## RANDOM\_FOREST

June 12, 2023

```
#DATE:-9-6-2023 ____**MOBILE _PRICE _CLASSIFICATION, using - RANDOM FOR-
    EST**
[]: import pandas as pd
     import numpy as np
     import matplotlib.pyplot as plt ,seaborn as sns
[]: train_df=pd.read_csv(r"/content/Mobile_Price_Classification_train.csv")
     train df
[]:
                                 clock_speed dual_sim
           battery_power
                           blue
                                                          fc
                                                              four_g int_memory
                      842
                              0
                                          2.2
                                                       0
                                                           1
                                                                   0
                                                                                7
     1
                     1021
                              1
                                          0.5
                                                           0
                                                                    1
                                                                               53
                                                       1
     2
                                                           2
                      563
                              1
                                          0.5
                                                       1
                                                                    1
                                                                               41
     3
                      615
                              1
                                          2.5
                                                           0
                                                                    0
                                                                               10
     4
                     1821
                              1
                                          1.2
                                                       0
                                                          13
                                                                    1
                                                                               44
                                            •••
     1995
                      794
                              1
                                          0.5
                                                       1
                                                           0
                                                                    1
                                                                                2
     1996
                     1965
                              1
                                          2.6
                                                       1
                                                           0
                                                                   0
                                                                               39
     1997
                              0
                                          0.9
                                                           1
                                                                               36
                     1911
                                                       1
                                                                    1
                                                           4
     1998
                     1512
                              0
                                          0.9
                                                                               46
                                          2.0
     1999
                      510
                                                           5
                                                                    1
                                                                               45
                              1
                                       ... px_height
           m_{dep}
                  mobile_wt n_cores
                                                      px_width
                                                                  ram
                                                                        sc_h
                                                                              SC_W
     0
             0.6
                         188
                                     2
                                                  20
                                                            756 2549
                                                                           9
                                                                                 7
     1
             0.7
                         136
                                     3
                                                 905
                                                           1988
                                                                 2631
                                                                          17
                                                                                 3
     2
             0.9
                         145
                                     5
                                                           1716 2603
                                                                          11
                                                                                 2
                                                1263
     3
             0.8
                         131
                                     6
                                                1216
                                                           1786 2769
                                                                          16
                                                                                 8
                                                                                 2
     4
             0.6
                                     2
                                                           1212 1411
                                                                          8
                         141
                                                1208
     1995
             0.8
                         106
                                     6
                                                1222
                                                           1890
                                                                  668
                                                                          13
                                                                                 4
     1996
             0.2
                         187
                                     4
                                                 915
                                                           1965 2032
                                                                          11
                                                                                10
     1997
             0.7
                         108
                                     8
                                                 868
                                                           1632 3057
                                                                           9
                                                                                 1
     1998
             0.1
                         145
                                     5
                                                  336
                                                            670
                                                                  869
                                                                          18
                                                                                10
     1999
             0.9
                         168
                                     6
                                                  483
                                                            754 3919
                                                                          19
                                                                                 4
                       three_g touch_screen wifi price_range
           talk_time
     0
                   19
                             0
                                            0
                                                   1
```

1	7	1	1	0	2
2	9	1	1	0	2
3	11	1	0	0	2
4	15	1	1	0	1
•••		<b></b>		•••	
1995	19	1	1	0	0
1996	16	1	1	1	2
1997	5	1	1	0	3
1998	19	1	1	1	0
1999	2	1	1	1	3

[2000 rows x 21 columns]

```
[ ]: test_df=pd.read_csv(r"/content/Mobile_Price_Classification_test.csv")
    test_df
```

[]:		id	batte	ery_pow	ær	blu	ıe	clock_s	peed	dual_	sim	fc	fou	r_g	int_m	emory	\
	0	1			)43		1		1.8	_	1	14		0	_	5	
	1	2		8	341		1		0.5		1	4		1		61	
	2	3		18	307		1		2.8		0	1		0		27	
	3	4		15	546		0		0.5		1	18		1		25	
	4	5		14	134		0		1.4		0	11		1		49	
		•••		•••				•••	•••								
	995	996		17	700		1		1.9		0	0		1		54	
	996	997		6	309		0		1.8		1	0		0		13	
	997	998		11	L85		0		1.4		0	1		1		8	
	998	999		15	533		1		0.5		1	0		0		50	
	999	1000		12	270		1		0.5		0	4		1		35	
		${\tt m\_dep}$	mobi	lle_wt	•••	рc	рэ	_height	px_	width	ran	n s	c_h	sc_w	<i>i</i> \		
	0	0.1		193	•••	16		226		1412	3476	6	12	7	7		
	1	0.8		191	•••	12		746		857	389	5	6	C	)		
	2	0.9		186	•••	4		1270		1366	2396	6	17	10	)		
	3	0.5		96	•••	20		295		1752	3893	3	10	C	)		
	4	0.5		108	•••	18		749		810	1773	3	15	8	}		
		•••						•••	•••		•••						
	995	0.5		170	•••	17		644		913	212:	1	14	8	3		
	996	0.9		186	•••	2		1152		1632	1933	3	8	1	L		
	997	0.5		80	•••	12		477		825	1223		5	C	)		
	998	0.4		171	•••	12		38		832	2509	9	15	11			
	999	0.1		140	•••	19		457		608	2828	3	9	2	2		
	_	talk_		three_	_	touc	h_s	screen									
	0		2		0			1	0								
	1		7		1			0	0								

	talk_time	three_g	touch_screen	Wifi
0	2	0	1	0
1	7	1	0	0
2	10	0	1	1
3	7	1	1	0

4	7	1		0	1
		•••	•••	•••	
995	15	1		1	0
996	19	0		1	1
997	14	1		0	0
998	6	0		1	0
999	3	1		0	1

[1000 rows x 21 columns]

## []: train\_df.head()

[]:	battery_power	blue	clock_speed	dual_sim	fc	four_g	<pre>int_memory</pre>	m_dep \	
0	842	2 0	2.2	0	1	0	7	0.6	
1	102:	l 1	0.5	1	0	1	53	0.7	
2	563	3 1	0.5	1	2	1	41	0.9	
3	619	5 1	2.5	0	0	0	10	0.8	
4	182:	l 1	1.2	0	13	1	44	0.6	
	mobile_wt n	_cores	px_height	px_width	ram	n sc_h	sc_w talk_	_time \	
0	188	2	20	756	2549	9	7	19	
	400	_	005	4000	0004	4	_	_	

U	100	2	•••	20	150	2549	9	1	19
1	136	3		905	1988	2631	17	3	7
2	145	5		1263	1716	2603	11	2	9
3	131	6		1216	1786	2769	16	8	11
4	141	2		1208	1212	1411	8	2	15

three\_g touch\_screen wifi price\_range 

[5 rows x 21 columns]

## []: test\_df.head()

[]: id battery\_power blue clock\_speed dual\_sim fc four\_g int\_memory \ 1.8 0.5 2.8 0.5 1 18 1.4 0 11 

```
m_dep mobile_wt ... pc px_height px_width
                                                 ram
                                                      sc_h sc_w \
0
    0.1
               193 ...
                       16
                                 226
                                          1412
                                                3476
                                                        12
                                                               7
1
    0.8
               191 ... 12
                                 746
                                           857
                                                3895
                                                         6
                                                               0
```

```
2
     0.9
                  186
                             4
                                      1270
                                                  1366
                                                         2396
                                                                  17
                                                                         10
3
     0.5
                   96
                                                  1752
                                                                          0
                            20
                                       295
                                                         3893
                                                                   10
4
     0.5
                  108
                            18
                                       749
                                                   810
                                                         1773
                                                                  15
                                                                           8
                          touch_screen
   talk_time
                three_g
                                           wifi
0
            2
                       0
                                       1
                                              0
            7
                                       0
                                              0
1
                       1
2
           10
                       0
                                       1
                                              1
            7
3
                       1
                                       1
                                              0
4
            7
                       1
                                       0
                                              1
```

[5 rows x 21 columns]

```
[]: train_df.shape
```

[]: (2000, 21)

```
[]: test_df.shape
```

[]: (1000, 21)

## []: train\_df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2000 entries, 0 to 1999
Data columns (total 21 columns):

Non-Null Count Column Dtype \_\_\_\_\_ 0 battery\_power 2000 non-null int64 1 2000 non-null int64 blue 2 2000 non-null float64 clock\_speed 3 2000 non-null int64 dual\_sim 4 2000 non-null int64 fc 5 four\_g 2000 non-null int64 6 int\_memory 2000 non-null int64 7  $m_{dep}$ 2000 non-null float64 8 mobile\_wt 2000 non-null int64 9 n\_cores 2000 non-null int64 10 рс 2000 non-null int64 2000 non-null int64 px\_height 12 px\_width 2000 non-null int64 13 ram2000 non-null int64 2000 non-null int64 14  $sc_h$ sc\_w 2000 non-null int64 15 2000 non-null 16 talk\_time int6417 2000 non-null int64 three\_g 18 touch\_screen 2000 non-null int64

```
20 price_range
    dtypes: float64(2), int64(19)
    memory usage: 328.2 KB
[]: test_df.info()
    <class 'pandas.core.frame.DataFrame'>
    RangeIndex: 1000 entries, 0 to 999
    Data columns (total 21 columns):
         Column
                         Non-Null Count
                                         Dtype
     0
         id
                         1000 non-null
                                         int64
     1
                         1000 non-null
         battery_power
                                         int64
     2
                         1000 non-null
                                         int64
         blue
     3
         clock_speed
                         1000 non-null
                                         float64
     4
         {\tt dual\_sim}
                         1000 non-null
                                         int64
     5
         fc
                         1000 non-null
                                         int64
     6
                         1000 non-null
                                         int64
         four_g
     7
         int_memory
                         1000 non-null
                                         int64
         m_dep
                         1000 non-null
                                         float64
         mobile_wt
                         1000 non-null
                                         int64
     10
         n_cores
                         1000 non-null
                                         int64
                         1000 non-null
                                         int64
     11
         рс
     12
         px_height
                         1000 non-null
                                         int64
     13 px_width
                         1000 non-null
                                         int64
     14
                         1000 non-null
        ram
                                         int64
     15
                         1000 non-null
                                         int64
         sc_h
                         1000 non-null
         sc_w
                                         int64
     17 talk_time
                         1000 non-null
                                         int64
     18
        three_g
                         1000 non-null
                                         int64
     19
        touch_screen
                         1000 non-null
                                         int64
     20 wifi
                         1000 non-null
                                         int64
    dtypes: float64(2), int64(19)
    memory usage: 164.2 KB
[]: x=train_df.drop('wifi',axis=1)
     y=train_df['wifi']
[ ]: x=test_df.drop('wifi',axis=1)
     y=test_df['wifi']
[]: train_df['dual_sim'].value_counts()
[]:1
          1019
           981
     0
     Name: dual_sim, dtype: int64
```

int64

int64

19 wifi

2000 non-null

2000 non-null

```
[]: test_df['dual_sim'].value_counts()
[]:1
           517
     0
           483
     Name: dual_sim, dtype: int64
[]: m={"three g":{"yes":1,"No":0}}
     train_df=train_df.replace(m)
     print(train_df)
                                  clock_speed dual_sim fc
           battery_power blue
                                                                four_g
                                                                         int memory \
    0
                      842
                               0
                                            2.2
                                                         0
                                                             1
                                                                       0
                                                                                    7
                     1021
                                            0.5
                                                             0
    1
                               1
                                                         1
                                                                       1
                                                                                   53
    2
                                            0.5
                                                             2
                      563
                               1
                                                         1
                                                                       1
                                                                                   41
    3
                      615
                               1
                                            2.5
                                                         0
                                                             0
                                                                       0
                                                                                   10
    4
                                            1.2
                                                         0
                                                            13
                                                                                   44
                     1821
                               1
                                                                       1
                                              •••
                                            0.5
                                                                                    2
    1995
                      794
                               1
                                                         1
                                                             0
                                                                       1
                                            2.6
    1996
                     1965
                                                         1
                                                             0
                                                                                   39
                               1
                                                                       0
    1997
                     1911
                               0
                                            0.9
                                                         1
                                                             1
                                                                       1
                                                                                   36
    1998
                     1512
                               0
                                            0.9
                                                         0
                                                             4
                                                                                   46
                                                                       1
                                                                       1
    1999
                      510
                               1
                                            2.0
                                                         1
                                                             5
                                                                                   45
           m dep
                   mobile_wt n_cores
                                         ... px_height
                                                         px_width
                                                                          sc h
                                                                     ram
                                                                                  sc_w
                          188
    0
             0.6
                                      2
                                                    20
                                                               756
                                                                    2549
                                                                              9
                                                                                     7
                                         ...
             0.7
                                      3
                                                   905
                                                                                     3
    1
                          136
                                                              1988
                                                                    2631
                                                                             17
    2
             0.9
                          145
                                      5
                                                  1263
                                                              1716
                                                                    2603
                                                                                     2
                                                                             11
    3
             0.8
                          131
                                      6
                                                  1216
                                                              1786
                                                                    2769
                                                                             16
                                                                                     8
    4
                                                                                     2
             0.6
                          141
                                      2
                                                  1208
                                                              1212
                                                                    1411
                                                                              8
    1995
             0.8
                          106
                                      6
                                                  1222
                                                              1890
                                                                     668
                                                                             13
                                                                                     4
    1996
             0.2
                          187
                                      4
                                                   915
                                                              1965
                                                                    2032
                                                                                    10
                                                                             11
             0.7
                                      8
                                                                              9
    1997
                          108
                                                   868
                                                              1632
                                                                    3057
                                                                                     1
                                      5
    1998
             0.1
                          145
                                                   336
                                                               670
                                                                     869
                                                                             18
                                                                                    10
    1999
             0.9
                          168
                                      6
                                                   483
                                                              754
                                                                    3919
                                                                             19
                                                                                     4
           talk_time
                       three_g
                                 touch_screen wifi
                                                       price_range
    0
                   19
                              0
                                              0
                                                     1
                                                                   1
    1
                    7
                              1
                                                    0
                                                                   2
                                              1
                    9
                                                                   2
    2
                              1
                                              1
                                                    0
    3
                   11
                              1
                                              0
                                                    0
                                                                   2
    4
                              1
                                                    0
                   15
                                              1
                                                                   1
    1995
                   19
                              1
                                              1
                                                    0
                                                                   0
    1996
                   16
                              1
                                              1
                                                                   2
                                                    1
                    5
    1997
                              1
                                              1
                                                    0
                                                                   3
                                                                   0
    1998
                   19
                              1
                                              1
                                                     1
```

1999 2 1 1 1 3

[2000 rows x 21 columns]

```
[]: m={"three_g":{"yes":1,"No":0}}
test_df=test_df.replace(m)
print(test_df)
```

	id	batte	ry_pow		blu	e clock_sp	-	dual_	sim		our_g	int_m	nemory	١
0	1		10	)43		1	1.8		1	14	0		5	
1	2		8	341		1	0.5		1	4	1		61	
2	3		18	307		1	2.8		0	1	0		27	
3	4		15	546		0	0.5		1	18	1		25	
4	5		14	134		0	1.4		0	11	1		49	
	•••			•••		•••	•••		•••		•••			
995	996			700		1	1.9		0	0	1		54	
996	997		$\epsilon$	309		0	1.8		1	0	0		13	
997	998		11	185		0	1.4		0	1	1		8	
998	999		15	533		1	0.5		1	0	0		50	
999	1000		12	270		1	0.5		0	4	1		35	
	m_dep	mobi	le_wt		рс	px_height	px_	width	ran	n sc_	h sc_	_w \		
0	0.1		193		16	226		1412	3476	5 1	2	7		
1	0.8		191		12	746		857	3895	5	6	0		
2	0.9		186		4	1270		1366	2396	5 1	7 1	.0		
3	0.5		96		20	295		1752	3893	3 1	0	0		
4	0.5		108	•••	18	749		810	1773	3 1	5	8		
	•••					•••	•••		•••					
995	0.5		170		17	644		913	2121	l 1	4	8		
996	0.9		186	•••	2	1152		1632	1933	3	8	1		
997	0.5		80	•••	12	477		825	1223	3	5	0		
998	0.4		171	•••	12	38		832	2509	9 1	5 1	.1		
999	0.1		140	•••	19	457		608	2828	3	9	2		
	talk_	time	three_	g.	touc	h_screen v	wifi							
0		2		0		1	0							
1		7		1		0	0							
2		10		0		1	1							
3		7		1		1	0							
4		7		1		0	1							
 995		 15	•••	1		 1	0							
996		19		0		1	1							
997		14		1		0	0							
998		6		0		1	0							
999		3		1		0	1							
223		3		1		U	1							

[1000 rows x 21 columns]

```
[]: from sklearn.model_selection import train_test_split
     x_train,x_test,y_train,y_test=train_test_split(x,y,train_size=0.
      →7, random_state=42)
     x_train.shape,x_test.shape
[]: ((700, 20), (300, 20))
[]: from sklearn.ensemble import RandomForestClassifier
     rfc=RandomForestClassifier()
     rfc.fit(x_train,y_train)
[ ]: RandomForestClassifier()
[]: rf=RandomForestClassifier()
     params={'max_depth': [2,3,5,10,20],
      'min_samples_leaf':[5,10,20,50,100,200],
      'n_estimators':[10,25,30,50,100,200]}
[]: from sklearn.model_selection import GridSearchCV
     grid_search=GridSearchCV(estimator=rf,param_grid=params,cv=2,scoring="accuracy")
     grid_search.fit(x_train,y_train)
[]: GridSearchCV(cv=2, estimator=RandomForestClassifier(),
                 param_grid={'max_depth': [2, 3, 5, 10, 20],
                              'min_samples_leaf': [5, 10, 20, 50, 100, 200],
                              'n_estimators': [10, 25, 30, 50, 100, 200]},
                  scoring='accuracy')
[]: grid_search.best_score_
[]: 0.5642857142857143
[]: rf_best=grid_search.best_estimator_
     print(rf_best)
    RandomForestClassifier(max_depth=2, min_samples_leaf=100, n_estimators=50)
[]: from sklearn.tree import plot_tree
     plt.figure(figsize=(80,40))
     plot_tree(rf_best.estimators_[5],feature_names=x.
      ⇔columns,class_names=["Yes","No"],filled=True);
```

```
class = No
                                            px height \leq 462.0
                  gini = 0.49
                                               gini = 0.499
                samples = 190
                                              samples = 264
              value = [124, 165]
                                            value = [216, 195]
                  class = No
                                                class = Yes
                                gini = 0.491
                                                                qini = 0.5
                               samples = 107
                                                             samples = 157
                               value = [98, 75]
                                                           value = [118, 120]
                                 class = Yes
                                                               class = No
[]: from sklearn.tree import plot_tree
     plt.figure(figsize=(80,40))
     plot_tree(rf_best.estimators_[6],feature_names=x.

columns,class_names=["Yes","No"],filled=True);
                                            mobile wt <= 171.5
                                               gini = 0.499
                                              samples = 445
                                            value = [363, 337]
                                                class = Yes
                                sc w <= 5.5
                                                               gini = 0.468
                                gini = 0.492
                                                             samples = 106
                               samples = 339
                                                            value = [63, 106]
                             value = [300, 231]
                                                               class = No
                                 class = Yes
                 gini = 0.481
                                                gini = 0.499
                samples = 204
                                              samples = 135
              value = [188, 126]
                                            value = [112, 105]
                  class = Yes
                                                class = Yes
[]: rf_best.feature_importances_
[]: array([0.05791869, 0.07187716, 0.00564471, 0.08348674, 0.00433501,
            0.07757976, 0.03435722, 0.12360436, 0.07147417, 0.14299703,
                      , 0.01379549, 0.04942909, 0.09405257, 0.1162142 ,
```

m\_dep <= 0.45 gini = 0.5 samples = 454 value = [340, 360]

```
])
           0.03188349, 0.00312057, 0.01822975, 0.
                                                    , 0.
[]: imp_df=pd.DataFrame({"Varname":x_train.columns,"IMP":rf_best.
     →feature_importances_})
     imp_df.sort_values(by="IMP",ascending=False)
[]:
                            IMP
              Varname
    9
            mobile_wt 0.142997
    7
           int_memory 0.123604
    14
                  ram 0.116214
    13
             px_width 0.094053
    3
          clock_speed 0.083487
    5
                   fc 0.077580
     1
        battery_power 0.071877
    8
                m_dep 0.071474
    0
                   id 0.057919
    12
            px_height 0.049429
               four_g 0.034357
    6
    15
                 sc_h 0.031883
    17
            talk_time 0.018230
    11
                   pc 0.013795
    2
                 blue 0.005645
    4
             dual_sim 0.004335
    16
                 sc_w 0.003121
    18
              three_g 0.000000
```

10

19

n\_cores 0.000000

touch\_screen 0.000000