RandomForest & Decision Tree

February 21, 2022

0.1 All library would be using on this code training dataset

[2]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import GridSearchCV, train_test_split
from sklearn.metrics import recall_score
from sklearn.metrics import auc
from sklearn.metrics import confusion_matrix
from sklearn.metrics import roc_curve
from sklearn.metrics import RocCurveDisplay
from sklearn.metrics import PrecisionRecallDisplay
import seaborn as sns
import math
```

[3]:
```
dataset = "C:/Users/User/OneDrive/Python/dataset/divorce.csv"
divorce = pd.read_csv(dataset)
divorce.head() # Showing the first 120 lines from original dataframe
```

[3]:

	Sorry_end	Ignore_diff	begin_correct	Contact	Special_time	No_home_time	\
0	2	2	4	1	0	0	
1	4	4	4	4	4	0	
2	2	2	2	2	1	3	
3	3	2	3	2	3	3	
4	2	2	1	1	1	1	

	2_strangers	enjoy_holiday	enjoy_travel	common_goals	…	\
0	0	0	0	0	…	
1	0	4	4	4	…	
2	2	1	1	2	…	
3	3	3	3	3	…	
4	0	0	0	0	…	

	silence_for_harm	silence_fear_anger	I'm_right	accusations	\
0	2	1	3	3	
1	2	2	3	4	
2	3	2	3	1	

```
3                 2                   2          3           3
4                 2                   1          2           3

   I'm_not_guilty  I'm_not_wrong  no_hesitancy_inadequate  you're_inadequate  \
0               3              2                        3                  2
1               4              4                        4                  2
2               1              1                        2                  2
3               3              3                        2                  2
4               2              2                        2                  1

   incompetence  Divorce_Y_N
0             1            1
1             2            1
2             2            1
3             2            1
4             0            1

[5 rows x 55 columns]
```

0.2 It‘s rather important to get to know the data, understanting the meaning from each values.

[4]:
```
divorce.describe() #
```

[4]:
```
        Sorry_end  Ignore_diff  begin_correct     Contact  Special_time  \
count  170.000000   170.000000     170.000000  170.000000    170.000000
mean     1.776471     1.652941       1.764706    1.482353      1.541176
std      1.627257     1.468654       1.415444    1.504327      1.632169
min      0.000000     0.000000       0.000000    0.000000      0.000000
25%      0.000000     0.000000       0.000000    0.000000      0.000000
50%      2.000000     2.000000       2.000000    1.000000      1.000000
75%      3.000000     3.000000       3.000000    3.000000      3.000000
max      4.000000     4.000000       4.000000    4.000000      4.000000

       No_home_time  2_strangers  enjoy_holiday  enjoy_travel  common_goals  \
count    170.000000   170.000000     170.000000    170.000000    170.000000
mean       0.747059     0.494118       1.452941      1.458824      1.576471
std        0.904046     0.898698       1.546371      1.557976      1.421529
min        0.000000     0.000000       0.000000      0.000000      0.000000
25%        0.000000     0.000000       0.000000      0.000000      0.000000
50%        0.000000     0.000000       1.000000      1.000000      2.000000
75%        1.000000     1.000000       3.000000      3.000000      3.000000
max        4.000000     4.000000       4.000000      4.000000      4.000000

       …  silence_for_harm  silence_fear_anger   I'm_right  accusations  \
count  …        170.000000          170.000000  170.000000   170.000000
mean   …          2.552941            2.270588    2.741176     2.382353
```

```
std    …          1.371786            1.586841    1.137348     1.511587
min    …          0.000000            0.000000    0.000000     0.000000
25%    …          2.000000            1.000000    2.000000     1.000000
50%    …          3.000000            2.000000    3.000000     3.000000
75%    …          4.000000            4.000000    4.000000     4.000000
max    …          4.000000            4.000000    4.000000     4.000000

       I'm_not_guilty  I'm_not_wrong  no_hesitancy_inadequate  \
count      170.000000     170.000000               170.000000
mean         2.429412       2.476471                 2.517647
std          1.405090       1.260238                 1.476537
min          0.000000       0.000000                 0.000000
25%          1.000000       2.000000                 1.000000
50%          2.000000       3.000000                 3.000000
75%          4.000000       4.000000                 4.000000
max          4.000000       4.000000                 4.000000

       you're_inadequate  incompetence  Divorce_Y_N
count         170.000000    170.000000   170.000000
mean            2.241176      2.011765     0.494118
std             1.505634      1.667611     0.501442
min             0.000000      0.000000     0.000000
25%             1.000000      0.000000     0.000000
50%             2.000000      2.000000     0.000000
75%             4.000000      4.000000     1.000000
max             4.000000      4.000000     1.000000

[8 rows x 55 columns]
```

[5]:
```
divorce.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 170 entries, 0 to 169
Data columns (total 55 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   Sorry_end                       170 non-null    int64
 1   Ignore_diff                     170 non-null    int64
 2   begin_correct                   170 non-null    int64
 3   Contact                         170 non-null    int64
 4   Special_time                    170 non-null    int64
 5   No_home_time                    170 non-null    int64
 6   2_strangers                     170 non-null    int64
 7   enjoy_holiday                   170 non-null    int64
 8   enjoy_travel                    170 non-null    int64
 9   common_goals                    170 non-null    int64
 10  harmony                         170 non-null    int64
```

```
 11  freeom_value                   170 non-null    int64
 12  entertain                      170 non-null    int64
 13  people_goals                   170 non-null    int64
 14  dreams                         170 non-null    int64
 15  love                           170 non-null    int64
 16  happy                          170 non-null    int64
 17  marriage                       170 non-null    int64
 18  roles                          170 non-null    int64
 19  trust                          170 non-null    int64
 20  likes                          170 non-null    int64
 21  care_sick                      170 non-null    int64
 22  fav_food                       170 non-null    int64
 23  stresses                       170 non-null    int64
 24  inner_world                    170 non-null    int64
 25  anxieties                      170 non-null    int64
 26  current_stress                 170 non-null    int64
 27  hopes_wishes                   170 non-null    int64
 28  know_well                      170 non-null    int64
 29  friends_social                 170 non-null    int64
 30  Aggro_argue                    170 non-null    int64
 31  Always_never                   170 non-null    int64
 32  negative_personality           170 non-null    int64
 33  offensive_expressions          170 non-null    int64
 34  insult                         170 non-null    int64
 35  humiliate                      170 non-null    int64
 36  not_calm                       170 non-null    int64
 37  hate_subjects                  170 non-null    int64
 38  sudden_discussion              170 non-null    int64
 39  idk_what's_going_on            170 non-null    int64
 40  calm_breaks                    170 non-null    int64
 41  argue_then_leave               170 non-null    int64
 42  silent_for_calm                170 non-null    int64
 43  good_to_leave_home             170 non-null    int64
 44  silence_instead_of_discussion  170 non-null    int64
 45  silence_for_harm               170 non-null    int64
 46  silence_fear_anger             170 non-null    int64
 47  I'm_right                      170 non-null    int64
 48  accusations                    170 non-null    int64
 49  I'm_not_guilty                 170 non-null    int64
 50  I'm_not_wrong                  170 non-null    int64
 51  no_hesitancy_inadequate        170 non-null    int64
 52  you're_inadequate              170 non-null    int64
 53  incompetence                   170 non-null    int64
 54  Divorce_Y_N                    170 non-null    int64
dtypes: int64(55)
memory usage: 73.2 KB
```

[6]:
```
## Same idea from SQL, Excel or any other visual software here, it`s filter, clean , delete any kind of
## pointless data that just increase the memory and make the code run longer than should
divorce.isna().sum()
```

[6]:
```
Sorry_end                  0
Ignore_diff                0
begin_correct              0
Contact                    0
Special_time               0
No_home_time               0
2_strangers                0
enjoy_holiday              0
enjoy_travel               0
common_goals               0
harmony                    0
freeom_value               0
entertain                  0
people_goals               0
dreams                     0
love                       0
happy                      0
marriage                   0
roles                      0
trust                      0
likes                      0
care_sick                  0
fav_food                   0
stresses                   0
inner_world                0
anxieties                  0
current_stress             0
hopes_wishes               0
know_well                  0
friends_social             0
Aggro_argue                0
Always_never               0
negative_personality       0
offensive_expressions      0
insult                     0
humiliate                  0
not_calm                   0
hate_subjects              0
sudden_discussion          0
idk_what's_going_on        0
calm_breaks                0
```

```
argue_then_leave                  0
silent_for_calm                   0
good_to_leave_home                0
silence_instead_of_discussion     0
silence_for_harm                  0
silence_fear_anger                0
I'm_right                         0
accusations                       0
I'm_not_guilty                    0
I'm_not_wrong                     0
no_hesitancy_inadequate           0
you're_inadequate                 0
incompetence                      0
Divorce_Y_N                       0
dtype: int64
```

[7]:
```
X =divorce.drop(['Divorce_Y_N'], axis =1)#Erasing the target dataset
X # Showing the usefull dataset
```

[7]:
```
     Sorry_end  Ignore_diff  begin_correct  Contact  Special_time  \
0            2            2              4        1             0
1            4            4              4        4             4
2            2            2              2        2             1
3            3            2              3        2             3
4            2            2              1        1             1
..         …          …            …      …          …
165          0            0              0        0             0
166          0            0              0        0             0
167          1            1              0        0             0
168          0            0              0        0             0
169          0            0              0        0             0

     No_home_time  2_strangers  enjoy_holiday  enjoy_travel  common_goals  \
0               0            0              0             0             0
1               0            0              4             4             4
2               3            2              1             1             2
3               3            3              3             3             3
4               1            0              0             0             0
..            …          …            …          …           …
165             0            0              0             0             0
166             0            0              0             0             0
167             0            0              0             0             1
168             0            0              0             0             0
169             0            0              1             0             0

     …  silence_instead_of_discussion  silence_for_harm  silence_fear_anger  \
0    …                              3                 2                   1
```

```
1    …                              2                2                    2
2    …                              2                3                    2
3    …                              3                2                    2
4    …                              2                2                    1
..   …                            …                …                    …
165  …                              0                1                    0
166  …                              3                4                    1
167  …                              2                3                    0
168  …                              4                3                    3
169  …                              1                3                    4

     I'm_right  accusations  I'm_not_guilty  I'm_not_wrong  \
0            3            3               3              2
1            3            4               4              4
2            3            1               1              1
3            3            3               3              3
4            2            3               2              2
..         …            …               …              …
165          4            1               1              4
166          2            2               2              2
167          2            0               1              1
168          2            2               3              2
169          4            0               1              3

     no_hesitancy_inadequate  you're_inadequate  incompetence
0                          3                  2             1
1                          4                  2             2
2                          2                  2             2
3                          2                  2             2
4                          2                  1             0
..                       …                  …             …
165                        2                  2             2
166                        3                  2             2
167                        3                  0             0
168                        4                  3             1
169                        3                  3             1

[170 rows x 54 columns]
```

[8]:
```
y = divorce['Divorce_Y_N']
y
```

[8]:
```
0      1
1      1
2      1
3      1
4      1
```

```
..
165    0
166    0
167    0
168    0
169    0
Name: Divorce_Y_N, Length: 170, dtype: int64
```

[9]:
```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33,␣
↪random_state=None)
method = RandomForestClassifier()
parametrs = {'n_estimators': range(10, 50, 10),
'max_depth': range(1, 12, 2),
'min_samples_leaf': range(1, 7),
'min_samples_split': range(2, 9, 2)}
grid_search = GridSearchCV(method, parametrs, cv=5, n_jobs=-1, verbose=1)
grid_search.fit(X_train, y_train)
grid_search.best_params_
```

```
Fitting 5 folds for each of 576 candidates, totalling 2880 fits
```

[9]:
```
{'max_depth': 1,
 'min_samples_leaf': 3,
 'min_samples_split': 4,
 'n_estimators': 10}
```

[10]:
```
best_estimator = grid_search.best_estimator_
best_estimator.score(X_test,y_test)
#Above we have the measure from our RandomForestClassifier on this dataset
```

[10]:
```
0.9649122807017544
```

[ ]:
```

```

[11]:
```
feature_importance =best_estimator.feature_importances_
feature_importance_df = pd.DataFrame({'features': list(X_train),
'feature_importance': feature_importance})
feature_importance_df = feature_importance_df.sort_values('feature_importance',␣
↪ascending=False)
feature_importance_df
```

[11]:

	features	feature_importance
25	anxieties	0.2
36	not_calm	0.2
17	marriage	0.2
15	love	0.1
16	happy	0.1

7	enjoy_holiday	0.1
8	enjoy_travel	0.1
32	negative_personality	0.0
33	offensive_expressions	0.0
34	insult	0.0
35	humiliate	0.0
37	hate_subjects	0.0
38	sudden_discussion	0.0
39	idk_what's_going_on	0.0
40	calm_breaks	0.0
41	argue_then_leave	0.0
0	Sorry_end	0.0
30	Aggro_argue	0.0
42	silent_for_calm	0.0
43	good_to_leave_home	0.0
44	silence_instead_of_discussion	0.0
45	silence_for_harm	0.0
46	silence_fear_anger	0.0
47	I'm_right	0.0
48	accusations	0.0
49	I'm_not_guilty	0.0
50	I'm_not_wrong	0.0
51	no_hesitancy_inadequate	0.0
52	you're_inadequate	0.0
31	Always_never	0.0
27	hopes_wishes	0.0
29	friends_social	0.0
13	people_goals	0.0
2	begin_correct	0.0
3	Contact	0.0
4	Special_time	0.0
5	No_home_time	0.0
6	2_strangers	0.0
9	common_goals	0.0
10	harmony	0.0
11	freeom_value	0.0
12	entertain	0.0
14	dreams	0.0
28	know_well	0.0
18	roles	0.0
19	trust	0.0
20	likes	0.0
21	care_sick	0.0
22	fav_food	0.0
23	stresses	0.0
24	inner_world	0.0
26	current_stress	0.0

```
1                    Ignore_diff                0.0
53                   incompetence               0.0
```

[12]:
```
X_short = X[['hate_subjects', 'roles', 'dreams', 'humiliate', 'likes',
'happy', 'marriage', 'love','freeom_value', 'anxieties',
'enjoy_travel', 'Ignore_diff', 'Special_time', 'care_sick']]
```

[13]:
```
X_short ## Printing some characteristics from broken marriage
```

[13]:
```
     hate_subjects  roles  dreams  humiliate  likes  happy  marriage  love  \
0                1      0       0          1      0      0         0     1
1                4      3       4          2      1      4         4     4
2                1      3       3          1      1      3         3     3
3                3      3       3          1      1      3         3     3
4                0      2       1          0      1      1         1     1
..             …    …     …         …    …    …       …   …
165              0      0       0          0      0      0         0     0
166              1      0       0          1      0      0         0     0
167              1      0       0          0      0      0         0     1
168              0      0       0          0      0      0         0     0
169              0      0       0          0      1      0         0     0

     freeom_value  anxieties  enjoy_travel  Ignore_diff  Special_time  \
0               0          0             0            2             0
1               3          1             4            4             4
2               4          2             1            2             1
3               3          1             3            2             3
4               1          2             0            2             1
..            …        …           …          …           …
165             0          0             0            0             0
166             0          0             0            0             0
167             1          1             0            1             0
168             1          0             0            0             0
169             1          0             0            0             0

     care_sick
0            0
1            1
2            0
3            1
4            0
..         …
165          0
166          0
167          0
168          0
169          0
```

```
[170 rows x 14 columns]
```

[14]:
```
X_train_sh, X_test_sh, y_train_sh, y_test_sh = train_test_split(X_short, y,␣
↪test_size=0.33)
method = RandomForestClassifier()
parametrs = {'n_estimators': range(10, 50, 10),
'max_depth': range(1, 12, 2),
'min_samples_leaf': range(1, 7),
'min_samples_split': range(2, 9, 2)}
grid_search = GridSearchCV(method, parametrs, cv=5, n_jobs=-1, verbose=1)
grid_search.fit(X_train_sh, y_train_sh)
grid_search.best_params_
```

```
Fitting 5 folds for each of 576 candidates, totalling 2880 fits
```

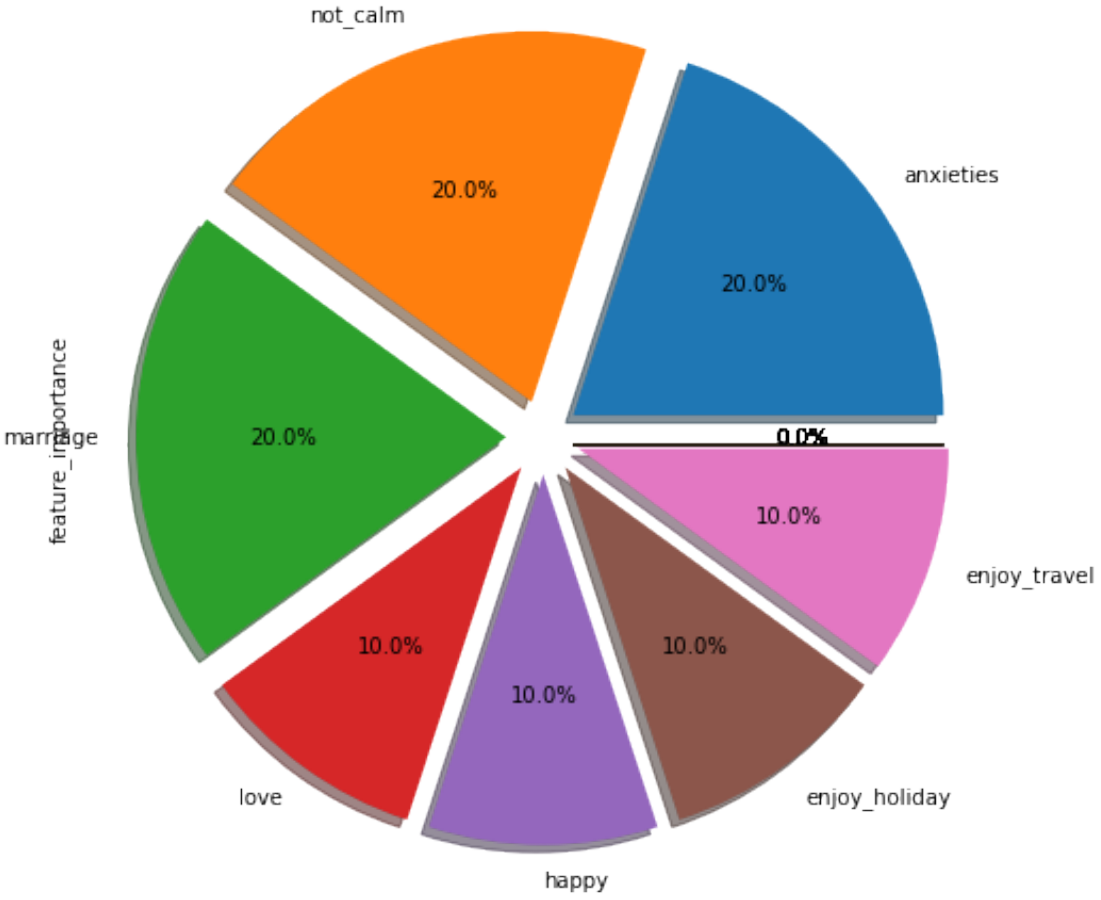
[14]:
```
{'max_depth': 1,
 'min_samples_leaf': 3,
 'min_samples_split': 8,
 'n_estimators': 20}
```

[15]:
```
best_estimator = grid_search.best_estimator_
best_estimator.score(X_test_sh, y_test_sh)
```

[15]:
```
0.9824561403508771
```

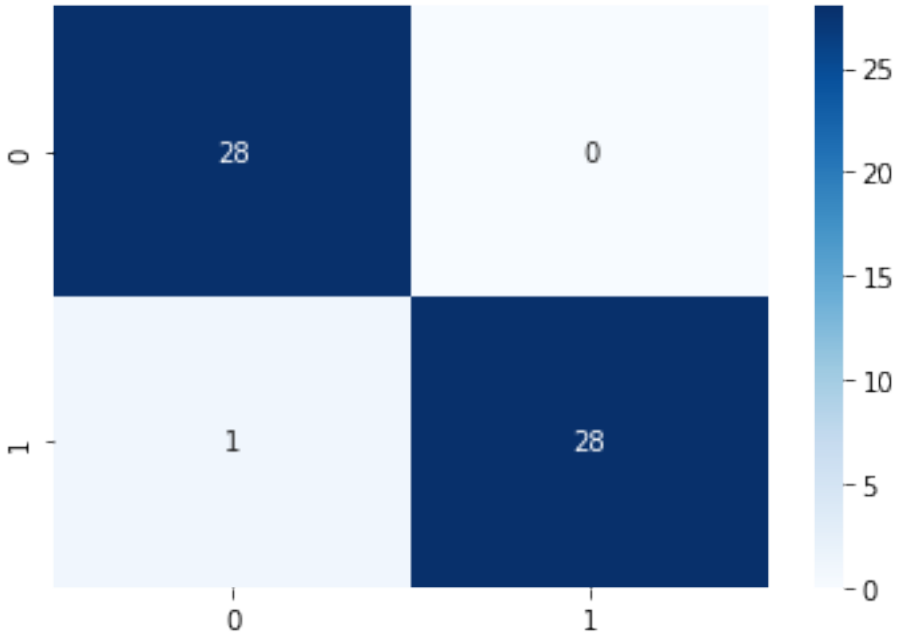
0.3 Showing by pizza graphics the aproximately value from each characteristics from a broken marriage

[16]:
```
feature_importance_df.plot.pie(
explode=[0.1]*len(X_train.columns),
labels = feature_importance_df.features,
y = 'feature_importance',
autopct='%1.1f%%',
shadow=True,
legend=False,
figsize=(8, 8));
```



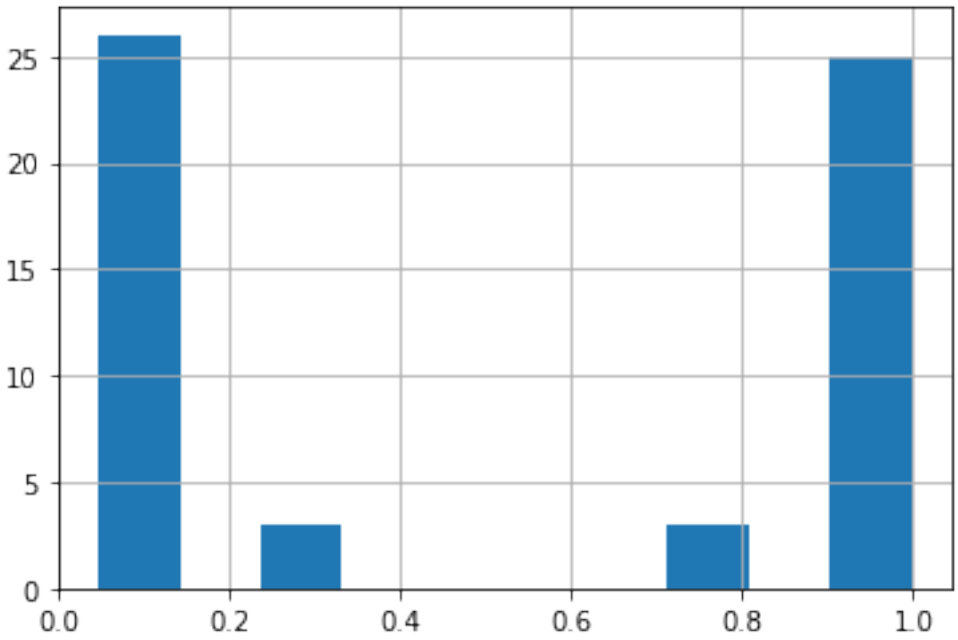
[17]:
```
y_pred = best_estimator.predict(X_test_sh)
sns.heatmap(confusion_matrix(y_test_sh, y_pred), annot=True, cmap="Blues")
```

[17]: <AxesSubplot:>

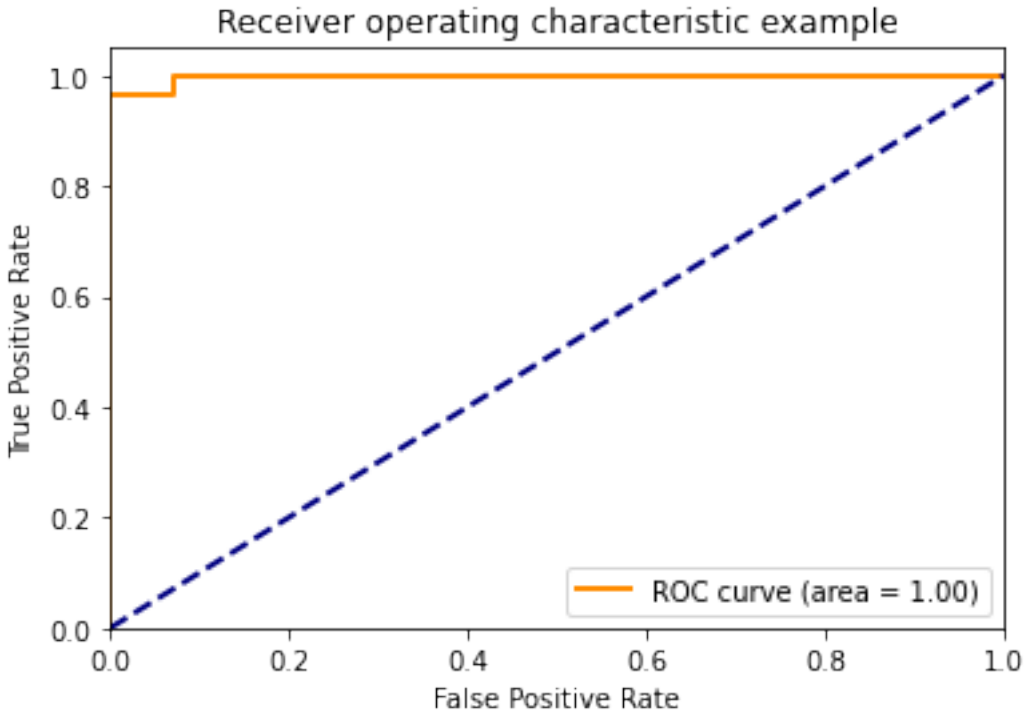


[18]:
```
propability = best_estimator.predict_proba(X_test_sh)
true_probability = propability[:,1]
pd.Series(true_probability).hist()
```

[18]: <AxesSubplot:>

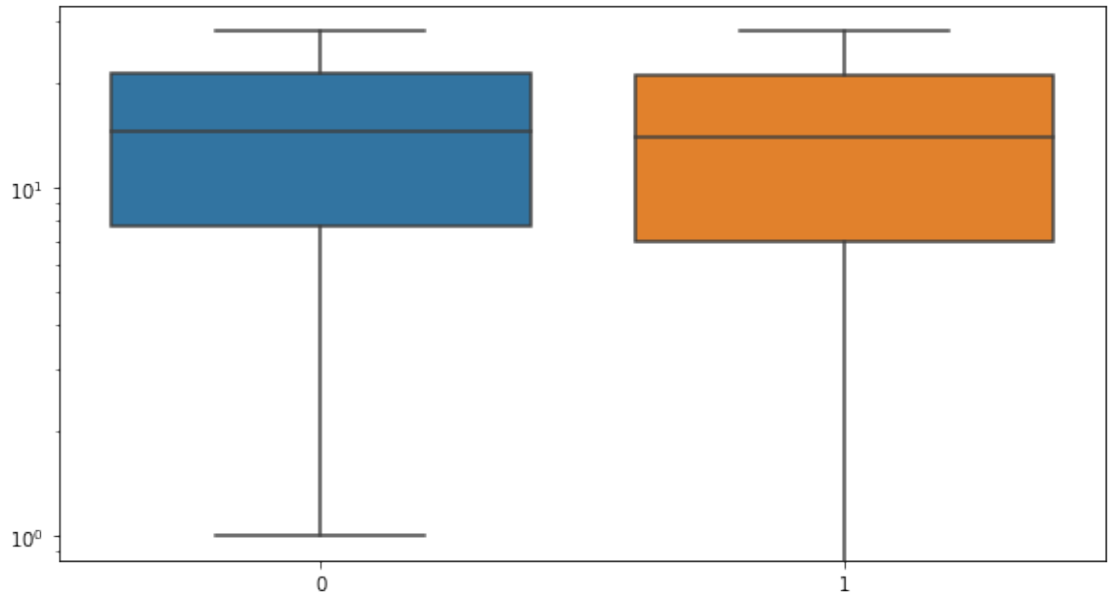


[19]:
```
fpr, tpr, thresholds = roc_curve(y_test_sh, true_probability)
roc_auc= auc(fpr, tpr)
lw=2
plt.figure()
plt.plot(fpr, tpr, color='darkorange',
lw=lw, label='ROC curve (area = %0.2f)' % roc_auc)
plt.plot([0, 1], [0, 1], color='navy', lw=lw, linestyle='--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver operating characteristic example')
plt.legend(loc="lower right")
plt.show()
```



[ ]:
```
```

[20]:
```
a4_dims = (9.7, 5.27)
fig, ax = plt.subplots(figsize=a4_dims)
g = sns.boxplot(data=confusion_matrix(y_test_sh, y_pred),linewidth=1.5,ax=ax)
g.set_yscale("log")
```



[ ]:

0.3.1 Using DecisionTreeClassifier with pedagogical way to explain and understand your only aplication

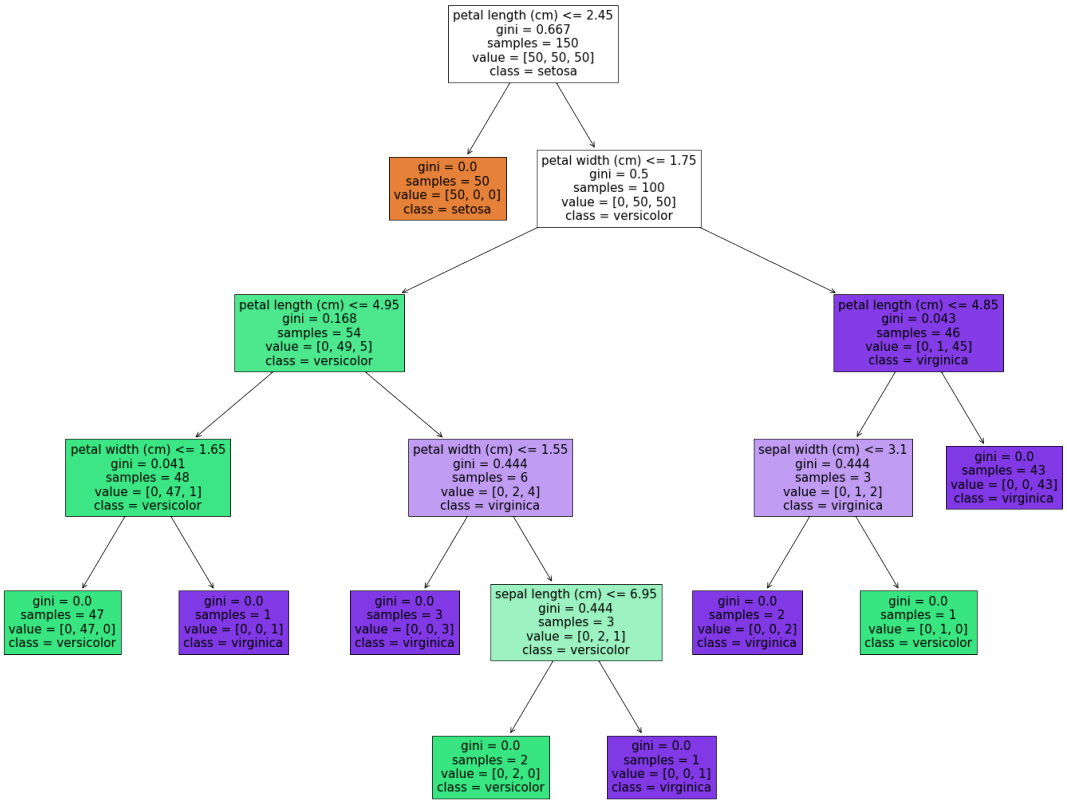
[28]:
```
from sklearn.model_selection import cross_val_score
from sklearn.tree import DecisionTreeRegressor
from sklearn.tree import DecisionTreeClassifier
from sklearn import datasets
from sklearn import tree
# Fit the classifier with default hyper-parameters
# Prepare the data data
iris = datasets.load_iris()
X = iris.data
y = iris.target
# Fit the classifier with default hyper-parameters
clf = DecisionTreeClassifier(random_state=1234)
model = clf.fit(X, y)
```

[29]:
```
text_representation = tree.export_text(clf)
print(text_representation)
```

```
|--- feature_2 <= 2.45
|   |--- class: 0
|--- feature_2 >  2.45
|   |--- feature_3 <= 1.75
|   |   |--- feature_2 <= 4.95
```

```
|   |   |   |--- feature_3 <= 1.65
|   |   |   |   |--- class: 1
|   |   |   |--- feature_3 >  1.65
|   |   |   |   |--- class: 2
|   |   |--- feature_2 >  4.95
|   |   |   |--- feature_3 <= 1.55
|   |   |   |   |--- class: 2
|   |   |   |--- feature_3 >  1.55
|   |   |   |   |--- feature_0 <= 6.95
|   |   |   |   |   |--- class: 1
|   |   |   |   |--- feature_0 >  6.95
|   |   |   |   |   |--- class: 2
|   |--- feature_3 >  1.75
|   |   |--- feature_2 <= 4.85
|   |   |   |--- feature_1 <= 3.10
|   |   |   |   |--- class: 2
|   |   |   |--- feature_1 >  3.10
|   |   |   |   |--- class: 1
|   |   |--- feature_2 >  4.85
|   |   |   |--- class: 2
```

[38]:
```
with open("decistion_tree.log", "w") as fout:
    fout.write(text_representation)
fig = plt.figure(figsize=(25,20))
_ = tree.plot_tree(clf,
                   feature_names=iris.feature_names,
                   class_names=iris.target_names,
                   filled=True)
```


petal length (cm) <= 2.45
gini = 0.667
samples = 150
value = [50, 50, 50]
class = setosa
gini = 0.0
samples = 50
value = [50, 0, 0]
class = setosa
petal width (cm) <= 1.75
gini = 0.5
samples = 100
value = [0, 50, 50]
class = versicolor
petal length (cm) <= 4.95
gini = 0.168
samples = 54
value = [0, 49, 5]
class = versicolor
petal length (cm) <= 4.85
gini = 0.043
samples = 46
value = [0, 1, 45]
class = virginica
petal width (cm) <= 1.65
gini = 0.041
samples = 48
value = [0, 47, 1]
class = versicolor
petal width (cm) <= 1.55
gini = 0.444
samples = 6
value = [0, 2, 4]
class = virginica
sepal width (cm) <= 3.1
gini = 0.444
samples = 3
value = [0, 1, 2]
class = virginica
gini = 0.0
samples = 43
value = [0, 0, 43]
class = virginica
gini = 0.0
samples = 47
value = [0, 47, 0]
class = versicolor
gini = 0.0
samples = 1
value = [0, 0, 1]
class = virginica
gini = 0.0
samples = 3
value = [0, 0, 3]
class = virginica
sepal length (cm) <= 6.95
gini = 0.444
samples = 3
value = [0, 2, 1]
class = versicolor
gini = 0.0
samples = 2
value = [0, 0, 2]
class = virginica
gini = 0.0
samples = 1
value = [0, 1, 0]
class = versicolor
gini = 0.0
samples = 2
value = [0, 2, 0]
class = versicolor
gini = 0.0
samples = 1
value = [0, 0, 1]
class = virginica

[33]: