In [1]:
```
#importing libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

from sklearn.linear_model import LinearRegression
from sklearn.ensemble import RandomForestRegressor
from sklearn.svm import SVR
from sklearn.tree import DecisionTreeRegressor
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
```

In [2]:
```
#loading Data
data=pd.read_csv("Desktop\\medical_cost_data.csv")
data
```

Out[2]:

	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 [3]:
```
data.describe()
```

Out[3]:

	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 [4]:
```
data.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 [5]:
```
#categorical_data
data['sex'].unique()
```

Out[5]:
```
array(['female', 'male'], dtype=object)
```

In [6]:
```
data['smoker'].unique()
```

Out[6]:
```
array(['yes', 'no'], dtype=object)
```

In [7]:
```
data['region'].unique()
```

Out[7]:
```
array(['southwest', 'southeast', 'northwest', 'northeast'], dtype=object)
```

In [9]:
```
#label_encoder
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
data['sex']=le.fit_transform(data['sex'])
data['smoker']=le.fit_transform(data['smoker'])
le.fit_transform(data['region'])
data.head()
```

Out[9]:

	age	sex	bmi	children	smoker	region	charges
0	19	0	27.900	0	1	southwest	16884.92400
1	18	1	33.770	1	0	southeast	1725.55230
2	28	1	33.000	3	0	southeast	4449.46200
3	33	1	22.705	0	0	northwest	21984.47061
4	32	1	28.880	0	0	northwest	3866.85520

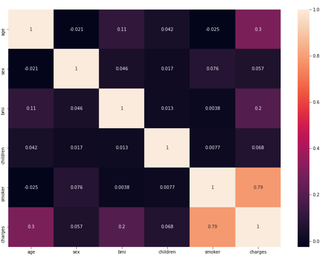
In [10]:
```
#data Correlation
data.corr()
```

Out[10]:

	age	sex	bmi	children	smoker	charges
age	1.000000	-0.020856	0.109272	0.042469	-0.025019	0.299008
sex	-0.020856	1.000000	0.046371	0.017163	0.076185	0.057292
bmi	0.109272	0.046371	1.000000	0.012759	0.003750	0.198341
children	0.042469	0.017163	0.012759	1.000000	0.007673	0.067998
smoker	-0.025019	0.076185	0.003750	0.007673	1.000000	0.787251
charges	0.299008	0.057292	0.198341	0.067998	0.787251	1.000000

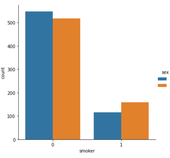
In [14]:
```
plt.figure(figsize=(14,10))
sns.heatmap(data.corr(),annot=True)
#from this heatmap we are going to knoe that that Smoker and charges have a very high relati
```

Out[14]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2b5579f3f10>
```



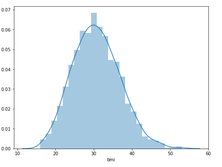
In [20]:
```
#Smoker Analysis
sns.factorplot(data=data,x='smoker',hue='sex',kind='count')
```

Out[20]:
```
<seaborn.axisgrid.FacetGrid at 0x2b55797ab50>
```



In [23]:
```
#BMI Analysis
plt.figure(figsize=(8,6))
sns.distplot(data['bmi'])
#so the average bmi is 30
```

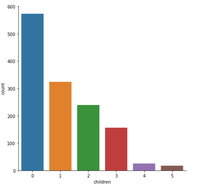
Out[23]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2b557cdff70>
```



In [25]:
```
#children count analysis
sns.factorplot(data=data,x='children',kind='count',size=6)
```

```
C:\Users\Muthi karunakar\anaconda3\lib\site-packages\seaborn\categorical.py:3672: UserWarnin
g:The `size` parameter has been renamed to `height`; please update your code.
  warnings.warn(msg, UserWarning)
```

Out[25]:
```
<seaborn.axisgrid.FacetGrid at 0x2b557ca16a0>
```



In [30]:
```
#splitting data
x=data.drop(data.columns[[5,6]],axis=1)
y=data['charges']
x
```

Out[30]:

	age	sex	bmi	children	smoker
0	19	0	27.900	0	1
1	18	1	33.770	1	0
2	28	1	33.000	3	0
3	33	1	22.705	0	0
4	32	1	28.880	0	0
...	...	...	...	...	...
1333	50	1	30.970	3	0
1334	18	0	31.920	0	0
1335	18	0	36.850	0	0
1336	21	0	25.800	0	0
1337	61	0	29.070	0	1

1338 rows × 5 columns

In [31]:
```
#splitting data for training and testing output
xtrain,xtest,ytrain,ytest=train_test_split(x,y,test_size=0.2,random_state=0)
```

In [34]:
```
#feature Scaling
from sklearn.preprocessing import StandardScaler
sc_x=StandardScaler()
xtrain=sc_x.fit_transform(xtrain)
xtest=sc_x.fit_transform(xtest)
```

In [38]:
```
#ML Models
linear=LinearRegression()
linear.fit(xtrain,ytrain)
dt=DecisionTreeRegressor()
dt.fit(xtrain,ytrain)
svr=SVR()
svr.fit(xtrain,ytrain)
rf=RandomForestRegressor(n_estimators=1000,random_state=0)
rf.fit(xtrain,ytrain)
```

Out[38]:
```
RandomForestRegressor(n_estimators=1000, random_state=0)
```

In [39]:
```
#prediction
linear_pred=linear.predict(xtest)
dt_pred=dt.predict(xtest)
svr_pred=svr.predict(xtest)
rf_pred=rf.predict(xtest)
```

In [43]:
```
#RMSE Error
import math
print("Mean Squared Error of Linear Regression ",math.sqrt(mean_squared_error(ytest,linear_pr
ed)),"\n")
print("Mean Squared Error of Decision Tree Regressor ",math.sqrt(mean_squared_error(ytest,dt
_pred)),"\n")
```

```
Mean Squared Error of Linear Regression  5674.794942435093

Mean Squared Error of Decision Tree Regressor  6586.085970276423

Mean Squared Error of SVR  13223.669474876637

Mean Squared Error of RandomForest Regressor  5674.794942435093
```

In [ ]:

In [ ]:

In [ ]:

In [ ]: