgery  I131_treatment  query_hypothyroid  ...  goitre  tumor  hypothyroidism
4      32   F             f             f             f             f             f             f             f             f             ...             f             f
18     63   F             t             f             f             t             f             f             f             f             ...             f             f
32     41   M             f             f             f             f             f             f             f             f             ...             f             f
33     71   F             t             f             f             f             f             f             f             f             ...             f             f
39     55   F             t             f             f             f             f             f             f             t             ...             f             f
...
9153    64   M             f             f             f             f             f             f             f             f             ...             f             f
9157    60   M             f             f             t             t             f             f             f             f             ...             f             f
9158    64   M             f             f             f             f             f             f             f             t             ...             f             f
```

```
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.20, random_state=0)

[87] from imblearn.over_sampling import SMOTE
y_train.value_counts()

target
4      492
3      353
2      295
7      278
6      222
5      109
1       17
0        12
dtype: int64
```

```
[87] 6      222
5      109
1       17
0        12
dtype: int64
```

```
[88] x.replace('F','0',inplace=True)
```

```
[89] x
```

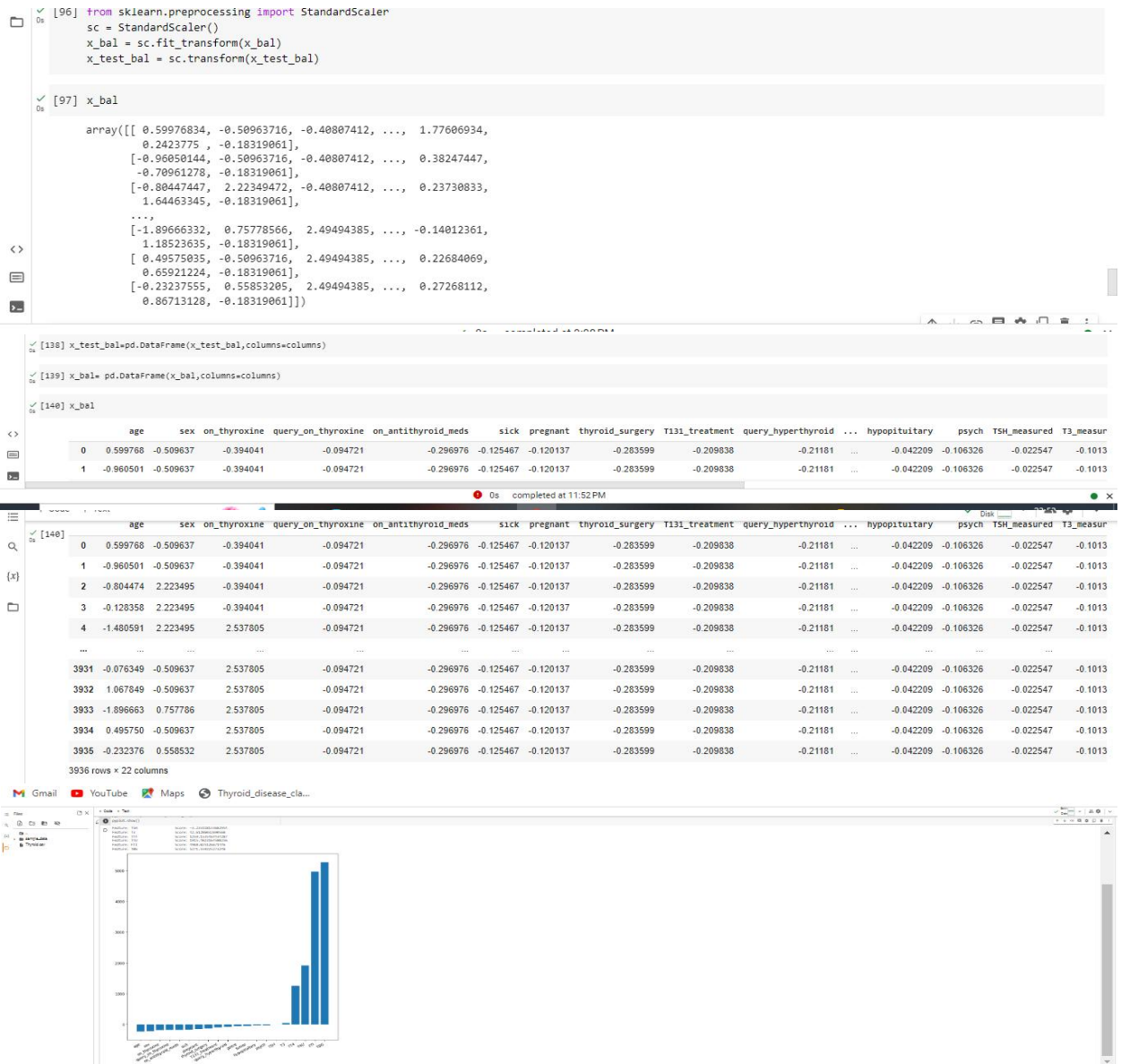
	age	sex	on_thyroxine	query_on_thyroxine	on_antithyroid_meds	sick	pregnant	thyroid_surgery	I131_treatment	query_hypothyroid	...	goitre	tumor	hypo
4	32	0	0	0	0	0	0	0	0	0	0	0	0	0
18	63	0	1	0	0	1	0	0	0	0	0	0	0	0
32	41	1	0	0	0	0	0	0	0	0	0	0	0	0
33	71	0	1	0	0	0	0	0	0	0	0	0	0	0
39	55	0	1	0	0	0	0	0	0	1	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
9153	64	1	0	0	0	0	0	0	0	0	0	0	0	0
9157	60	1	0	0	1	0	0	0	0	0	0	0	0	0

```
[92] target
```

0	6
1	4
2	6
3	2
4	7
...	...
2218	3
2219	3
2220	2
2221	2
2222	2

2223 rows x 1 columns

```
[95] os = SMOTE(random_state=0,k_neighbors=1)
x_bal,y_bal=os.fit_resample(x_train,y_train)
```



thyroid.ipynb - Colaboratory

colab.research.google.com/drive/12ZdF4L\_HBfQ40c34pp1yqa\_5KLYJWUdj#scrollTo=B-inR5Ri\_Q7U

thyroid.ipynb

Files

- sample\_data
- Thyroid.csv

```
[39] [-1.45217796, 0.7754693 ],
      [ 1.19596942, 0.09083978, -1.89809328, ..., 0.28862153,
        -1.1662518 , -0.86229552]]]

[40] columns=['age', 'sex', 'on_thyroxine', 'query_on_thyroxine', 'on_antithyroid_meds', 'sick', 'pregnant', 'thyroid_surgery', 't131_treatment', 'query_hyperthyroid', 'goitre', 'tumor', 'hypopituitary']
x_test_bal=pd.DataFrame(X_test_bal,columns=columns)
x_bal= pd.DataFrame(X_bal,columns=columns)

x_bal.pop('target')
x_bal.pop('patient_id')
x_bal=x_bal.drop(X_bal.loc[:, 'age':'query_hyperthyroid'].columns, axis=1)

x_test_bal.pop('target')
x_test_bal.pop('patient_id')
x_test_bal=x_test_bal.drop(x_test_bal.loc[:, 'age':'query_hyperthyroid'].columns, axis=1)

[41] x_bal.head()
```

	goitre	tumor	hypopituitary	psych	TSH	T3	TT4	T4U	FTI	TBG
0	-0.263344	-0.970383	0.094400	-0.402630	-1.663683	1.220589	0.114703	-1.104442	0.264070	-0.682082
1	-0.251281	-0.360056	-1.402210	0.591212	0.326383	0.532064	-0.577926	0.166417	0.158883	-0.558107
2	-0.361855	-1.281444	1.164418	0.605601	1.721827	0.749161	-0.172085	-0.565012	1.360131	0.040365
3	0.813261	-1.152375	0.779849	1.018223	1.667778	1.641049	2.022576	0.407516	1.843095	-0.071011
4	1.020153	-0.703395	0.278401	0.581238	0.114784	-0.015001	0.359461	1.792613	1.152007	-0.639097

```
[42] x_test_bal.head()
```

	goitre	tumor	hypopituitary	psych	TSH	T3	TT4	T4U	FTI	TBG
--	--------	-------	---------------	-------	-----	----	-----	-----	-----	-----

Executing (49m 29s) <cell line: 1> > fit() > \_run\_search() > evaluate\_candidate() > \_\_call\_\_() > \_\_call\_\_() > retrieve() > wrap\_future\_result() > result() > wait()

35°C Mostly sunny 03:18 23-04-2023

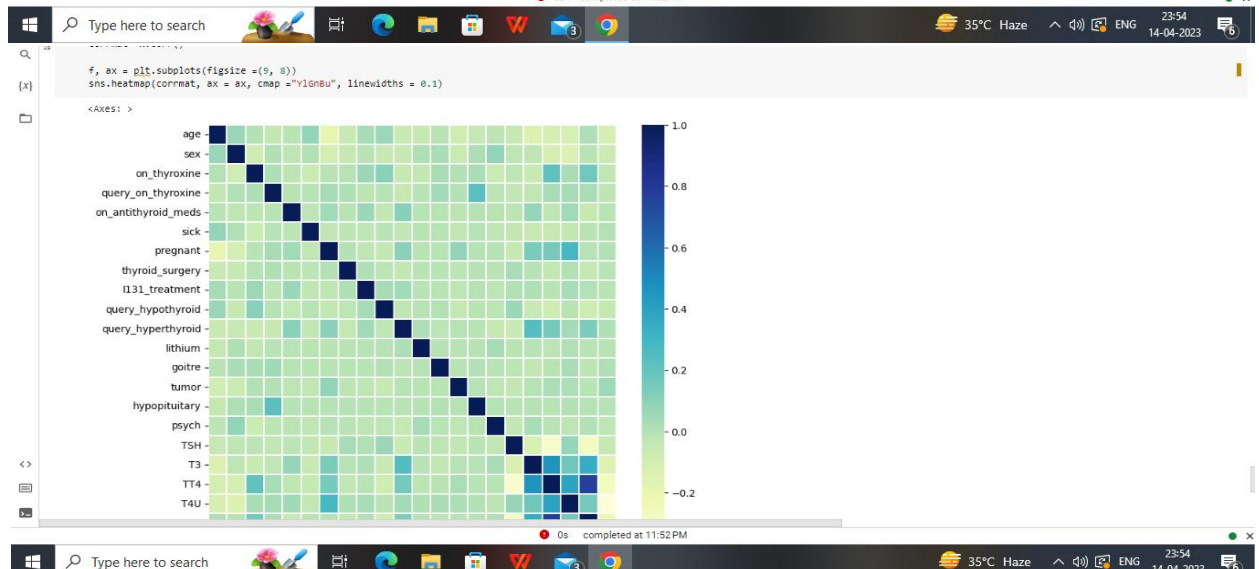
```
thyroid.ipynb - Colaboratory
python - 'numpy.ndarray' object
python - TypeError: corr() missing
colab.research.google.com/drive/1Z2dF4L_HBfQ40c34pp1yqa_5KLYJWUdj#scrollTo=iUzqA1tGq8-Y&uniqifier=2
Gmail YouTube Maps Thyroid_disease_cla...

+ Code + Text
[58] data.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 2223 entries, 4 to 9169
Data columns (total 23 columns):
#   Column                Non-Null Count  Dtype
---  -
0   age                   2223 non-null   int64
1   sex                   2140 non-null   object
2   on_thyroxine          2223 non-null   object
3   query_on_thyroxine    2223 non-null   object
4   on_antithyroid_meds   2223 non-null   object
5   sick                  2223 non-null   object
6   pregnant             2223 non-null   object
7   thyroid_surgery       2223 non-null   object
8   t131_treatment        2223 non-null   object
9   query_hypothyroid     2223 non-null   object
10  query_hyperthyroid    2223 non-null   object
11  lithium               2223 non-null   object
12  goitre                2223 non-null   object
13  tumor                 2223 non-null   object
14  hypopituitary         2223 non-null   object
15  psych                 2223 non-null   object
16  TSH                   2073 non-null   float64
17  T3                    1629 non-null   float64
18  TT4                   2126 non-null   float64
19  T4U                   2045 non-null   float64
20  FTI                   2046 non-null   float64
21  TBG                   96 non-null    float64
22  target               2223 non-null   object
dtypes: float64(6), int64(1), object(16)
memory usage: 416.8+ KB

[58] #checking correlation using Heatmap
import seaborn as sns
corrmat = xx.corr()

f, ax = plt.subplots(figsize=(9, 8))
sns.heatmap(corrmat, ax = ax, cmap = "YlGnBu", linewidths = 0.1)
```





thyroid.ipynb - Collaboratory

colab.research.google.com/drive/1ZZdF4L\_HBfQ40c34pp1yqa\_5KLYJWUdj#scrollTo=B-inRSRi\_Q7U

Files

- sample\_data
- Thyroid.csv

```
[46] rf1 = RandomForestClassifier()
[47] rf1.fit(x,y.values.reshape(-1))
[48] y_pred = rf1.predict(x_test)
[49] y_pred = rf1.predict(x_test)
[50] from sklearn.metrics import classification_report
print(classification_report(y_pred,y_pred))
```

	precision	recall	f1-score	support
0	1.00	1.00	1.00	2
1	1.00	1.00	1.00	2
2	1.00	1.00	1.00	81
3	1.00	1.00	1.00	83
4	1.00	1.00	1.00	122
5	1.00	1.00	1.00	38
6	1.00	1.00	1.00	59
7	1.00	1.00	1.00	58
accuracy			1.00	445
macro avg	1.00	1.00	1.00	445
weighted avg	1.00	1.00	1.00	445

```
[51] from sklearn.metrics import accuracy_score
train_score = accuracy_score(y, rf1.predict(x))
train_score
```

Executing (53m 44s) <cell line: 1> > fit() > \_run\_search() > evaluate\_candidate() > \_\_call\_\_() > \_\_call\_\_() > retrieve() > wrap\_future\_result() > result() > wait()

Type here to search

35°C Partly sunny 03:22 23-04-2023

```
[219] from xgboost import XGBClassifier
xgb1 = XGBClassifier()
xgb1.fit(x,y)
[220] y_pred = xgb1.predict(x_test)
[236] print(classification_report(y_test,y_pred))
```

	precision	recall	f1-score	support
antithyroid treatment	0.09	1.00	0.16	2
antithyroid treatment	0.02	0.50	0.04	2
binding protein	0.08	0.75	0.78	81
general health	0.71	0.75	0.73	83
hyperthyroid condition	0.87	0.49	0.63	122
hyperthyroid conditions	0.52	0.71	0.60	38
miscellaneous	0.89	0.53	0.66	59
replacement therapy	0.35	0.34	0.35	58
accuracy			0.59	445
macro avg	0.53	0.63	0.49	445

0s completed at 2:05 AM

```
[228] sv.fit(x_bal,y_bal)
[230] x_test_bal
[233] y_pred = sv.predict(x_test)
[234] print(classification_report(y_test,y_pred))
```

	precision	recall	f1-score	support
antithyroid treatment	0.09	1.00	0.16	2
antithyroid treatment	0.02	0.50	0.04	2
binding protein	0.08	0.75	0.78	81
general health	0.71	0.75	0.73	83
hyperthyroid condition	0.87	0.49	0.63	122
hyperthyroid conditions	0.52	0.71	0.60	38
miscellaneous	0.89	0.53	0.66	59
replacement therapy	0.35	0.34	0.35	58
accuracy			0.59	445
macro avg	0.53	0.63	0.49	445
weighted avg	0.73	0.59	0.63	445

0s completed at 2:05 AM

Type here to search

36°C Mostly cloudy 02:06 15-04-2023

```
Total params: 100,481
Trainable params: 100,481
Non-trainable params: 0
```

```
Epoch 1/15
32/32 [=====] - 2s 15ms/step - loss: 104.3973 - accuracy: 0.0000e+00 - val_loss: 15.5883 - val_accuracy: 0.1250
Epoch 2/15
32/32 [=====] - 0s 8ms/step - loss: 8186.0161 - accuracy: 0.0000e+00 - val_loss: 515.8256 - val_accuracy: 0.1250
Epoch 3/15
32/32 [=====] - 0s 8ms/step - loss: 140062.1250 - accuracy: 0.0000e+00 - val_loss: 5200.1826 - val_accuracy: 0.1250
Epoch 4/15
32/32 [=====] - loss: 1188986.1250 - accuracy: 0.0000e+00 - val_loss: 30561.3047 - val_accuracy: 0.1250
Epoch 5/15
32/32 [=====] - 0s 10ms/step - loss: 5914651.0000 - accuracy: 0.0000e+00 - val_loss: 176951.7188 - val_accuracy: 0.1250
Epoch 6/15
32/32 [=====] - 0s 8ms/step - loss: 26695312.0000 - accuracy: 0.0000e+00 - val_loss: 493499.5312 - val_accuracy: 0.1250
Epoch 7/15
32/32 [=====] - 0s 8ms/step - loss: 58659834.0000 - accuracy: 0.0000e+00 - val_loss: 1344703.5000 - val_accuracy: 0.1250
Epoch 8/15
32/32 [=====] - 0s 8ms/step - loss: 136603872.0000 - accuracy: 0.0000e+00 - val_loss: 3209238.2500 - val_accuracy: 0.1250
Epoch 9/15
32/32 [=====] - 0s 8ms/step - loss: 38232348.0000 - accuracy: 0.0000e+00 - val_loss: 645806.0000 - val_accuracy: 0.1250
Epoch 10/15
32/32 [=====] - 0s 8ms/step - loss: 667415560.0000 - accuracy: 0.0000e+00 - val_loss: 13070441.0000 - val_accuracy: 0.1250
Epoch 11/15
32/32 [=====] - 0s 5ms/step - loss: 112643696.0000 - accuracy: 0.0000e+00 - val_loss: 2327458.0000 - val_accuracy: 0.1250
Epoch 12/15
32/32 [=====] - 0s 5ms/step - loss: 194724940.0000 - accuracy: 0.0000e+00 - val_loss: 3896190.0000 - val_accuracy: 0.1250
Epoch 13/15
32/32 [=====] - 0s 5ms/step - loss: 320643584.0000 - accuracy: 0.0000e+00 - val_loss: 62535448.0000 - val_accuracy: 0.1250
Epoch 14/15
32/32 [=====] - 0s 5ms/step - loss: 5507363552.0000 - accuracy: 0.0000e+00 - val_loss: 94893600.0000 - val_accuracy: 0.1250
Epoch 15/15
32/32 [=====] - 0s 5ms/step - loss: 7505127600.0000 - accuracy: 0.0000e+00 - val_loss: 142974304.0000 - val_accuracy: 0.1250
clear_callbacks(history at 0x27f640340)
```

```
[73] print(classification_report(y_pred,y_pred))
```

Executing (59m 11s) <cell line 1> > fit() > run\_search() > evaluate\_candidates() > \_call\_\_() > \_call\_\_() > retrieve() > wrap\_future\_result() > result() > wait()



thyroid.ipynb - Colaboratory

colab.research.google.com/drive/12ZdF4L\_HBfQ40c34pp1yqa\_5KLYJWUdj#scrollTo=B-inR5Ri\_Q7U

thyroid.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Files

- sample\_data
- Thyroid.csv

```
[73] print(classification_report(y_pred,y_pred))
```

	precision	recall	f1-score	support
2	1.00	1.00	1.00	73
3	1.00	1.00	1.00	100
4	1.00	1.00	1.00	140
5	1.00	1.00	1.00	20
6	1.00	1.00	1.00	27
7	1.00	1.00	1.00	85
accuracy			1.00	445
macro avg	1.00	1.00	1.00	445
weighted avg	1.00	1.00	1.00	445

```
[74] train_score = accuracy_score(y_rfml.predict(x))
```

```
[75] train_score
```

```
1.0
```

```
print(classification_report(y_pred,y_pred))
```

	precision	recall	f1-score	support
2	1.00	1.00	1.00	73
3	1.00	1.00	1.00	100
4	1.00	1.00	1.00	140
5	1.00	1.00	1.00	20
6	1.00	1.00	1.00	27
7	1.00	1.00	1.00	85
accuracy			1.00	445
macro avg	1.00	1.00	1.00	445
weighted avg	1.00	1.00	1.00	445

```
[79] train_score = accuracy_score(y_bal, xgb.predict(x_bal))
```

```
train_score
```

Executing (1h 1m 46s) <cell line: 1> > fit() > \_run\_search() > evaluate\_candidates() > \_call\_\_() > \_call\_\_() > retrieve() > wrap\_future\_result() > result() > wait()

Type here to search

35°C Partly sunny

03:30  
23-04-2023

thyroid.ipynb - Colaboratory

colab.research.google.com/drive/12ZdF4L\_HBfQ40c34pp1yqa\_5KLYjWUdj#scrollTo=TRJ\_UI-seNgJ

thyroid.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Files

- sample\_data
- Thyroid.csv

```
[80] y_pred = sv.predict(x_test)

[81] print(classification_report(y_pred,y_pred))

              precision    recall  f1-score   support

     2         1.00        1.00        1.00        73
     3         1.00        1.00        1.00       100
     4         1.00        1.00        1.00       140
     5         1.00        1.00        1.00        20
     6         1.00        1.00        1.00        27
     7         1.00        1.00        1.00        85

 accuracy          1.00          1.00          1.00       445
 macro avg          1.00          1.00          1.00       445
 weighted avg          1.00          1.00          1.00       445

[82] train_score=accuracy_score(y,sv.predict(x))
train_score

0.6923076923076923

[83] y_pred = model.predict(x_test_bal)

31/31 [WARNING:tensorflow] - 8s 44s/step
/usr/local/lib/python3.9/dist-packages/tensorflow/python/data/ops/structured_function.py:254: UserWarning: Even though the 'tf.config.experimental_run_functions_eagerly' option is set to True, this function is not configured to run eagerly.
warnings.warn(
/usr/local/lib/python3.9/dist-packages/tensorflow/python/data/ops/structured_function.py:254: UserWarning: Even though the 'tf.config.experimental_run_functions_eagerly' option is set to True, this function is not configured to run eagerly.
warnings.warn(

[84] print(classification_report(y_pred,y_pred))

              precision    recall  f1-score   support

     2         1.00        1.00        1.00        73
     3         1.00        1.00        1.00       100
     4         1.00        1.00        1.00       140
     5         1.00        1.00        1.00        20
     6         1.00        1.00        1.00        27
     7         1.00        1.00        1.00        85

 accuracy          1.00          1.00          1.00       445
 macro avg          1.00          1.00          1.00       445
 weighted avg          1.00          1.00          1.00       445
```

Executing (1h 3m 26s) <cell line: 1> > fit() > \_run\_search() > evaluate\_candidates() > \_call\_\_() > \_call\_\_() > retrieve() > wrap\_future\_result() > result() > wait()

thyroid.ipynb - Colaboratory

colab.research.google.com/drive/12ZdF4L\_HBfQ40c34pp1yqa\_5KLYjWUdj#scrollTo=TRJ\_UI-seNgJ

thyroid.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Files

- sample\_data
- Thyroid.csv

```
[83] print(classification_report(y_pred,y_pred))

              precision    recall  f1-score   support

     2         1.00        1.00        1.00        73
     3         1.00        1.00        1.00       100
     4         1.00        1.00        1.00       140
     5         1.00        1.00        1.00        20
     6         1.00        1.00        1.00        27
     7         1.00        1.00        1.00        85

 accuracy          1.00          1.00          1.00       445
 macro avg          1.00          1.00          1.00       445
 weighted avg          1.00          1.00          1.00       445

[84] print(classification_report(y_pred,y_pred))

              precision    recall  f1-score   support

     2         1.00        1.00        1.00        73
     3         1.00        1.00        1.00       100
     4         1.00        1.00        1.00       140
     5         1.00        1.00        1.00        20
     6         1.00        1.00        1.00        27
     7         1.00        1.00        1.00        85

 accuracy          1.00          1.00          1.00       445
 macro avg          1.00          1.00          1.00       445
 weighted avg          1.00          1.00          1.00       445

[85] accuracy_score(y_test_bal,y_pred)

0.0

[117] from sklearn.model_selection import RandomizedSearchCV
      params = {

          'C': [0.1,1,10,100,1000],
          'gamma': [1,0.1,0.01,0.001,0.0001],
          'kernel': ['rbf','sqr','linear','poly','precomputed','sigmoid']

      }

[118] random_svc =RandomizedSearchCV(sv,params, scoring='accuracy',cv=5,n_jobs=-1)
```

Executing (1h 5m 18s) <cell line: 1> > fit() > \_run\_search() > evaluate\_candidates() > \_call\_\_() > \_call\_\_() > retrieve() > wrap\_future\_result() > result() > wait()

thyroid.ipynb - Colaboratory

colab.research.google.com/drive/12ZdF4l\_HBfQ40c34pp1yqa\_SKLYJWUdj#scrollTo=A22GRJY1-c7G

File Edit View Insert Runtime Tools Help All changes saved

Files

- sample\_data
- Thyroid.csv
- thyroid\_1\_model.pkl

```
[205] random_svc = RandomizedSearchCV(svc, params, scoring='accuracy', cv=5, n_jobs=-1)
[206] svc = SVC(kernel='rbf', gamma=0.1,)
[210] svc.fit(X,y)
/usr/local/lib/python3.9/dist-packages/sklearn/utils/validation.py:1143: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y = column or 1d(y, warn=True)
> SVC
SVC(gamma=0.1)
[212] y_pred = svc.predict(x)
[213] print(classification_report(y_pred, y_pred))
```

	precision	recall	f1-score	support
0	1.00	1.00	1.00	13
1	1.00	1.00	1.00	19
2	1.00	1.00	1.00	376
3	1.00	1.00	1.00	436
4	1.00	1.00	1.00	616
5	1.00	1.00	1.00	146
6	1.00	1.00	1.00	282
7	1.00	1.00	1.00	325
accuracy	1.00	1.00	1.00	2223
macro avg	1.00	1.00	1.00	2223
weighted avg	1.00	1.00	1.00	2223

```
[215] train_score = accuracy_score(y, svc.predict(x))
```

completed at 5:39 AM

Type here to search

sample\_data

Thyroid.csv

```
precision recall f1-score support
2 1.00 1.00 1.00 73
3 1.00 1.00 1.00 100
4 1.00 1.00 1.00 140
5 1.00 1.00 1.00 20
6 1.00 1.00 1.00 27
7 1.00 1.00 1.00 85
accuracy 1.00 1.00 1.00 445
macro avg 1.00 1.00 1.00 445
weighted avg 1.00 1.00 1.00 445
```

```
[82] train_score = accuracy_score(y, svc.predict(x))
train_score
0.6923076923076923
```

thyroid.ipynb - Colaboratory

colab.research.google.com/drive/12ZdF4l\_HBfQ40c34pp1yqa\_SKLYJWUdj#scrollTo=SRZL6YRU6cj

File Edit View Insert Runtime Tools Help All changes saved

Files

- sample\_data
- Thyroid.csv
- thyroid\_1\_model.pkl

```
[213]
0 1.00 1.00 1.00 282
7 1.00 1.00 1.00 335
accuracy 1.00 1.00 1.00 2223
macro avg 1.00 1.00 1.00 2223
weighted avg 1.00 1.00 1.00 2223
```

```
[215] train_score = accuracy_score(y, svc.predict(x))
train_score
0.996504723346828
```

```
[216] import pickle
pickle.dump(svc, open('thyroid_1_model.pkl', 'wb'))
```

```
[218] features = np.array([[0,0,0,0,0.000000,0,0,0,1.00,0,0,0,0]])
```

```
[219] pickle.dump(label_encoder, open('label_encoder.pkl', 'wb'))
```

```
data['target'].unique()
array(['miscellaneous', 'hyperthyroid condition', 'binding protein',
'replacement therapy', 'general health', 'hyperthyroid conditions',
'antithyroid treatment', 'antithyroid treatment'], dtype=object)
```

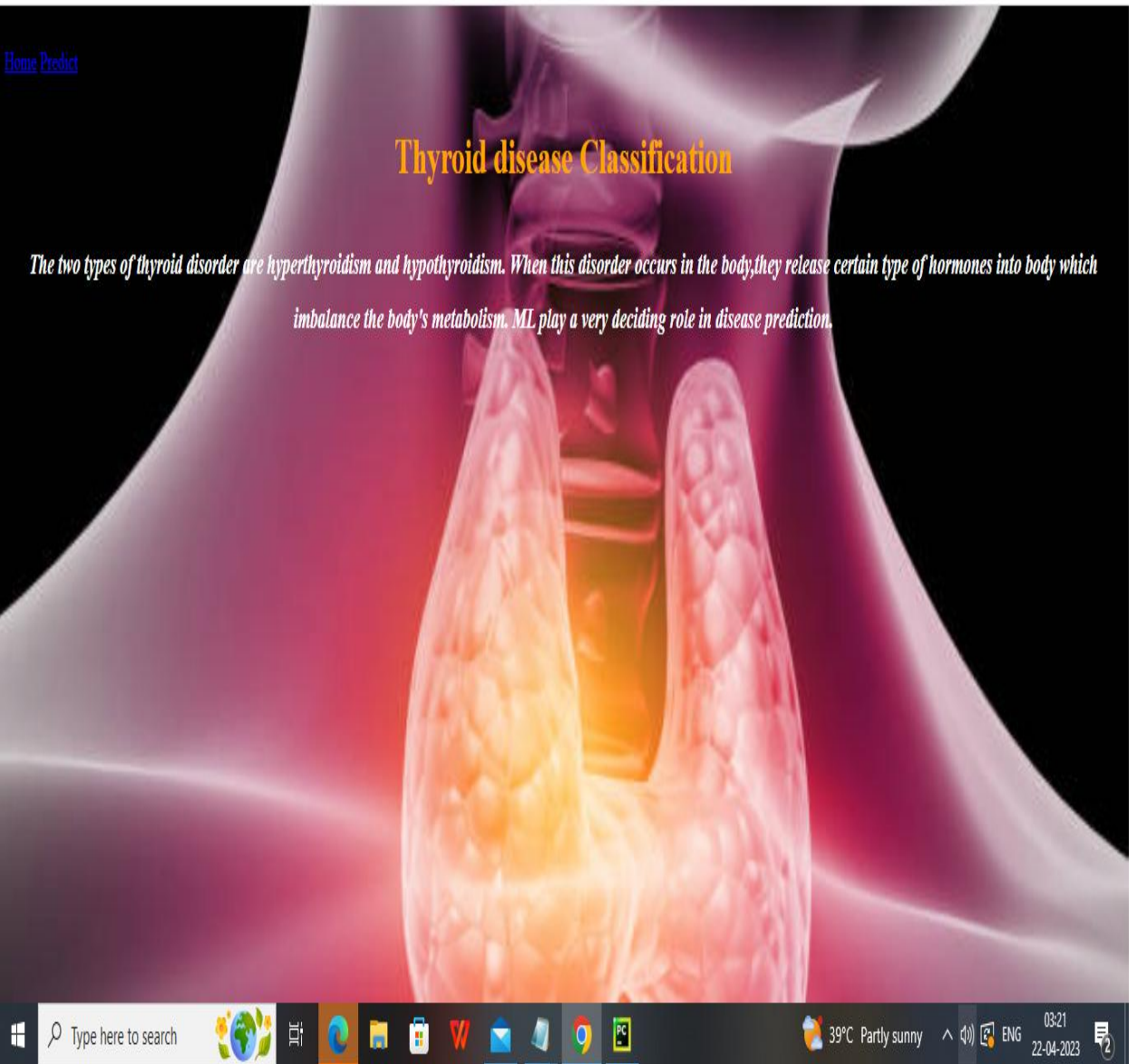
```
y['target'].unique()
array([6, 4, 2, 7, 3, 5, 1, 0])
```

completed at 5:39 AM

[Home](#) [Predict](#)

## Thyroid disease Classification

*The two types of thyroid disorder are hyperthyroidism and hypothyroidism. When this disorder occurs in the body, they release certain type of hormones into body which imbalance the body's metabolism. ML play a very deciding role in disease prediction.*



Windows taskbar: Type here to search, 39°C Partly sunny, 03:21, 22-04-2023

Submit



thyroid.ipynb - Colaboratory x Home x +

localhost:63342/pythonProject/thyroid/templates/submit.html?\_ijt=kiauKn04614ar06u8f959ovj82&\_ij\_reload=RELOAD\_ON\_SAVE

Gmail YouTube Maps Thyroid\_disease\_cla...

[Home](#) [Predict](#)

## Thyroid disease Classification

Based on given input, it predicts Thyroid disease for your body condition is ['miscellaneous']

Type here to search

35°C Mostly sunny 02:16 23-04-2023