

```

import pandas as pd
from sklearn import metrics
from sklearn.model_selection import train_test_split

from sklearn.ensemble import RandomForestRegressor
from sklearn.linear_model import LinearRegression
from sklearn.tree import DecisionTreeRegressor

from sklearn.metrics import classification_report
import pickle

df = pd.read_csv('energy_efficiency_data.csv')
df.head()

```

	Relative_Compactness	Surface_Area	Wall_Area	Roof_Area	Overall_Height	Orienta
0	0.98	514.5	294.0	110.25	7.0	
1	0.98	514.5	294.0	110.25	7.0	
2	0.98	514.5	294.0	110.25	7.0	
3	0.98	514.5	294.0	110.25	7.0	
4	0.90	563.5	318.5	122.50	7.0	



## ▼ New Section

```

x = df.iloc[:, :-2]
y = df.iloc[:, lambda df: [-2, -1]]

x_train, x_test, y_train, y_test = train_test_split(x,y, test_size=0.2)

rf = RandomForestRegressor()
rf.fit(x_train,y_train)
y_pred = rf.predict(x_test)

rf.score(x_test,y_test)

0.9781272142241262

lr = LinearRegression()
lr.fit(x_train,y_train)
y_pred = lr.predict(x_test)

```

```
lr.score(x_test,y_test)
```

```
0.8877247182819802
```

```
dt = DecisionTreeRegressor()
dt.fit(x_train,y_train)
y_pred= dt.predict(x_test)
```

```
dt.score(x_test,y_test)
```

```
0.9596553075915775
```

```
Acc = pd.DataFrame(index=None,
                    columns=['model','train_y', 'test_y'])
regressors = [ ["RFR", RandomForestRegressor()],
                ["DTR", DecisionTreeRegressor()],
                ["LR", LinearRegression()]
              ]
```

```
from sklearn.metrics import r2_score
```

```
for mod in regressors:
```

```
    name = mod[0]
```


```
    model = mod[1]
```

```
    model.fit(x_train,y_train)
```

```
    actr1 = r2_score(y_train, model.predict(x_train))
```

```
    acte1 = r2_score(y_test, model.predict(x_test))
```

```
    Acc = Acc.append(pd.Series({'model':name, 'train_y':actr1,'test_y':acte1}),ignore_index=True)
Acc.sort_values(by='test_y', ascending=False)
```

	model	train_y	test_y	
0	RFR	0.996924	0.978295	
1	DTR	1.000000	0.959642	
2	LR	0.903991	0.887725	

```
filename = 'randomforest.sav'
```

```
pickle.dump(rf,open(filename, 'wb'))
```

```
filename = 'decisiontree.sav'
```

```
pickle.dump(dt,open(filename, 'wb'))
```

```
filename = 'linearregression.sav'
```

```
pickle.dump(lr,open(filename, 'wb'))
```

---

✓ 0s completed at 9:36 AM ● ✕