Random_forest_hw

January 26, 2022

1 what are N_Estimators?

- n_estimators : This is the number of trees you want to build before taking the maximum voting or averages of predictions. Higher number of trees give you better performance but makes your code slower

2 Random forest regressor?

- Predictions of sepal_width using (petal_length, petal_width, sepal_length)

```
[ ]: # load sample datasets
     import pandas as pd
     import seaborn as sns
     import numpy as np
     df= sns.load_dataset('iris')
     df.head()
```

```
[ ]:    sepal_length  sepal_width  petal_length  petal_width species
     0           5.1          3.5           1.4          0.2  setosa
     1           4.9          3.0           1.4          0.2  setosa
     2           4.7          3.2           1.3          0.2  setosa
     3           4.6          3.1           1.5          0.2  setosa
     4           5.0          3.6           1.4          0.2  setosa
```

```
[ ]: x= df[['sepal_length','petal_width', 'petal_length']]
     y= df['sepal_width']
```

```
[ ]: x.head()
```

```
[ ]:    sepal_length  petal_width  petal_length
     0           5.1          0.2           1.4
     1           4.9          0.2           1.4
     2           4.7          0.2           1.3
     3           4.6          0.2           1.5
     4           5.0          0.2           1.4
```

```
[ ]: y.tail()
[ ]: 145    3.0
     146    2.5
     147    3.0
     148    3.4
     149    3.0
     Name: sepal_width, dtype: float64
```

```
[ ]: from sklearn.ensemble import RandomForestRegressor
     model = RandomForestRegressor(n_estimators=50)
     model
     model.fit(x,y)
     model.predict([[4,2,6]])
```

```
C:\Users\Haier\AppData\Local\Programs\Python\Python310\lib\site-
packages\sklearn\base.py:450: UserWarning: X does not have valid feature names,
but RandomForestRegressor was fitted with feature names
  warnings.warn(
```

```
[ ]: array([2.688])
```

```
[ ]: 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)
     predictions= model.predict(x_test)
     predictions
```

```
[ ]: array([3.152     , 2.866     , 2.906     , 2.66      , 2.786     ,
            3.104     , 3.442     , 3.2874    , 3.12      , 2.478     ,
            2.954     , 3.31      , 3.4715    , 3.126     , 2.86      ,
            2.742     , 2.952     , 3.09393333, 3.24425714, 3.457     ,
            3.926     , 3.272     , 2.87      , 3.048     , 2.152     ,
            3.66      , 3.064     , 2.69      , 3.662     , 3.19266667])
```

```
[ ]: # accuracy test
     score= model.score(x_test, y_test)
     print ('The accuracy score is', score)
```

```
The accuracy score is 0.939524052615136
```

- confusion matrix is not working instead it says continuous