

```
In [1]: import pandas as pd
import numpy as np
```

```
In [2]: dataset=pd.read_csv("C://Users//ALWAYS RAMESH//Downloads//insurance.csv")
```

```
In [3]: dataset.head(3)
```

```
Out[3]:
```

	age	sex	bmi	children	smoker	region	charges
0	19	female	27.90	0	yes	southwest	16884.9240
1	18	male	33.77	1	no	southeast	1725.5523
2	28	male	33.00	3	no	southeast	4449.4620

```
In [4]: from sklearn.preprocessing import LabelEncoder
```

```
In [5]: le=LabelEncoder()
dataset['sex']=le.fit_transform(dataset['sex'])
dataset['smoker']=le.fit_transform(dataset['smoker'])
#dataset['region']=le.fit_transform(dataset['region'])
```

```
In [6]: dataset.head(3)
```

```
Out[6]:
```

	age	sex	bmi	children	smoker	region	charges
0	19	0	27.90	0	1	southwest	16884.9240
1	18	1	33.77	1	0	southeast	1725.5523
2	28	1	33.00	3	0	southeast	4449.4620

```
In [7]: x=dataset.iloc[:, :-2]
```

```
In [8]: y=dataset['region']
```

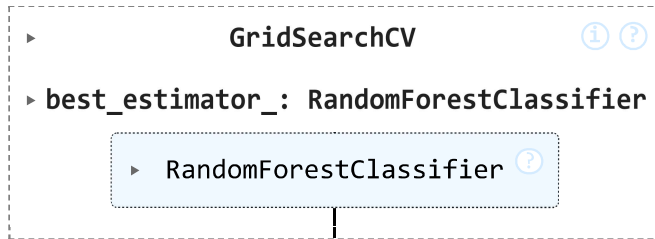
```
In [9]: from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=0)
```

```
In [32]: from sklearn.ensemble import RandomForestClassifier
li=RandomForestClassifier()
```

```
In [36]: forest_params=[{'n_estimators':[50,75,100], 'max_depth':list(range(10,15)), 'max_feat
```

```
In [37]: from sklearn.model_selection import GridSearchCV
model= GridSearchCV(estimator=li, param_grid=forest_params, cv=5, scoring='accuracy')
model.fit(x_train,y_train)
```

Out[37]:



In [ ]:

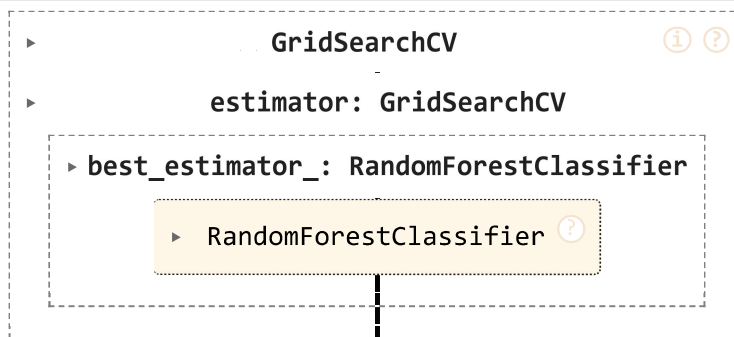
```

In [52]: #clf=GridSearchCV(estimator=li,forest_params,cv=10,scoring='accuracy')
#clf = GridSearchCV(estimator=rf, param_grid=param_grid, cv=5, scoring='accuracy')
clf = GridSearchCV(estimator=li, param_grid=forest_params, cv=10, scoring='accuracy')

```

In [39]: `clf`

Out[39]:



```

In [49]: clf.fit(x_train, y_train)
print(clf.best_params_)
print(clf.best_score_)

```

```

{'max_depth': 10, 'max_features': 'sqrt', 'n_estimators': 75}
0.35467856325783575

```

```

In [50]: print("Best Parameters:", clf.best_params_)
print("Best Score:", clf.best_score_)

```

```

Best Parameters: {'max_depth': 10, 'max_features': 'sqrt', 'n_estimators': 75}
Best Score: 0.35467856325783575

```

```

In [57]: y_pred=model.predict(x_test)

```

```

In [58]: from sklearn.metrics import accuracy_score
print(accuracy_score(y_test,y_pred))

```

```

0.2960199004975124

```

In [ ]: