In [1]:
```
import pandas as pd
import pickle
import warnings
warnings.filterwarnings("ignore")
```

In [2]:
```
a=pd.read_csv("Health_insurance.csv")
```

In [3]:
```
a
```

Out[3]:

	age	sex	bmi	children	smoker	region	charges
0	19	female	27.900	0	yes	southwest	16884.92400
1	18	male	33.770	1	no	southeast	1725.55230
2	28	male	33.000	3	no	southeast	4449.46200
3	33	male	22.705	0	no	northwest	21984.47061
4	32	male	28.880	0	no	northwest	3866.85520
...	...	...	...	...	...	...	...
1333	50	male	30.970	3	no	northwest	10600.54830
1334	18	female	31.920	0	no	northeast	2205.98080
1335	18	female	36.850	0	no	southeast	1629.83350
1336	21	female	25.800	0	no	southwest	2007.94500
1337	61	female	29.070	0	yes	northwest	29141.36030

1338 rows × 7 columns

In [4]:
```
a.shape
```

Out[4]:
```
(1338, 7)
```

In [5]:
```
a.describe()
```

Out[5]:

	age	bmi	children	charges
count	1338.000000	1338.000000	1338.000000	1338.000000
mean	39.207025	30.663397	1.094918	13270.422265
std	14.049960	6.098187	1.205493	12110.011237
min	18.000000	15.960000	0.000000	1121.873900
25%	27.000000	26.296250	0.000000	4740.287150
50%	39.000000	30.400000	1.000000	9382.033000
75%	51.000000	34.693750	2.000000	16639.912515
max	64.000000	53.130000	5.000000	63770.428010

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

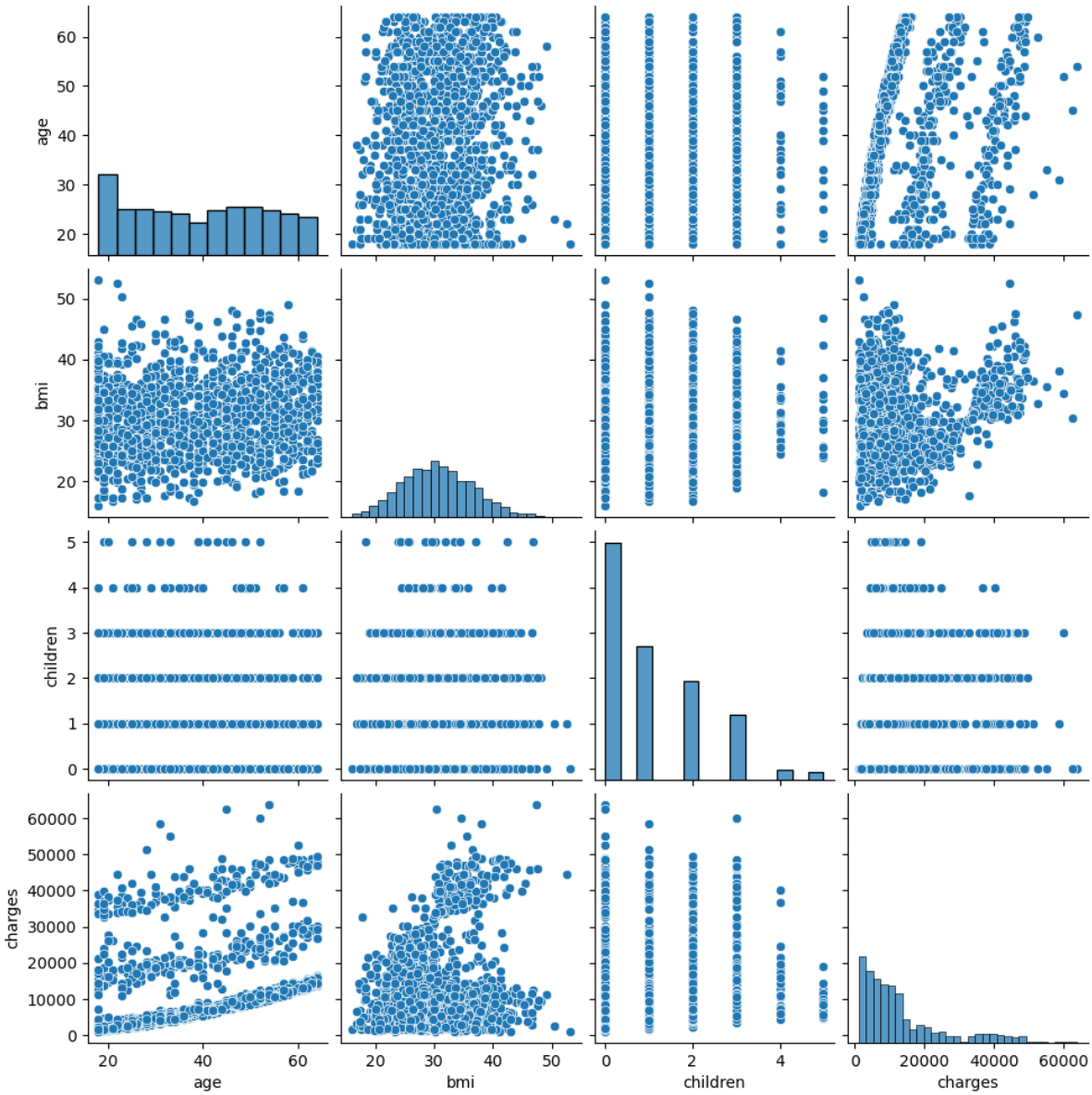
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1338 entries, 0 to 1337
Data columns (total 7 columns):
 #   Column    Non-Null Count  Dtype  
---  ------    --------------  -----  
 0   age       1338 non-null   int64  
 1   sex       1338 non-null   object 
 2   bmi       1338 non-null   float64
 3   children  1338 non-null   int64  
 4   smoker    1338 non-null   object 
 5   region    1338 non-null   object 
 6   charges   1338 non-null   float64
dtypes: float64(2), int64(2), object(3)
memory usage: 73.3+ KB
```

In [7]:
```
a.isnull().sum()
```

Out[7]:
```
age         0
sex         0
bmi         0
children    0
smoker      0
region      0
charges     0
dtype: int64
```

In [8]:
```
import seaborn as sns
import matplotlib.pyplot as plt
```

In [9]:
```
sns.pairplot(a)
plt.show()
```



In [10]:
```
a['sex']=a['sex'].map({'male':1,'female':0})
```

In [11]:
```
a['sex'].unique()
```

Out[11]:
```
array([0, 1], dtype=int64)
```

In [12]:
```
a['smoker']=a['smoker'].map({'yes':1,'no':0})
```

In [13]:
```
a['smoker'].unique()
```

Out[13]:
```
array([1, 0], dtype=int64)
```

In [14]:
```
data=pd.get_dummies(a,dtype=int)
```

In [15]:
```
data.shape
```

Out[15]:
```
(1338, 10)
```

In [16]:
```
data.head(10)
```

Out[16]:

	age	sex	bmi	children	smoker	charges	region_northeast	region_northwest	region_so
0	19	0	27.900	0	1	16884.92400	0	0	
1	18	1	33.770	1	0	1725.55230	0	0	
2	28	1	33.000	3	0	4449.46200	0	0	
3	33	1	22.705	0	0	21984.47061	0	1	
4	32	1	28.880	0	0	3866.85520	0	1	
5	31	0	25.740	0	0	3756.62160	0	0	
6	46	0	33.440	1	0	8240.58960	0	0	
7	37	0	27.740	3	0	7281.50560	0	1	
8	37	1	29.830	2	0	6406.41070	1	0	
9	60	0	25.840	0	0	28923.13692	0	1	

In [17]:
```
y =data['charges']
x =data.drop('charges',axis=1)
```

In [18]:
```
x.head(5)
```

Out[18]:

	age	sex	bmi	children	smoker	region_northeast	region_northwest	region_southeast	regi
0	19	0	27.900	0	1	0	0	0	
1	18	1	33.770	1	0	0	0	1	
2	28	1	33.000	3	0	0	0	1	
3	33	1	22.705	0	0	0	1	0	
4	32	1	28.880	0	0	0	1	0	

In [19]:
```
y.head(5)
```

Out[19]:
```
0    16884.92400
1     1725.55230
2     4449.46200
3    21984.47061
4     3866.85520
Name: charges, dtype: float64
```

In [20]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.33,random_state=42)
```

In [21]:
```
x_train
```

Out[21]:

	age	sex	bmi	children	smoker	region_northeast	region_northwest	region_southeast	r
1046	43	0	25.080	0	0	1	0	0	
682	39	1	35.300	2	1	0	0	0	
1037	45	0	30.495	1	1	0	1	0	
490	19	0	32.900	0	0	0	0	0	
39	60	1	39.900	0	1	0	0	0	
...	...	...	...	...	...	...	...	...	
1095	18	0	31.350	4	0	1	0	0	
1130	39	0	23.870	5	0	0	0	1	
1294	58	1	25.175	0	0	1	0	0	
860	37	0	47.600	2	1	0	0	0	
1126	55	1	29.900	0	0	0	0	0	

896 rows × 9 columns

In [22]:
```
y_train
```

Out[22]:
```
1046     7325.04820
682     40103.89000
1037    39725.51805
490      1748.77400
39      48173.36100
           ...
1095     4561.18850
1130     8582.30230
1294    11931.12525
860     46113.51100
1126    10214.63600
Name: charges, Length: 896, dtype: float64
```

In [23]:
```
#from sklearn.model_selection import GridSearchCV #GridSearchCV is for parameter tu
#from sklearn.ensemble import RandomForestRegressor
#reg=RandomForestRegressor()
#n_estimators=[25,50,75,100,125,150,175,200] #number of decision trees in the fores
#criterion=['mse'] #criteria for choosing nodes default = 'gini'
#max_depth=[3,5,10] #maximum number of nodes in a tree default = None (it will go t
#parameters={'n_estimators': n_estimators,'criterion':criterion,'max_depth':max_dep
#RFC_reg = GridSearchCV(reg, parameters)
#RFC_reg.fit(x_train,y_train)
```

Further changes may come in it

In [24]:
```
from sklearn.ensemble import RandomForestRegressor
from sklearn.datasets import make_regression
x, y = make_regression(n_features=4, n_informative=2,random_state=0, shuffle=False)
regr = RandomForestRegressor(max_depth=2, random_state=0)
regr.fit(x_train, y_train)
```

Out[24]:

```
RandomForestRegressor(max_depth=2, random_state=0)
```

In [25]:

```
ypred=regr.predict(x_test)
```

In [26]:

```
ypred
```

Out[26]:
```
array([ 8718.82463052,  5618.90635056, 21297.00147926,  8718.82463052,
       41675.0494835 ,  5618.90635056,  5618.90635056, 12937.3401415 ,
        5618.90635056, 12865.4786582 , 21297.00147926,  5618.90635056,
        5618.90635056, 41675.0494835 , 41675.0494835 , 41675.0494835 ,
       12865.4786582 , 41675.0494835 ,  8718.82463052, 21297.00147926,
        5618.90635056,  7272.90208529,  5618.90635056,  5618.90635056,
       12937.3401415 , 12937.3401415 , 12937.3401415 ,  5618.90635056,
       12118.2299179 ,  5618.90635056,  5618.90635056, 12937.3401415 ,
        5618.90635056,  5618.90635056,  5618.90635056,  8718.82463052,
        5618.90635056,  7272.90208529, 21297.00147926, 41675.0494835 ,
        5618.90635056,  5618.90635056, 12937.3401415 , 12937.3401415 ,
        5618.90635056, 12937.3401415 ,  5618.90635056,  5618.90635056,
       41675.0494835 ,  5618.90635056, 12937.3401415 ,  5618.90635056,
        5618.90635056,  5618.90635056, 12937.3401415 , 12937.3401415 ,
        5618.90635056, 41675.0494835 , 12937.3401415 , 12937.3401415 ,
       12937.3401415 ,  5618.90635056, 12937.3401415 ,  5618.90635056,
       12937.3401415 ,  5618.90635056, 21297.00147926, 12937.3401415 ,
        5618.90635056,  5618.90635056,  5618.90635056,  8718.82463052,
       12865.4786582 ,  5618.90635056,  8718.82463052,  5618.90635056,
        5618.90635056, 12937.3401415 ,  5618.90635056,  8718.82463052,
        5618.90635056, 21297.00147926,  5618.90635056, 41675.0494835 ,
       41675.0494835 , 41675.0494835 ,  5618.90635056, 12937.3401415 ,
        7272.90208529, 12937.3401415 , 12937.3401415 , 21297.00147926,
       21297.00147926,  5618.90635056, 41675.0494835 ,  5618.90635056,
       21297.00147926,  5618.90635056, 21297.00147926,  5618.90635056,
        5618.90635056,  5618.90635056,  5618.90635056, 12937.3401415 ,
       12937.3401415 ,  5618.90635056,  8718.82463052, 21297.00147926,
        5618.90635056, 21297.00147926,  5618.90635056,  5618.90635056,
       12937.3401415 , 41675.0494835 , 12865.4786582 ,  5618.90635056,
       12937.3401415 , 21297.00147926,  5618.90635056,  5618.90635056,
        5618.90635056,  5618.90635056, 12937.3401415 ,  5618.90635056,
        5618.90635056,  5618.90635056,  7272.90208529, 12865.4786582 ,
       12937.3401415 ,  5618.90635056,  5618.90635056,  5618.90635056,
        5618.90635056,  8718.82463052,  5618.90635056, 12937.3401415 ,
        5618.90635056, 33977.85199676, 41675.0494835 , 41675.0494835 ,
        5618.90635056, 12937.3401415 ,  5618.90635056, 12937.3401415 ,
        5618.90635056, 21297.00147926,  5618.90635056,  5618.90635056,
       12937.3401415 ,  5618.90635056, 41675.0494835 ,  5618.90635056,
        5618.90635056, 41675.0494835 ,  5618.90635056,  5618.90635056,
       12937.3401415 , 12118.2299179 , 41675.0494835 , 41675.0494835 ,
       12937.3401415 ,  5618.90635056, 12937.3401415 ,  5618.90635056,
        5618.90635056,  5618.90635056, 41675.0494835 , 41675.0494835 ,
       41675.0494835 ,  5618.90635056, 12865.4786582 ,  5618.90635056,
        5618.90635056,  5618.90635056,  5618.90635056, 21297.00147926,
       21297.00147926, 12937.3401415 , 21297.00147926, 12937.3401415 ,
       21297.00147926,  5618.90635056, 12118.2299179 ,  5618.90635056,
        5618.90635056,  5618.90635056,  5618.90635056,  5618.90635056,
       12118.2299179 , 12937.3401415 ,  5618.90635056,  5618.90635056,
        5618.90635056, 41675.0494835 , 12937.3401415 , 12937.3401415 ,
        5618.90635056, 12937.3401415 ,  5618.90635056, 12937.3401415 ,
        5618.90635056, 21297.00147926,  5618.90635056,  5618.90635056,
       21297.00147926, 21297.00147926, 12937.3401415 ,  5618.90635056,
       12865.4786582 ,  5618.90635056, 12937.3401415 , 12937.3401415 ,
        8718.82463052, 21297.00147926,  5618.90635056,  5618.90635056,
        5618.90635056, 12937.3401415 , 12937.3401415 ,  5618.90635056,
        5618.90635056,  8718.82463052,  5618.90635056, 41675.0494835 ,
        5618.90635056, 41675.0494835 ,  5618.90635056,  5618.90635056,
       12937.3401415 , 12865.4786582 ,  5618.90635056, 12937.3401415 ,
        5618.90635056, 21297.00147926, 12937.3401415 , 12865.4786582 ,
        5618.90635056,  5618.90635056, 41675.0494835 ,  5618.90635056,
       12937.3401415 , 41675.0494835 ,  5618.90635056,  5618.90635056,
        5618.90635056,  5618.90635056,  5618.90635056,  5618.90635056,
       12937.3401415 ,  5618.90635056,  5618.90635056,  5618.90635056,
        5618.90635056, 12937.3401415 ,  5618.90635056,  5618.90635056,
```

```
12937.3401415 ,  5618.90635056, 12865.4786582 ,  5618.90635056,
 5618.90635056, 12937.3401415 , 21297.00147926, 41675.0494835 ,
12937.3401415 ,  5618.90635056, 41675.0494835 , 12937.3401415 ,
 8718.82463052, 12118.2299179 , 12937.3401415 ,  5618.90635056,
12937.3401415 , 12937.3401415 ,  5618.90635056, 21297.00147926,
 5618.90635056, 41675.0494835 ,  5618.90635056, 12937.3401415 ,
12865.4786582 , 12937.3401415 ,  8718.82463052, 12937.3401415 ,
41675.0494835 ,  5618.90635056, 12937.3401415 , 41675.0494835 ,
 5618.90635056,  5618.90635056,  5618.90635056, 12937.3401415 ,
21297.00147926, 12865.4786582 ,  8718.82463052, 12937.3401415 ,
12865.4786582 ,  5618.90635056,  5618.90635056, 41675.0494835 ,
41675.0494835 ,  5618.90635056,  5618.90635056, 21297.00147926,
12937.3401415 , 41675.0494835 ,  5618.90635056, 41675.0494835 ,
 5618.90635056,  5618.90635056,  5618.90635056,  5618.90635056,
21297.00147926, 12937.3401415 ,  5618.90635056, 12937.3401415 ,
12937.3401415 , 12937.3401415 , 41675.0494835 , 12937.3401415 ,
21297.00147926,  5618.90635056, 12937.3401415 ,  5618.90635056,
 8718.82463052,  8718.82463052, 12937.3401415 , 12937.3401415 ,
 5618.90635056, 12937.3401415 , 12937.3401415 , 12937.3401415 ,
 5618.90635056, 41675.0494835 ,  5618.90635056,  5618.90635056,
 5618.90635056, 12937.3401415 , 41675.0494835 ,  8718.82463052,
12937.3401415 ,  5618.90635056,  5618.90635056,  5618.90635056,
 5618.90635056,  8718.82463052,  5618.90635056,  5618.90635056,
12937.3401415 ,  8718.82463052,  5618.90635056,  5618.90635056,
12937.3401415 ,  5618.90635056, 41675.0494835 , 41675.0494835 ,
41675.0494835 ,  5618.90635056,  8718.82463052,  5618.90635056,
 5618.90635056, 21297.00147926,  5618.90635056, 21297.00147926,
41675.0494835 ,  5618.90635056, 12937.3401415 ,  5618.90635056,
12865.4786582 , 12865.4786582 , 21297.00147926, 12937.3401415 ,
 5618.90635056,  5618.90635056,  5618.90635056, 12937.3401415 ,
21297.00147926, 12937.3401415 , 12937.3401415 ,  5618.90635056,
 5618.90635056,  5618.90635056, 12865.4786582 ,  8718.82463052,
21297.00147926, 41675.0494835 , 12937.3401415 ,  8718.82463052,
12937.3401415 , 12865.4786582 ,  5618.90635056,  5618.90635056,
 5618.90635056,  5618.90635056, 12865.4786582 ,  5618.90635056,
12937.3401415 , 12937.3401415 ,  5618.90635056,  5618.90635056,
12937.3401415 ,  5618.90635056, 12937.3401415 ,  8718.82463052,
12937.3401415 , 41675.0494835 , 41675.0494835 , 41675.0494835 ,
 5618.90635056,  5618.90635056, 12937.3401415 ,  5618.90635056,
41675.0494835 ,  5618.90635056,  5618.90635056, 12937.3401415 ,
21297.00147926,  5618.90635056,  5618.90635056, 12937.3401415 ,
 5618.90635056,  5618.90635056,  5618.90635056,  5618.90635056,
21297.00147926,  8718.82463052,  5618.90635056,  5618.90635056,
 5618.90635056, 12937.3401415 , 21297.00147926,  8718.82463052,
 8718.82463052,  5618.90635056,  5618.90635056,  5618.90635056,
 5618.90635056,  7272.90208529, 12937.3401415 , 12937.3401415 ,
 5618.90635056,  5618.90635056])
```

In [27]:
```
from sklearn.metrics import r2_score
r2_score(y_test,ypred)
```

Out[27]:
```
0.8337173331647353
```

In [ ]:
```
```