Importing necessary libraries

In [2]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

Loading Mercedes Benz dataset

In [3]:
```
benz_train= pd.read_csv(r"C:\Users\admin\Desktop\python\ML\simplilearn project
s\train.csv")
benz_test= pd.read_csv(r"C:\Users\admin\Desktop\python\ML\simplilearn projects
\test.csv")
```

Checking the first five rows of data set

In [4]:
```
benz_train.head()
```

Out[4]:

	ID	y	X0	X1	X2	X3	X4	X5	X6	X8	...	X375	X376	X377	X378	X379	X380	X38
0	0	130.81	k	v	at	a	d	u	j	o	...	0	0	1	0	0	0	
1	6	88.53	k	t	av	e	d	y	l	o	...	1	0	0	0	0	0	
2	7	76.26	az	w	n	c	d	x	j	x	...	0	0	0	0	0	0	
3	9	80.62	az	t	n	f	d	x	l	e	...	0	0	0	0	0	0	
4	13	78.02	az	v	n	f	d	h	d	n	...	0	0	0	0	0	0	

5 rows × 378 columns

In [5]:
```
benz_test.head()
```

Out[5]:

	ID	X0	X1	X2	X3	X4	X5	X6	X8	X10	...	X375	X376	X377	X378	X379	X380	X382
0	1	az	v	n	f	d	t	a	w	0	...	0	0	0	1	0	0	0
1	2	t	b	ai	a	d	b	g	y	0	...	0	0	1	0	0	0	0
2	3	az	v	as	f	d	a	j	j	0	...	0	0	0	1	0	0	0
3	4	az	l	n	f	d	z	l	n	0	...	0	0	0	1	0	0	0
4	5	w	s	as	c	d	y	i	m	0	...	1	0	0	0	0	0	0

5 rows × 377 columns

Checking for any missing values

In [6]:
```
benz_train.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4209 entries, 0 to 4208
Columns: 378 entries, ID to X385
dtypes: float64(1), int64(369), object(8)
memory usage: 12.1+ MB
```

Checking for unique values after 10th column

In [7]:
```
np.unique(benz_train[benz_train.columns[10:]])
```

Out[7]:
```
array([0, 1], dtype=int64)
```

In [8]:
```
benz_train.columns
```

Out[8]:
```
Index(['ID', 'y', 'X0', 'X1', 'X2', 'X3', 'X4', 'X5', 'X6', 'X8',
       ...
       'X375', 'X376', 'X377', 'X378', 'X379', 'X380', 'X382', 'X383', 'X38
4',
       'X385'],
      dtype='object', length=378)
```

Checking for null values

In [9]:
```
train_null= pd.concat([benz_train.isnull().sum()], keys=['Count of Null value
s'], axis=1)
train_null[train_null.sum(axis=1) > 0]
```

Out[9]:

Count of Null values

In [10]:
```
test_null= pd.concat([benz_test.isnull().sum()], keys=['Count of Null values'
], axis=1)
test_null[test_null.sum(axis=1) > 0]
```

Out[10]:

Count of Null values

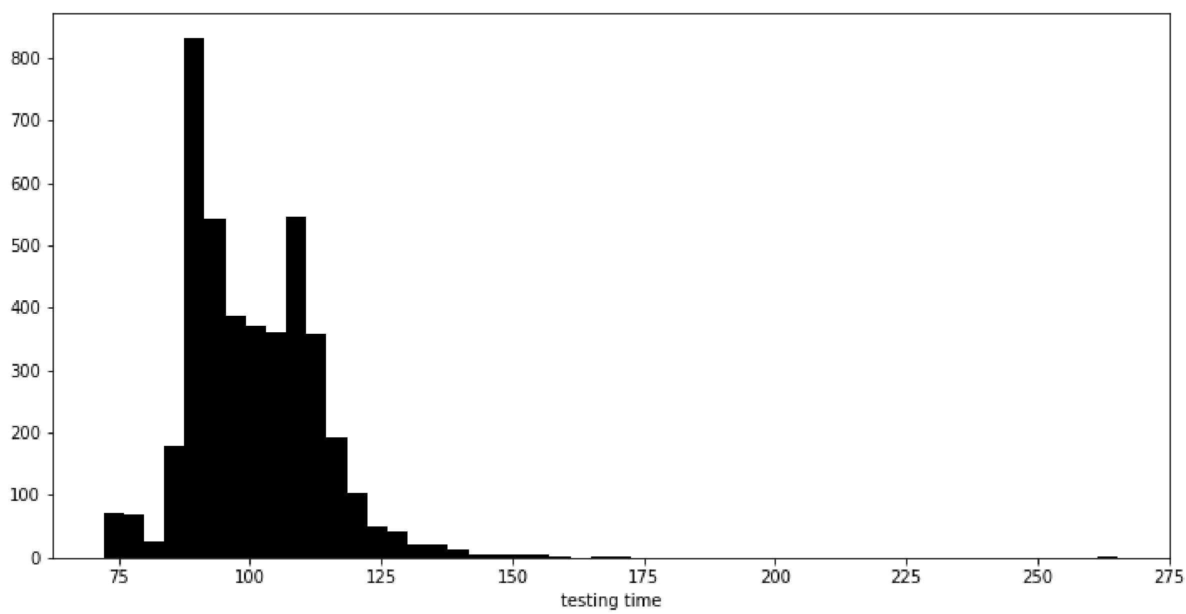
In [11]:
```
benz_train['y'].describe()
```

Out[11]:
```
count    4209.000000
mean      100.669318
std        12.679381
min        72.110000
25%        90.820000
50%        99.150000
75%       109.010000
max       265.320000
Name: y, dtype: float64
```

In [12]:
```
plt.figure(figsize=(12,6))


plt.hist(benz_train.y, bins=50, color='k')
plt.xlabel("testing time")
```

Out[12]:
```
Text(0.5, 0, 'testing time')
```



Checking for Normality using Shapiro and Anderson tests

In [13]:
```
from scipy.stats import shapiro
stat, p= shapiro(benz_train['y'])
if p>= 0.05:
    print("Alternate Hypothesis(H1): y is normally distributed")
else:
    print("Null Hypothesis(H0): y is not normally distributed")
print("Confidence level: {}%".format(((1-p)*100)))
```

```
Null Hypothesis(H0): y is not normally distributed
Confidence level: 100.0%
```

In [14]:
```
from scipy.stats import anderson
stat= anderson(benz_train['y'])
print(stat)
```

```
AndersonResult(statistic=43.24800930089441, critical_values=array([0.575, 0.6
55, 0.786, 0.917, 1.091]), significance_level=array([15. , 10. ,  5. ,  2.5,
1. ]))
```

Separating numeric and categorical columns in the training dataset

In [15]:
```
numerics= ["int16", "int32", "int64", "float16", "float32", "float64"]
cat= ["O"]
```

In [16]:
```
train_num= benz_train.select_dtypes(include= numerics)
train_cat= benz_train.select_dtypes(include= cat)
```

In [17]:
```
print(train_num.shape, train_cat.shape)
print("Numerical columns in training", train_num.columns)
print("Categorical columns in training", train_cat.columns)
```

```
(4209, 370) (4209, 8)
Numerical columns in training Index(['ID', 'y', 'X10', 'X11', 'X12', 'X13',
'X14', 'X15', 'X16', 'X17',
       ...
       'X375', 'X376', 'X377', 'X378', 'X379', 'X380', 'X382', 'X383', 'X38
4',
       'X385'],
      dtype='object', length=370)
Categorical columns in training Index(['X0', 'X1', 'X2', 'X3', 'X4', 'X5', 'X
6', 'X8'], dtype='object')
```

Finding unique values in categorical columns of training dataset

In [18]:
```
for i in train_cat.columns:
    print("The no of unique values in {} are:{}".format(i, train_cat[i].nuniqu
e()))
    print("Unique values are:", train_cat[i].unique())
    print("--------------------------------------------------------------------
------------------------")
```

```
The no of unique values in X0 are:47
Unique values are: ['k' 'az' 't' 'al' 'o' 'w' 'j' 'h' 's' 'n' 'ay' 'f' 'x'
'y' 'aj' 'ak' 'am'
 'z' 'q' 'at' 'ap' 'v' 'af' 'a' 'e' 'ai' 'd' 'aq' 'c' 'aa' 'ba' 'as' 'i'
 'r' 'b' 'ax' 'bc' 'u' 'ad' 'au' 'm' 'l' 'aw' 'ao' 'ac' 'g' 'ab']
-----------------------------------------------------------------------------
--------------
The no of unique values in X1 are:27
Unique values are: ['v' 't' 'w' 'b' 'r' 'l' 's' 'aa' 'c' 'a' 'e' 'h' 'z' 'j'
'o' 'u' 'p' 'n'
 'i' 'y' 'd' 'f' 'm' 'k' 'g' 'q' 'ab']
-----------------------------------------------------------------------------
--------------
The no of unique values in X2 are:44
Unique values are: ['at' 'av' 'n' 'e' 'as' 'aq' 'r' 'ai' 'ak' 'm' 'a' 'k' 'a
e' 's' 'f' 'd'
 'ag' 'ay' 'ac' 'ap' 'g' 'i' 'aw' 'y' 'b' 'ao' 'al' 'h' 'x' 'au' 't' 'an'
 'z' 'ah' 'p' 'am' 'j' 'q' 'af' 'l' 'aa' 'c' 'o' 'ar']
-----------------------------------------------------------------------------
--------------
The no of unique values in X3 are:7
Unique values are: ['a' 'e' 'c' 'f' 'd' 'b' 'g']
-----------------------------------------------------------------------------
--------------
The no of unique values in X4 are:4
Unique values are: ['d' 'b' 'c' 'a']
-----------------------------------------------------------------------------
--------------
The no of unique values in X5 are:29
Unique values are: ['u' 'y' 'x' 'h' 'g' 'f' 'j' 'i' 'd' 'c' 'af' 'ag' 'ab' 'a
c' 'ad' 'ae'
 'ah' 'l' 'k' 'n' 'm' 'p' 'q' 's' 'r' 'v' 'w' 'o' 'aa']
-----------------------------------------------------------------------------
--------------
The no of unique values in X6 are:12
Unique values are: ['j' 'l' 'd' 'h' 'i' 'a' 'g' 'c' 'k' 'e' 'f' 'b']
-----------------------------------------------------------------------------
--------------
The no of unique values in X8 are:25
Unique values are: ['o' 'x' 'e' 'n' 's' 'a' 'h' 'p' 'm' 'k' 'd' 'i' 'v' 'j'
'b' 'q' 'w' 'g'
 'y' 'l' 'f' 'u' 'r' 't' 'c']
-----------------------------------------------------------------------------
--------------
```

Separating numeric and categorical columns in testing dataset

In [19]:
```
test_num= benz_test.select_dtypes(include= numerics)
test_cat= benz_test.select_dtypes(include= cat)
```

In [20]:
```
print(test_num.shape, test_cat.shape)
print("Numerical columns in testing", test_num.columns)
print("Categorical columns in testing", test_cat.columns)
```

```
(4209, 369) (4209, 8)
Numerical columns in testing Index(['ID', 'X10', 'X11', 'X12', 'X13', 'X14',
'X15', 'X16', 'X17', 'X18',
       ...
       'X375', 'X376', 'X377', 'X378', 'X379', 'X380', 'X382', 'X383', 'X38
4',
       'X385'],
      dtype='object', length=369)
Categorical columns in testing Index(['X0', 'X1', 'X2', 'X3', 'X4', 'X5', 'X
6', 'X8'], dtype='object')
```

Finding the unique values of categorical columns in the testing dataset

In [21]:
```
for i in test_cat.columns:
    print("The no of unique values in {} are {}".format(i, test_cat[i].nunique
()))
    print("Unique values are", test_cat[i].unique())
    print("-----------------------------------------------------------------------------
--------------------------")
```

```
The no of unique values in X0 are 49
Unique values are ['az' 't' 'w' 'y' 'x' 'f' 'ap' 'o' 'ay' 'al' 'h' 'z' 'aj'
'd' 'v' 'ak'
 'ba' 'n' 'j' 's' 'af' 'ax' 'at' 'aq' 'av' 'm' 'k' 'a' 'e' 'ai' 'i' 'ag'
 'b' 'am' 'aw' 'as' 'r' 'ao' 'u' 'l' 'c' 'ad' 'au' 'bc' 'g' 'an' 'ae' 'p'
 'bb']
-----------------------------------------------------------------------------
----------------
The no of unique values in X1 are 27
Unique values are ['v' 'b' 'l' 's' 'aa' 'r' 'a' 'i' 'p' 'c' 'o' 'm' 'z' 'e'
'h' 'w' 'g' 'k'
 'y' 't' 'u' 'd' 'j' 'q' 'n' 'f' 'ab']
-----------------------------------------------------------------------------
----------------
The no of unique values in X2 are 45
Unique values are ['n' 'ai' 'as' 'ae' 's' 'b' 'e' 'ak' 'm' 'a' 'aq' 'ag' 'r'
'k' 'aj' 'ay'
 'ao' 'an' 'ac' 'af' 'ax' 'h' 'i' 'f' 'ap' 'p' 'au' 't' 'z' 'y' 'aw' 'd'
 'at' 'g' 'am' 'j' 'x' 'ab' 'w' 'q' 'ah' 'ad' 'al' 'av' 'u']
-----------------------------------------------------------------------------
----------------
The no of unique values in X3 are 7
Unique values are ['f' 'a' 'c' 'e' 'd' 'g' 'b']
-----------------------------------------------------------------------------
----------------
The no of unique values in X4 are 4
Unique values are ['d' 'b' 'a' 'c']
-----------------------------------------------------------------------------
----------------
The no of unique values in X5 are 32
Unique values are ['t' 'b' 'a' 'z' 'y' 'x' 'h' 'g' 'f' 'j' 'i' 'd' 'c' 'af'
'ag' 'ab' 'ac'
 'ad' 'ae' 'ah' 'l' 'k' 'n' 'm' 'p' 'q' 's' 'r' 'v' 'w' 'o' 'aa']
-----------------------------------------------------------------------------
----------------
The no of unique values in X6 are 12
Unique values are ['a' 'g' 'j' 'l' 'i' 'd' 'f' 'h' 'c' 'k' 'e' 'b']
-----------------------------------------------------------------------------
----------------
The no of unique values in X8 are 25
Unique values are ['w' 'y' 'j' 'n' 'm' 's' 'a' 'v' 'r' 'o' 't' 'h' 'c' 'k'
'p' 'u' 'd' 'g'
 'b' 'q' 'e' 'l' 'f' 'i' 'x']
-----------------------------------------------------------------------------
----------------
```

Checking for Outliers

In [22]:
```
def outlier(data):
    Q1, Q3= np.percentile(data, [25,75])
    IQR= Q3-Q1
    LR= Q1-(1.5*IQR)
    UR= Q3+(1.5*IQR)
    return LR, UR
```

In [23]:
```
lower,upper= outlier(benz_train.y)
```

In [24]:
```
benz_train[benz_train['y'] > upper]
```

Out[24]:

	ID	y	X0	X1	X2	X3	X4	X5	X6	X8	...	X375	X376	X377	X378	X379	X380
43	107	139.20	w	s	as	c	d	j	i	q	...	1	0	0	0	0	0
203	416	136.41	w	s	as	c	d	i	i	w	...	1	0	0	0	0	0
216	433	146.83	x	i	as	c	d	i	g	l	...	0	0	1	0	0	0
253	505	150.43	t	b	as	c	d	i	l	x	...	0	0	1	0	0	0
342	681	169.91	aa	l	ak	f	d	i	c	d	...	0	0	0	0	0	0
420	822	136.47	x	b	h	c	d	d	j	q	...	0	0	1	0	0	0
429	836	154.87	ak	l	ae	f	d	d	g	w	...	0	0	0	0	0	0
681	1322	147.72	x	i	ae	c	d	c	g	y	...	0	0	1	0	0	0
846	1671	140.49	x	aa	i	c	d	af	l	c	...	1	0	0	0	0	0
883	1770	265.32	y	r	ai	f	d	ag	l	t	...	0	0	0	0	0	0
889	1784	158.53	aj	l	as	f	d	ag	k	e	...	0	0	0	0	0	0
900	1799	141.31	x	aa	as	c	d	ag	j	j	...	1	0	0	0	0	0
995	1989	140.15	x	b	m	c	d	ag	j	j	...	0	0	1	0	0	0
998	1992	137.44	j	r	ae	c	d	ag	i	o	...	1	0	0	0	0	0
1033	2058	140.41	x	aa	n	e	d	ag	l	j	...	1	0	0	0	0	0
1036	2065	144.36	x	aa	as	d	d	ag	d	s	...	0	1	0	0	0	0
1060	2111	154.43	w	v	r	c	d	ag	d	q	...	1	0	0	0	0	0
1141	2264	149.63	ap	l	s	c	d	ab	j	w	...	0	0	0	0	0	0
1203	2396	160.87	j	o	as	f	d	ab	g	p	...	1	0	0	0	0	0
1205	2403	150.89	x	b	m	c	d	ab	j	j	...	0	0	1	0	0	0
1269	2511	152.32	s	aa	m	c	d	ab	g	g	...	1	0	0	0	0	0
1279	2531	139.08	x	b	as	c	d	ac	j	y	...	0	0	1	0	0	0
1349	2669	142.71	ak	l	ae	f	d	ac	i	v	...	0	0	0	0	0	0
1459	2903	167.45	ai	b	ae	a	d	ac	g	m	...	0	0	1	0	0	0
1730	3456	139.61	ak	v	ak	c	d	ae	a	x	...	1	0	0	0	0	0
2240	4481	154.16	w	n	as	f	d	k	j	r	...	1	0	0	0	0	0
2263	4530	136.96	ak	s	as	c	d	k	g	i	...	1	0	0	0	0	0
2348	4705	140.25	ay	i	as	c	d	n	j	k	...	0	0	1	0	0	0
2357	4722	142.71	a	v	k	c	d	n	j	e	...	0	1	0	0	0	0
2376	4762	148.94	w	s	as	c	d	n	h	w	...	0	0	1	0	0	0
2414	4847	136.56	ap	l	s	c	d	n	d	h	...	0	0	0	0	0	0
2470	4950	137.49	x	aa	as	c	d	n	j	r	...	1	0	0	0	0	0
2496	5000	137.09	ak	c	r	c	d	n	i	f	...	0	0	1	0	0	0
2735	5471	158.23	x	v	e	c	d	m	g	s	...	0	0	0	0	1	0
2736	5473	153.51	x	i	as	a	d	m	j	r	...	0	0	1	0	0	0

	ID	y	X0	X1	X2	X3	X4	X5	X6	X8	...	X375	X376	X377	X378	X379	X380
2852	5706	141.39	z	b	m	a	d	p	h	j	...	0	0	1	0	0	0
2887	5781	144.56	v	s	as	c	d	p	j	p	...	1	0	0	0	0	0
2888	5785	138.19	ak	s	as	c	d	p	g	m	...	1	0	0	0	0	0
2905	5820	147.22	ay	aa	as	c	d	p	j	f	...	1	0	0	0	0	0
2983	5979	139.16	j	i	as	c	d	q	l	r	...	0	0	1	0	0	0
3028	6078	140.31	ap	l	s	c	d	q	g	h	...	0	0	0	0	0	0
3090	6208	146.30	au	b	aa	c	d	q	c	n	...	0	0	1	0	0	0
3133	6273	165.52	aj	v	r	c	d	q	g	a	...	0	0	1	0	0	0
3177	6343	137.32	ap	l	s	c	d	q	j	h	...	0	0	0	0	0	0
3215	6422	141.09	at	f	ae	c	d	s	b	l	...	0	1	0	0	0	0
3442	6873	139.07	f	c	m	c	d	r	j	y	...	0	0	1	0	0	0
3744	7500	155.62	x	f	ak	c	d	v	d	d	...	1	0	0	0	0	0
3773	7559	136.75	w	s	as	c	d	v	g	a	...	1	0	0	0	0	0
3980	7980	142.46	w	s	as	a	d	w	j	k	...	0	1	0	0	0	0
4176	8344	149.52	ak	l	as	a	d	aa	j	r	...	0	0	1	0	0	0

50 rows × 378 columns

Since the number of unique values in the training and testing dataset are different therefore we need to merge the training and testing dataset and then convert the categorical columns to numeric columns and again separate the two.

Merging the two datasets and converting categorical to numerical columns

In [25]:
```
benz2= benz_train.append(benz_test, ignore_index=True)
benz2= pd.get_dummies(benz2)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\frame.py:6692: FutureW
arning: Sorting because non-concatenation axis is not aligned. A future versi
on
of pandas will change to not sort by default.

To accept the future behavior, pass 'sort=False'.

To retain the current behavior and silence the warning, pass 'sort=True'.

  sort=sort)
```

Separating the datasets(training and testing)

In [26]:
```
train, test= benz2[0:len(benz_train)], benz2[len(benz_test):]
```

In [27]:
```
train.shape, test.shape
```

Out[27]:
```
((4209, 581), (4209, 581))
```

In [28]:
```
train.head()
```

Out[28]:

	ID	X10	X100	X101	X102	X103	X104	X105	X106	X107	...	X8_p	X8_q	X8_r	X8_s	X
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	
1	6	0	1	1	0	0	0	0	0	0	...	0	0	0	0	
2	7	0	0	1	0	0	0	0	0	0	...	0	0	0	0	
3	9	0	0	1	0	0	0	0	0	0	...	0	0	0	0	
4	13	0	0	1	0	0	0	0	0	0	...	0	0	0	0	

5 rows × 581 columns

Separating features and labels

In [29]:
```
x_train1= train.drop(['y', 'ID'], axis=1)
y_train1= train['y']
x_test1= test.drop(['y', 'ID'], axis=1)
```

In [30]:
```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test= train_test_split(x_train1,
                                                   y_train1,
                                                   test_size= 0.25,
                                                   random_state= 4)
```

Random Forest

In [31]:
```
from sklearn.ensemble import RandomForestRegressor
RF= RandomForestRegressor(n_estimators= 15)
RF.fit(X_train, y_train)
RF.predict(X_test)
```

Out[31]:
```
array([ 95.14666667,  91.81066667, 113.352     , ..., 112.828     ,
        87.78266667, 105.16333333])
```

In [32]:
```
from sklearn.model_selection import cross_validate
scoring={'r2': 'r2',
        'MSE': 'neg_mean_squared_error'}
scores= cross_validate(estimator= RF,
                       X= x_train1,
                       y=y_train1,
                       cv= 10,
                       scoring= scoring,
                       return_train_score= True)

print("Training r2:{}".format(scores['train_r2'].mean()))
print("Testing r2:{}".format(scores['test_r2'].mean()))
print("Train MSE:{}".format(scores['train_MSE'].mean()))
print("Test MSE:{}".format(scores['test_MSE'].mean()))
```

```
Training r2:0.893533937656603
Testing r2:0.5098946542041107
Train MSE:-17.12054879810878
Test MSE:-81.46904837941155
```

In [33]:
```
from sklearn.model_selection import GridSearchCV
from sklearn.model_selection import KFold

scoring={'r2': 'r2',
        'MSE': 'neg_mean_squared_error'}

params= {'n_estimators': [5,10,15,20,25],
        'criterion': ['mse'],
        'max_depth': [2,4,6],
        'max_features': ['sqrt', 'log2'],
        'min_samples_split': [.05]}
kf= KFold(n_splits=10, random_state=20)
gs1= GridSearchCV(estimator= RF,
                 param_grid= params,
                 scoring= scoring,
                 refit='MSE',
                 cv= kf,
                 verbose=2)
```

In [34]:
```
gs1.fit(x_train1[:100], y_train1[:100])
```

```
Fitting 10 folds for each of 30 candidates, totalling 300 fits
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=5

[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent worke
rs.
[Parallel(n_jobs=1)]: Done   1 out of   1 | elapsed:    0.0s remaining:    0.
0s
```

```
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
```

```
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
```

```
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
```

```
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=10
```

```
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
```

```
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
```

```
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, 
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, 
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, 
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, 
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, 
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, 
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, 
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, 
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, 
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=2, max_features=log2, min_samples_split=0.05, 
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, 
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, 
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, 
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, 
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
```

```
_estimators=5
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
```

```
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
```

```
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
```

```
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
```

```
_estimators=10
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=15
```

```
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
```

```
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=4, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
```

```
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
```

```
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
```

```
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
```

```
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=sqrt, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=5
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=5, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
```

```
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=10
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=10, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
```

```
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=15
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=15, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=20
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=20
```

```
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=20, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s
[CV] criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05, n
_estimators=25
[CV]  criterion=mse, max_depth=6, max_features=log2, min_samples_split=0.05,
n_estimators=25, total=   0.0s

[Parallel(n_jobs=1)]: Done 300 out of 300 | elapsed:   16.7s finished
```

Out[34]:
```
GridSearchCV(cv=KFold(n_splits=10, random_state=20, shuffle=False),
       error_score='raise-deprecating',
       estimator=RandomForestRegressor(bootstrap=True, criterion='mse', max_d
epth=None,
           max_features='auto', max_leaf_nodes=None,
           min_impurity_decrease=0.0, min_impurity_split=None,
           min_samples_leaf=1, min_samples_split=2,
           min_weight_fraction_leaf=0.0, n_estimators=15, n_jobs=None,
           oob_score=False, random_state=None, verbose=0, warm_start=False),
       fit_params=None, iid='warn', n_jobs=None,
       param_grid={'n_estimators': [5, 10, 15, 20, 25], 'criterion': ['mse'],
'max_depth': [2, 4, 6], 'max_features': ['sqrt', 'log2'], 'min_samples_spli
t': [0.05]},
       pre_dispatch='2*n_jobs', refit='MSE', return_train_score='warn',
       scoring={'r2': 'r2', 'MSE': 'neg_mean_squared_error'}, verbose=2)
```

In [35]:
```
gs1.best_estimator_
gs1.best_score_
gs1.best_params_
```

Out[35]:
```
{'criterion': 'mse',
 'max_depth': 6,
 'max_features': 'sqrt',
 'min_samples_split': 0.05,
 'n_estimators': 25}
```

K-Nearest Neighbor

In [89]:
```
from sklearn.neighbors import KNeighborsRegressor
knn= KNeighborsRegressor(n_neighbors=13, metric='hamming', weights= 'distance'
)
knn.fit(x_train1, y_train1)
```

Out[89]:
```
KNeighborsRegressor(algorithm='auto', leaf_size=30, metric='hamming',
          metric_params=None, n_jobs=None, n_neighbors=13, p=2,
          weights='distance')
```

In [90]:
```
neighbors= [3,5,7,9,11,13,15,17,19]
metric=['hamming']
weights=['uniform', 'distance']
```

In [91]:
```
params= dict(n_neighbors= neighbors,
            metric= metric,
             weights=weights
            )
```

In [84]:
```
kf= KFold(n_splits=10, random_state=10)
gs2= GridSearchCV(estimator= knn,
                 param_grid= params,
                 scoring='r2',
                  cv= kf,
                 verbose=2)
```

In [87]:
```
gs2.fit(x_train1[:500], y_train1[:500])
```

```
Fitting 10 folds for each of 18 candidates, totalling 180 fits
[CV] metric=hamming, n_neighbors=3, weights=uniform ..................

[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent worke
rs.

[CV] ... metric=hamming, n_neighbors=3, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=3, weights=uniform ..................

[Parallel(n_jobs=1)]: Done   1 out of   1 | elapsed:    0.5s remaining:    0.
0s
```

```
[CV] ... metric=hamming, n_neighbors=3, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=3, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=3, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=3, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=3, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=3, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=3, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=3, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=3, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=3, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=3, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=3, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=3, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=3, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=3, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=3, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=3, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=3, weights=distance ................
[CV] .. metric=hamming, n_neighbors=3, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=3, weights=distance ................
[CV] .. metric=hamming, n_neighbors=3, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=3, weights=distance ................
[CV] .. metric=hamming, n_neighbors=3, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=3, weights=distance ................
[CV] .. metric=hamming, n_neighbors=3, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=3, weights=distance ................
[CV] .. metric=hamming, n_neighbors=3, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=3, weights=distance ................
[CV] .. metric=hamming, n_neighbors=3, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=3, weights=distance ................
[CV] .. metric=hamming, n_neighbors=3, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=3, weights=distance ................
[CV] .. metric=hamming, n_neighbors=3, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=3, weights=distance ................
[CV] .. metric=hamming, n_neighbors=3, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=3, weights=distance ................
[CV] .. metric=hamming, n_neighbors=3, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=5, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=5, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=5, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=5, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=5, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=5, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=5, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=5, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=5, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=5, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=5, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=5, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=5, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=5, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=5, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=5, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=5, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=5, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=5, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=5, weights=uniform, total=   0.0s
```

```
[CV] metric=hamming, n_neighbors=5, weights=distance .................
[CV] .. metric=hamming, n_neighbors=5, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=5, weights=distance .................
[CV] .. metric=hamming, n_neighbors=5, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=5, weights=distance .................
[CV] .. metric=hamming, n_neighbors=5, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=5, weights=distance .................
[CV] .. metric=hamming, n_neighbors=5, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=5, weights=distance .................
[CV] .. metric=hamming, n_neighbors=5, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=5, weights=distance .................
[CV] .. metric=hamming, n_neighbors=5, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=5, weights=distance .................
[CV] .. metric=hamming, n_neighbors=5, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=5, weights=distance .................
[CV] .. metric=hamming, n_neighbors=5, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=5, weights=distance .................
[CV] .. metric=hamming, n_neighbors=5, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=5, weights=distance .................
[CV] .. metric=hamming, n_neighbors=5, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=uniform ..................
[CV] ... metric=hamming, n_neighbors=7, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=uniform ..................
[CV] ... metric=hamming, n_neighbors=7, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=uniform ..................
[CV] ... metric=hamming, n_neighbors=7, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=uniform ..................
[CV] ... metric=hamming, n_neighbors=7, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=uniform ..................
[CV] ... metric=hamming, n_neighbors=7, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=uniform ..................
[CV] ... metric=hamming, n_neighbors=7, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=uniform ..................
[CV] ... metric=hamming, n_neighbors=7, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=uniform ..................
[CV] ... metric=hamming, n_neighbors=7, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=uniform ..................
[CV] ... metric=hamming, n_neighbors=7, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=uniform ..................
[CV] ... metric=hamming, n_neighbors=7, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=distance .................
[CV] .. metric=hamming, n_neighbors=7, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=distance .................
[CV] .. metric=hamming, n_neighbors=7, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=distance .................
[CV] .. metric=hamming, n_neighbors=7, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=distance .................
[CV] .. metric=hamming, n_neighbors=7, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=distance .................
[CV] .. metric=hamming, n_neighbors=7, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=distance .................
[CV] .. metric=hamming, n_neighbors=7, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=distance .................
[CV] .. metric=hamming, n_neighbors=7, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=distance .................
[CV] .. metric=hamming, n_neighbors=7, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=distance .................
```

```
[CV] .. metric=hamming, n_neighbors=7, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=7, weights=distance ................
[CV] .. metric=hamming, n_neighbors=7, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=9, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=9, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=9, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=9, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=9, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=9, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=9, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=9, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=9, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=uniform .................
[CV] ... metric=hamming, n_neighbors=9, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=distance ................
[CV] .. metric=hamming, n_neighbors=9, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=distance ................
[CV] .. metric=hamming, n_neighbors=9, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=distance ................
[CV] .. metric=hamming, n_neighbors=9, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=distance ................
[CV] .. metric=hamming, n_neighbors=9, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=distance ................
[CV] .. metric=hamming, n_neighbors=9, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=distance ................
[CV] .. metric=hamming, n_neighbors=9, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=distance ................
[CV] .. metric=hamming, n_neighbors=9, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=distance ................
[CV] .. metric=hamming, n_neighbors=9, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=distance ................
[CV] .. metric=hamming, n_neighbors=9, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=9, weights=distance ................
[CV] .. metric=hamming, n_neighbors=9, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=11, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=11, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=11, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=11, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=11, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=11, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=11, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=11, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=11, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=11, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=11, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=11, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=11, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=11, weights=uniform, total=   0.0s
```

```
[CV] metric=hamming, n_neighbors=11, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=11, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=11, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=11, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=11, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=11, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=11, weights=distance ...............
[CV] . metric=hamming, n_neighbors=11, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=11, weights=distance ...............
[CV] . metric=hamming, n_neighbors=11, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=11, weights=distance ...............
[CV] . metric=hamming, n_neighbors=11, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=11, weights=distance ...............
[CV] . metric=hamming, n_neighbors=11, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=11, weights=distance ...............
[CV] . metric=hamming, n_neighbors=11, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=11, weights=distance ...............
[CV] . metric=hamming, n_neighbors=11, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=11, weights=distance ...............
[CV] . metric=hamming, n_neighbors=11, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=11, weights=distance ...............
[CV] . metric=hamming, n_neighbors=11, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=11, weights=distance ...............
[CV] . metric=hamming, n_neighbors=11, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=11, weights=distance ...............
[CV] . metric=hamming, n_neighbors=11, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=13, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=13, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=13, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=13, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=13, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=13, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=13, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=13, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=13, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=uniform ................
[CV] .. metric=hamming, n_neighbors=13, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=distance ...............
[CV] . metric=hamming, n_neighbors=13, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=distance ...............
[CV] . metric=hamming, n_neighbors=13, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=distance ...............
[CV] . metric=hamming, n_neighbors=13, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=distance ...............
[CV] . metric=hamming, n_neighbors=13, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=distance ...............
[CV] . metric=hamming, n_neighbors=13, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=distance ...............
```

```
[CV] . metric=hamming, n_neighbors=13, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=distance ................
[CV] . metric=hamming, n_neighbors=13, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=distance ................
[CV] . metric=hamming, n_neighbors=13, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=distance ................
[CV] . metric=hamming, n_neighbors=13, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=13, weights=distance ................
[CV] . metric=hamming, n_neighbors=13, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=15, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=15, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=15, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=15, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=15, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=15, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=15, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=15, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=15, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=15, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=distance ................
[CV] . metric=hamming, n_neighbors=15, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=distance ................
[CV] . metric=hamming, n_neighbors=15, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=distance ................
[CV] . metric=hamming, n_neighbors=15, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=distance ................
[CV] . metric=hamming, n_neighbors=15, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=distance ................
[CV] . metric=hamming, n_neighbors=15, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=distance ................
[CV] . metric=hamming, n_neighbors=15, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=distance ................
[CV] . metric=hamming, n_neighbors=15, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=distance ................
[CV] . metric=hamming, n_neighbors=15, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=distance ................
[CV] . metric=hamming, n_neighbors=15, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=15, weights=distance ................
[CV] . metric=hamming, n_neighbors=15, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=17, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=17, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=17, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=17, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=17, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=17, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=17, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=17, weights=uniform, total=   0.0s
```

```
[CV] metric=hamming, n_neighbors=17, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=17, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=17, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=17, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=17, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=17, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=17, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=17, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=17, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=17, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=17, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=17, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=17, weights=distance ................
[CV] . metric=hamming, n_neighbors=17, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=17, weights=distance ................
[CV] . metric=hamming, n_neighbors=17, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=17, weights=distance ................
[CV] . metric=hamming, n_neighbors=17, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=17, weights=distance ................
[CV] . metric=hamming, n_neighbors=17, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=17, weights=distance ................
[CV] . metric=hamming, n_neighbors=17, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=17, weights=distance ................
[CV] . metric=hamming, n_neighbors=17, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=17, weights=distance ................
[CV] . metric=hamming, n_neighbors=17, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=17, weights=distance ................
[CV] . metric=hamming, n_neighbors=17, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=17, weights=distance ................
[CV] . metric=hamming, n_neighbors=17, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=17, weights=distance ................
[CV] . metric=hamming, n_neighbors=17, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=19, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=19, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=19, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=19, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=19, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=19, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=19, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=19, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=19, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=uniform .................
[CV] .. metric=hamming, n_neighbors=19, weights=uniform, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=distance ................
[CV] . metric=hamming, n_neighbors=19, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=distance ................
[CV] . metric=hamming, n_neighbors=19, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=distance ................
```

```
[CV] . metric=hamming, n_neighbors=19, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=distance ................
[CV] . metric=hamming, n_neighbors=19, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=distance ................
[CV] . metric=hamming, n_neighbors=19, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=distance ................
[CV] . metric=hamming, n_neighbors=19, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=distance ................
[CV] . metric=hamming, n_neighbors=19, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=distance ................
[CV] . metric=hamming, n_neighbors=19, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=distance ................
[CV] . metric=hamming, n_neighbors=19, weights=distance, total=   0.0s
[CV] metric=hamming, n_neighbors=19, weights=distance ................
[CV] . metric=hamming, n_neighbors=19, weights=distance, total=   0.0s

[Parallel(n_jobs=1)]: Done 180 out of 180 | elapsed:  1.8min finished
```

Out[87]:
```
GridSearchCV(cv=KFold(n_splits=10, random_state=10, shuffle=False),
       error_score='raise-deprecating',
       estimator=KNeighborsRegressor(algorithm='auto', leaf_size=30, metric
='hamming',
          metric_params=None, n_jobs=None, n_neighbors=5, p=2,
          weights='uniform'),
       fit_params=None, iid='warn', n_jobs=None,
       param_grid={'n_neighbors': [3, 5, 7, 9, 11, 13, 15, 17, 19], 'metric':
['hamming'], 'weights': ['uniform', 'distance']},
       pre_dispatch='2*n_jobs', refit=True, return_train_score='warn',
       scoring='r2', verbose=2)
```

In [88]:
```
gs2.best_estimator_
gs2.best_score_
gs2.best_params_
```

Out[88]:
```
{'metric': 'hamming', 'n_neighbors': 13, 'weights': 'distance'}
```

In [92]:
```
y_pred= knn.predict(X_train)
print("Train r2:", metrics.r2_score(y_train, y_pred))
print("Train mse:", metrics.mean_squared_error(y_train, y_pred))

y_pred= knn.predict(X_test)
print("\n")
print("Test r2:", metrics.r2_score(y_test, y_pred))
print("Test mse:", metrics.mean_squared_error(y_test, y_pred))
```

```
Train r2: 0.9692400425176216
Train mse: 5.008564558525219


Test r2: 0.9783220832423499
Test mse: 3.3478110869556015
```

Adaboost

In [76]:
```
from sklearn.ensemble import AdaBoostRegressor
ABR= AdaBoostRegressor(n_estimators=500,
                      learning_rate=.01)
```

In [77]:
```
ABR.fit(X_train, y_train)
```

Out[77]:
```
AdaBoostRegressor(base_estimator=None, learning_rate=0.01, loss='linear',
         n_estimators=500, random_state=None)
```

In [78]:
```
from sklearn import metrics
y_pred= ABR.predict(X_train)
print("Train r2:", metrics.r2_score(y_train, y_pred))
print("Train mse:", metrics.mean_squared_error(y_train, y_pred))

y_pred= ABR.predict(X_test)
print("\n")
print("Test r2:", metrics.r2_score(y_test, y_pred))
print("Test mse:", metrics.mean_squared_error(y_test, y_pred))
```

```
Train r2: 0.607669286860829
Train mse: 63.88219834749393


Test r2: 0.5462656503921461
Test mse: 70.07208779015666
```

Gradient Boosting

In [67]:
```
from sklearn.ensemble import GradientBoostingRegressor
params={'n_estimators': 600,
       'max_depth': 6,
       'min_samples_split': 3,
       'learning_rate': 0.01}

GBR= GradientBoostingRegressor(**params)
```

In [68]:
```
GBR.fit(X_train, y_train)
```

Out[68]:
```
GradientBoostingRegressor(alpha=0.9, criterion='friedman_mse', init=None,
             learning_rate=0.01, loss='ls', max_depth=6, max_features=None,
             max_leaf_nodes=None, min_impurity_decrease=0.0,
             min_impurity_split=None, min_samples_leaf=1,
             min_samples_split=3, min_weight_fraction_leaf=0.0,
             n_estimators=600, n_iter_no_change=None, presort='auto',
             random_state=None, subsample=1.0, tol=0.0001,
             validation_fraction=0.1, verbose=0, warm_start=False)
```

In [69]:
```
y_pred= GBR.predict(X_train)
print("Train mse:", metrics.mean_squared_error(y_train, y_pred))
print("Train r2:", metrics.r2_score(y_train, y_pred))

y_pred= GBR.predict(X_test)
print("\n")
print("Test mse:", metrics.mean_squared_error(y_test, y_pred))
print("Test r2:", metrics.r2_score(y_test, y_pred))
```

```
Train mse: 40.11874414289443
Train r2: 0.7536118682984057


Test mse: 69.75828176328316
Test r2: 0.5482976231504413
```

Xg boost

In [44]:
```
import xgboost as xgb
btrain= xgb.DMatrix(X_train, y_train)
btest= xgb.DMatrix(X_test)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\xgboost\core.py:587: FutureWarnin
g: Series.base is deprecated and will be removed in a future version
  if getattr(data, 'base', None) is not None and \
```

In [45]:
```
xgb_params={'n_trees': 500,
         'eta': .01,
        'learning_rate': .01,
        'max_depth': 4,
        'subsample': 0.5,
         'objective': 'reg:squarederror',
         'eval_metric': 'rmse',
         'silent':1
        }
```

In [46]:
```
cv_results= xgb.cv(xgb_params,
                   btrain,
                    num_boost_round= 500,
                   verbose_eval=50,
                   show_stdv= False)
```

```
[0]     train-rmse:99.9976      test-rmse:99.996
[50]    train-rmse:61.0124      test-rmse:61.0445
[100]   train-rmse:37.6058      test-rmse:37.6417
[150]   train-rmse:23.7622      test-rmse:23.8258
[200]   train-rmse:15.8105      test-rmse:16.0004
[250]   train-rmse:11.4676      test-rmse:11.8719
[300]   train-rmse:9.28282      test-rmse:9.93559
[350]   train-rmse:8.24186      test-rmse:9.14331
[400]   train-rmse:7.74004      test-rmse:8.87874
[450]   train-rmse:7.47607      test-rmse:8.80487
[499]   train-rmse:7.31012      test-rmse:8.80651
```

In [47]:
```
model= xgb.train(dict(xgb_params, silent=0),
                 btrain,
                 num_boost_round= len(cv_results))
```

In [48]:
```
from sklearn import metrics
y_pred= model.predict(btrain)
print("Train mse:", metrics.mean_squared_error(y_train, y_pred))
print("Train r2:", metrics.r2_score(y_train, y_pred))

y_pred= model.predict(btest)
print("\n")
print("Test mse:", metrics.mean_squared_error(y_test, y_pred))
print("Test r2:", metrics.r2_score(y_test, y_pred))
```

```
Train mse: 58.77134180119759
Train r2: 0.639057467691007


Test mse: 66.70954459351108
Test r2: 0.5680389612563419
```

In [ ]: