Insurance

July 28, 2019

Assignment - 1

0.1 1. Necessary Libraries are imported

[1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from scipy import stats
%matplotlib inline
```

0.2 2. Reading the data as a data frame

[2]:
```
df = pd.read_csv("insurance.csv")
```

0.3 3. Basic Exploratory Data Analysis

0.3.1 a. Shape of the data

[3]:
```
df.shape
```

[3]:
```
(1338, 7)
```

0.3.2 b. Data type of each attribute

[4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1338 entries, 0 to 1337
Data columns (total 7 columns):
age         1338 non-null int64
sex         1338 non-null object
bmi         1338 non-null float64
children    1338 non-null int64
smoker      1338 non-null object
region      1338 non-null object
```

```
charges     1338 non-null float64
dtypes: float64(2), int64(2), object(3)
memory usage: 73.2+ KB
```

0.3.3 c. Checking the presence of missing values

[5]:
```python
print(df.isnull().values.any())
```

```
False
```

0.3.4 d. 5 Point Summary of numerical attributes

[6]:
```python
df.describe()
```

[6]:

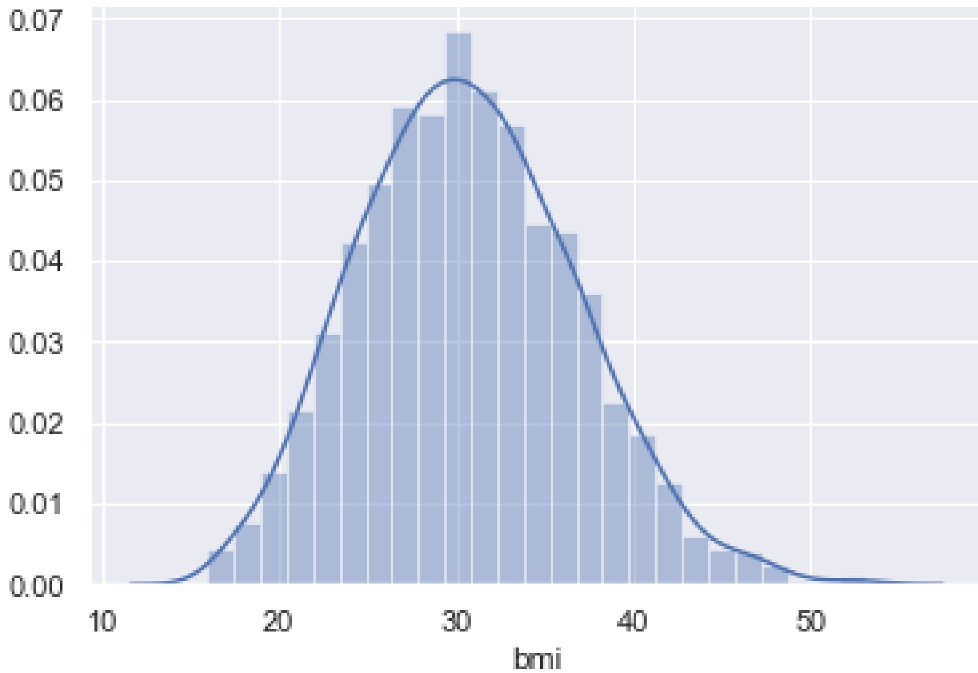
	age	bmi	children	charges
count	1338.000000	1338.000000	1338.000000	1338.000000
mean	39.207025	30.663397	1.094918	13270.422265
std	14.049960	6.098187	1.205493	12110.011237
min	18.000000	15.960000	0.000000	1121.873900
25%	27.000000	26.296250	0.000000	4740.287150
50%	39.000000	30.400000	1.000000	9382.033000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

0.3.5 e. Univariate Analysis

Distribution of column - 'bmi'

[7]:
```python
sns.set(color_codes=True)

#Uni-variate distribution using seaborn

sns.distplot(df['bmi'])
```

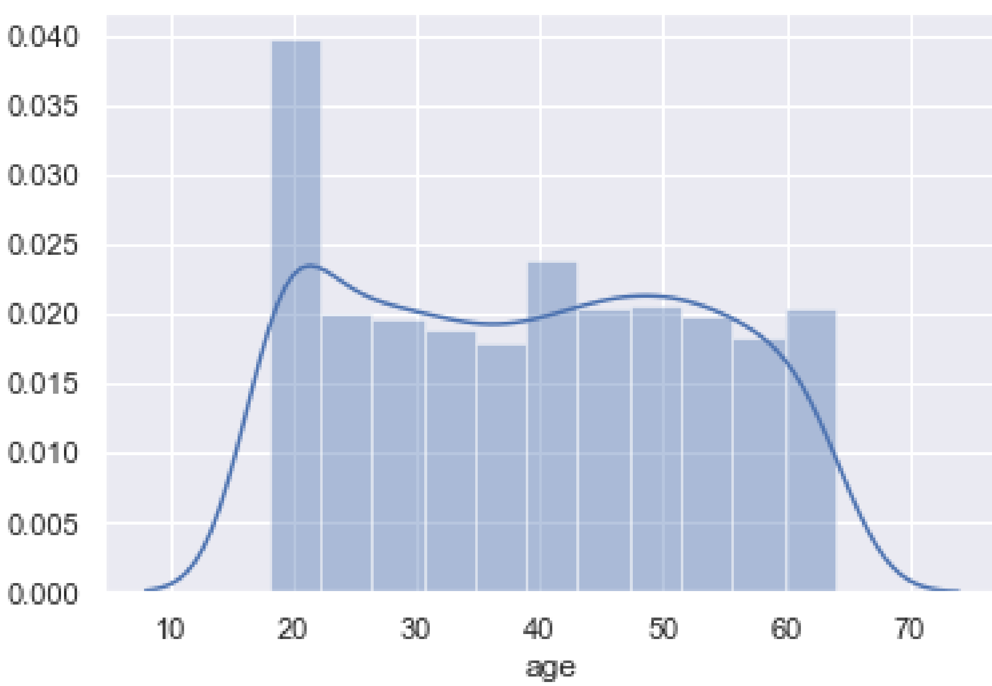
[7]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x11f378f60>
```



Distribution of column - 'age'

[8]:
```
sns.distplot(df['age'])
```

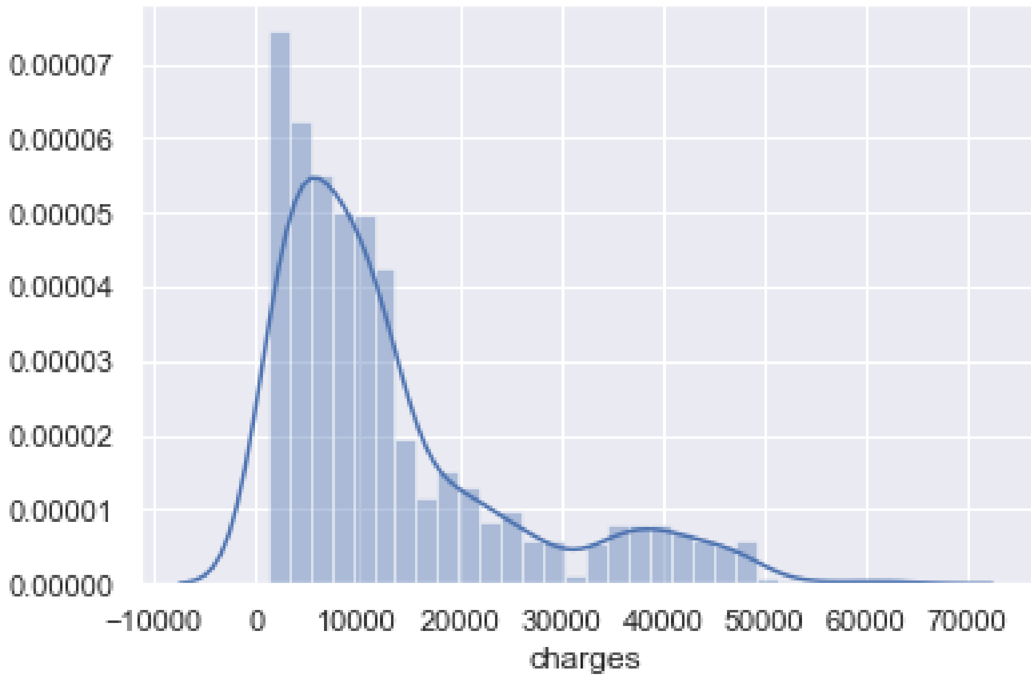
[8]: `<matplotlib.axes._subplots.AxesSubplot at 0x12371bda0>`



Distribution of column - 'charges'

[9]:
```
sns.distplot(df['charges'])
```

[9]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x123825390>
```



0.3.6 f. Measure of skewness

[10]:
```
#bmi

df['bmi'].skew()
```

[10]:
```
0.2840471105987448
```

Skew value is 0.2, we can say that this distribution will be almost symmetric

[11]:
```
#age

df['age'].skew()
```

[11]:
```
0.05567251565299186
```

Skew value is 0.05, we can say that this distribution will be almost symmetric

[12]:
```
#charges

df['charges'].skew()
```

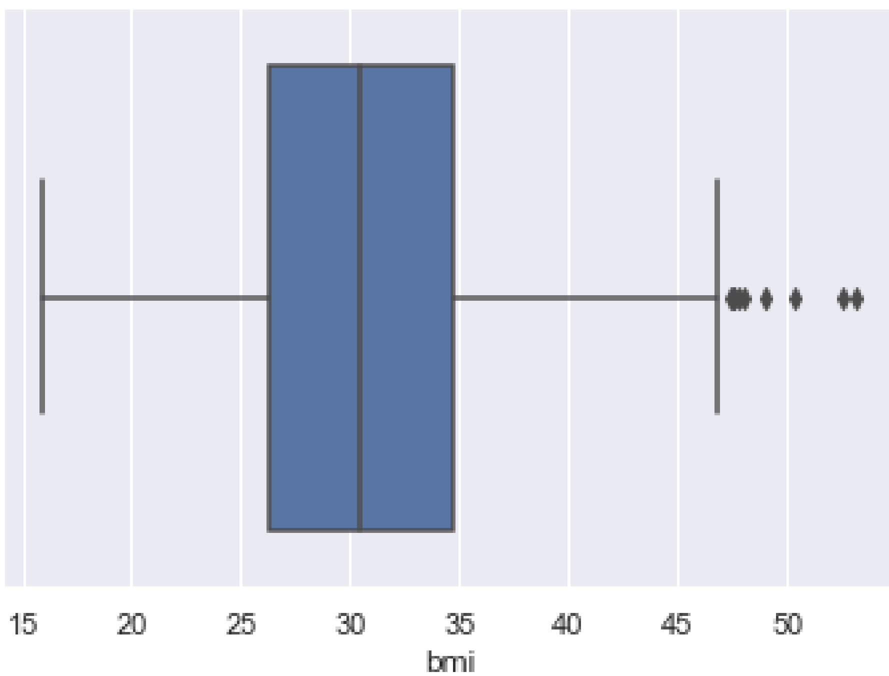
[12]: 1.5158796580240388

Skew value is 1.5, we can say that this distribution will be highly skewed (Positive / Right Skew)

0.3.7 g. Checking the presence of outliers

[13]:
```
#Checking the presence of outliers in column 'bmi'

sns.boxplot(x='bmi',data=df)
```

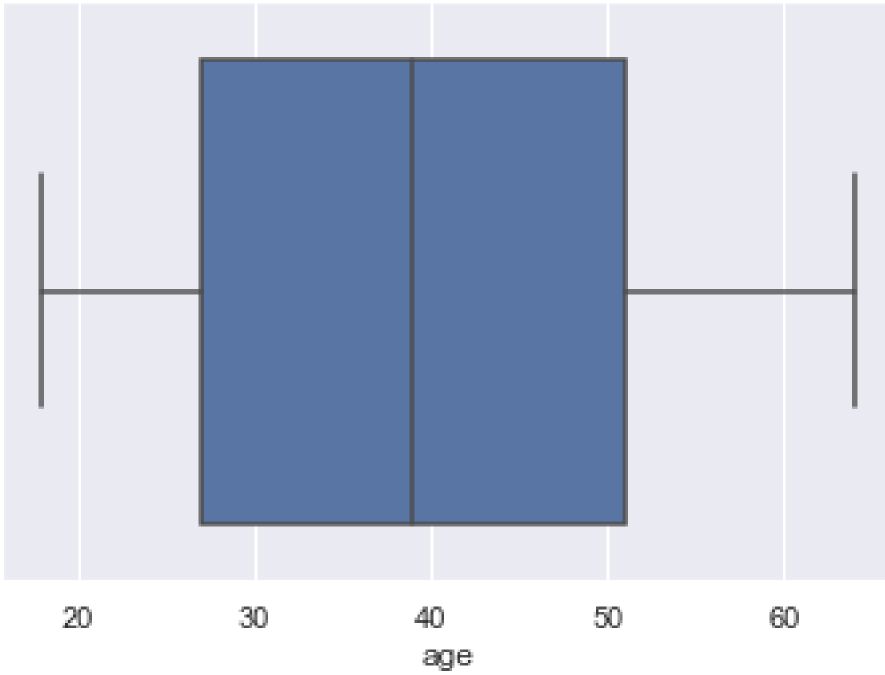
[13]: <matplotlib.axes._subplots.AxesSubplot at 0x1238e20b8>



There are some points placed between (approximate values) 47 to 55, which is not inside the box nor near the quartiles, that says the presence of outliers in the column 'bmi'

[14]:
```
#Checking the presence of outliers in column 'age'

sns.boxplot(x='age',data=df)
```

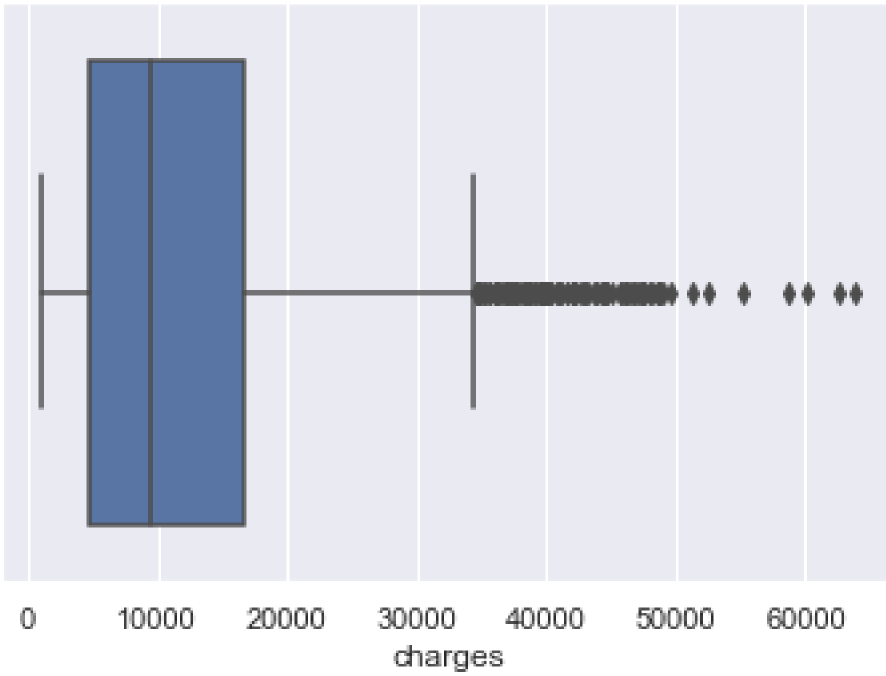
[14]: <matplotlib.axes._subplots.AxesSubplot at 0x1239b3128>



There are no visible points lying outside the box nor near the quartiles, that says outliers is not present in the column 'age'

[15]:
```
#Checking the presence of outliers in column 'charges'

sns.boxplot(x='charges',data=df)
```

[15]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x123a8a2b0>
```

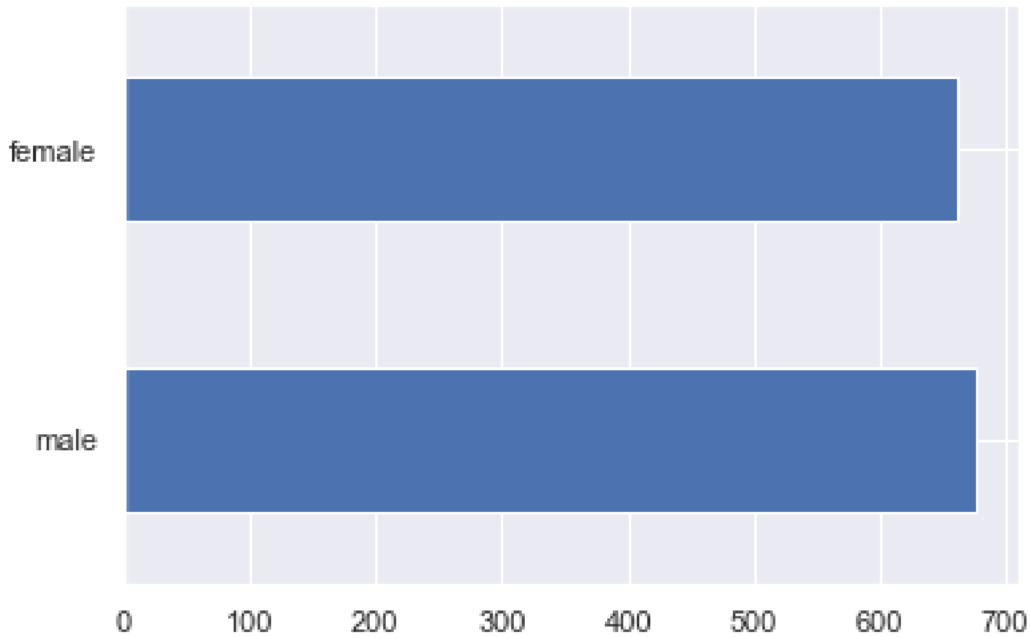


There are plenty of points placed between (approximate values) 35K to 65K, which is not inside the box nor near the quartiles, that says the presence of outliers in the column 'charges'

0.3.8 h. Categorical Column Distribution

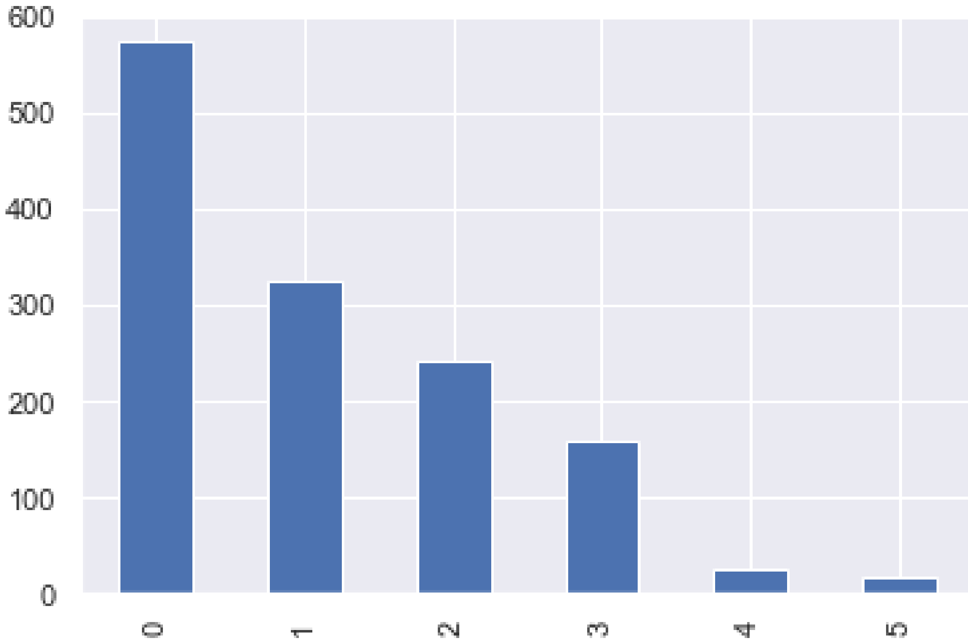
[16]:
```
#value-counts() helps us in analyzing the categorical variables

sex = df['sex'].value_counts()

#Kind of plot - Bar Horizontal

sex.plot(kind='barh')
```

[16]: <matplotlib.axes._subplots.AxesSubplot at 0x123b5ea90>



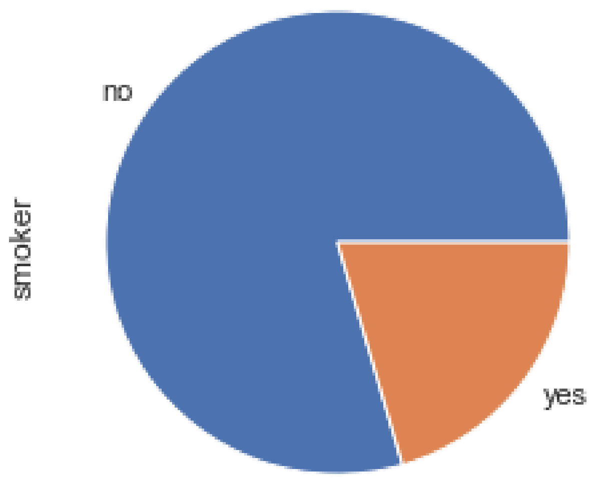
[17]:
```
children = df['children'].value_counts()

#Kind of plot - Bar

children.plot(kind='bar')
```

[17]: <matplotlib.axes._subplots.AxesSubplot at 0x123c19630>



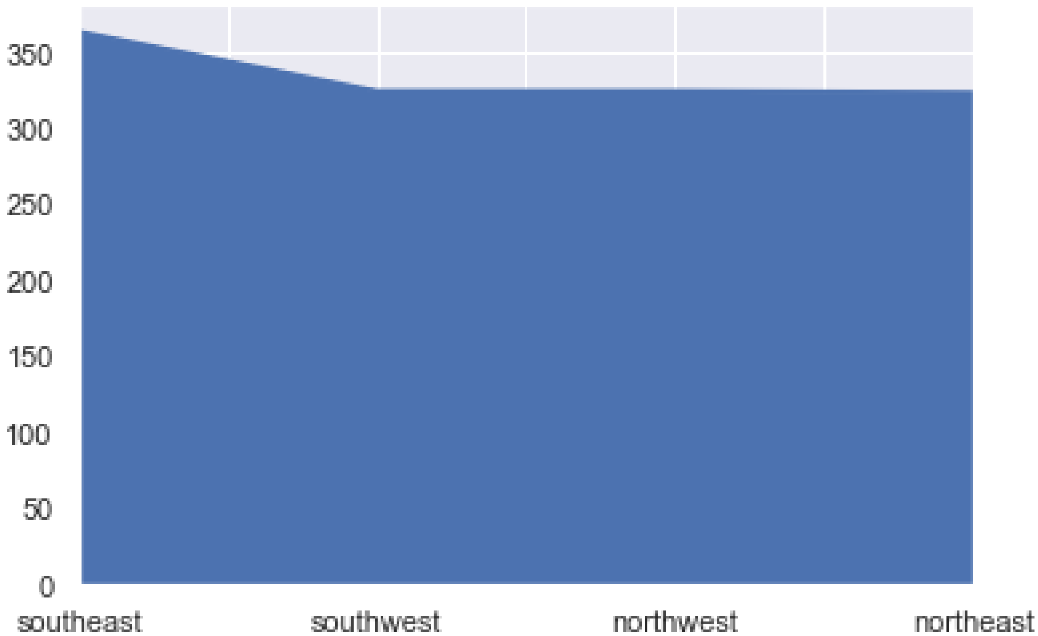
[18]:
```
smoker = df['smoker'].value_counts()

#Kind of plot = Pie

smoker.plot(kind='pie')
```

[18]: <matplotlib.axes._subplots.AxesSubplot at 0x123cccf28>



[19]:
```
region = df['region'].value_counts()

#Kind of plot = Area

region.plot(kind='area')
```

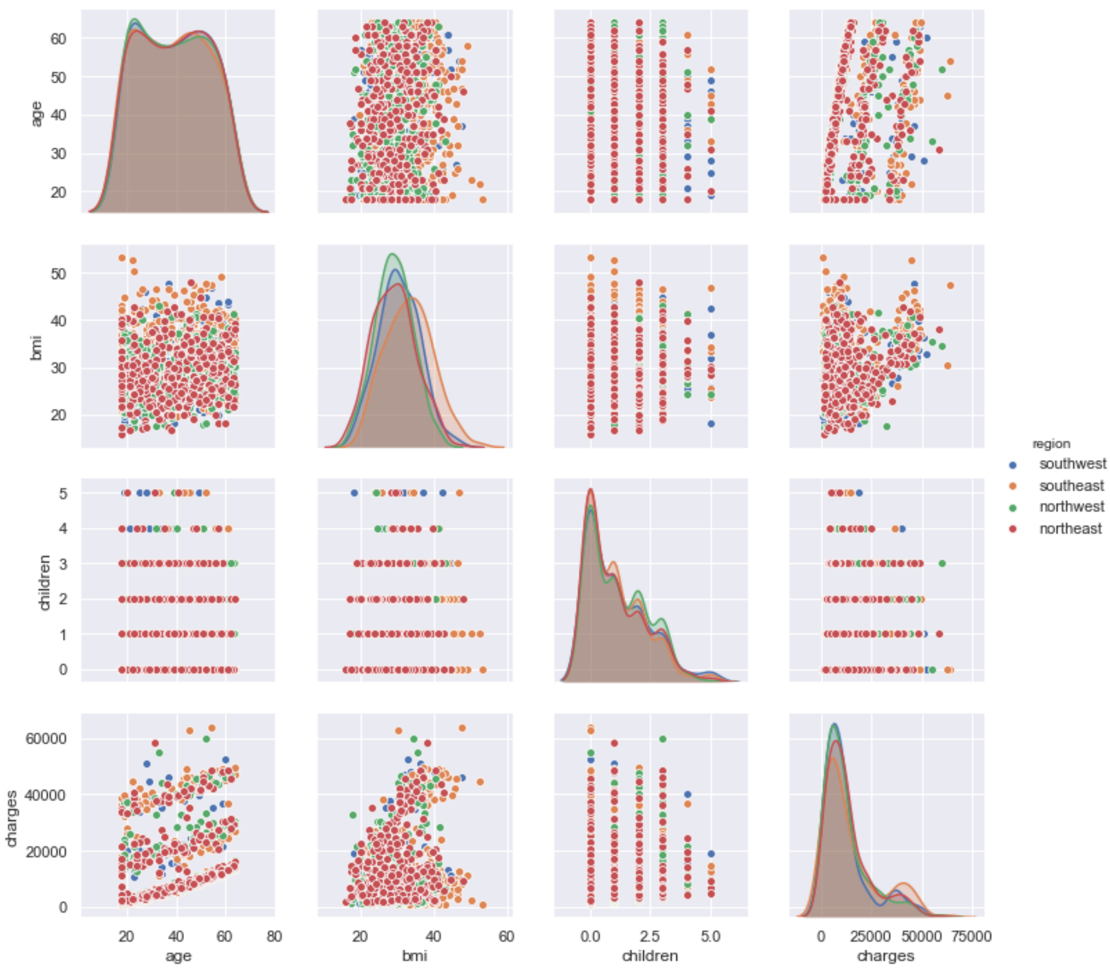
[19]: <matplotlib.axes._subplots.AxesSubplot at 0x123d3e6d8>



0.3.9 i. Pair plot

[20]:
```
#Pair-plot of the columns in the data frame, with column 'region' as hue value

sns.pairplot(df, hue = 'region')
```

[20]: <seaborn.axisgrid.PairGrid at 0x123e254a8>

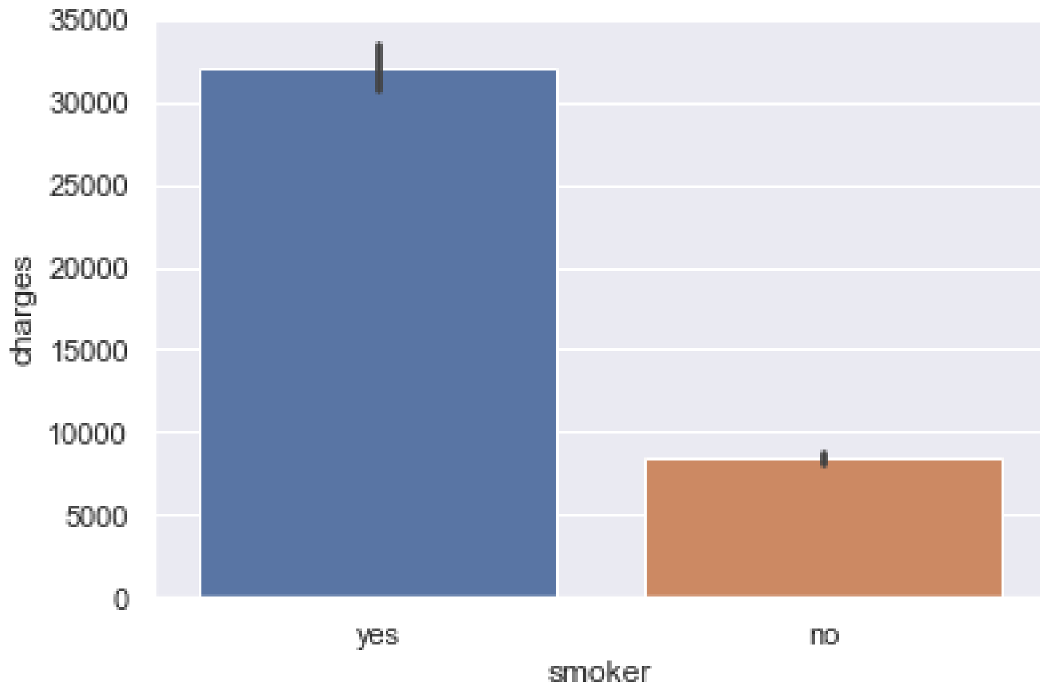


0.4 4. Answer the following with statistical evidence

0.4.1 a. Do charges of people who smoke differ significantly from the people who don't?

[21]:
```
#Barplot gives a spectacular bi-variate visualization over categorical␣
↪variables

sns.barplot(x = "smoker", y = "charges", data = df)
```

[21]: <matplotlib.axes._subplots.AxesSubplot at 0x124764898>

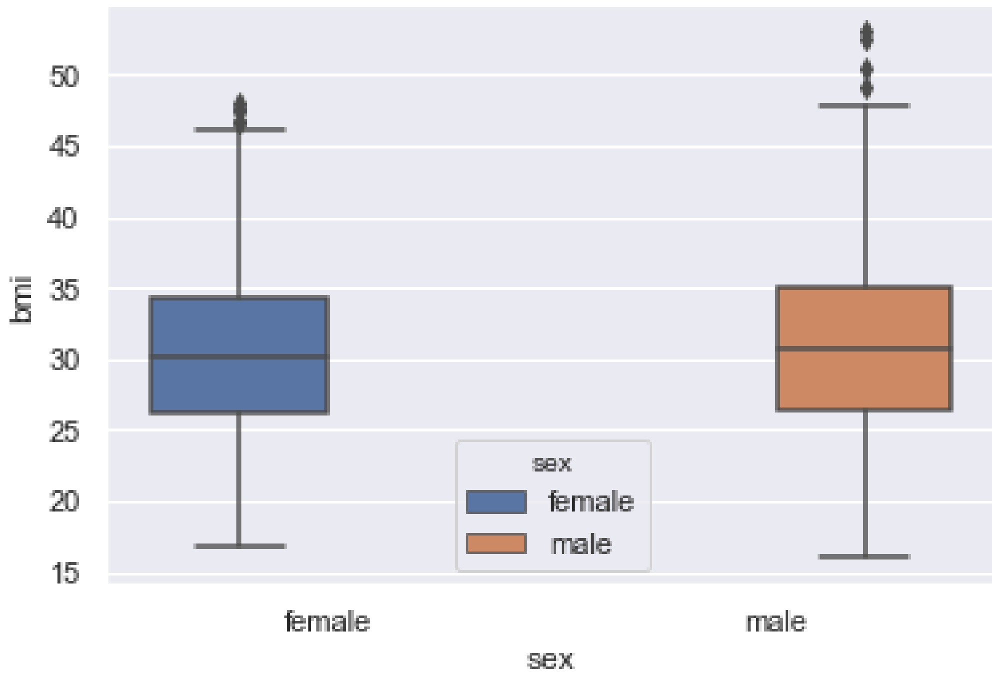


Observation: Yes, charges of people who smoke differ significantly from the people who don't, where smokers are charged way higher than the non-smokers

0.4.2 b. Does bmi of males differ significantly from that of females?

[22]:
```
#Tried the bi-variate abalysis with boxplot, which is more informative

sns.boxplot(x = "sex", y ="bmi", data = df, hue = 'sex')
```

[22]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x11f564668>
```



Observation: No, bmi of males does not differ significantly from that of females

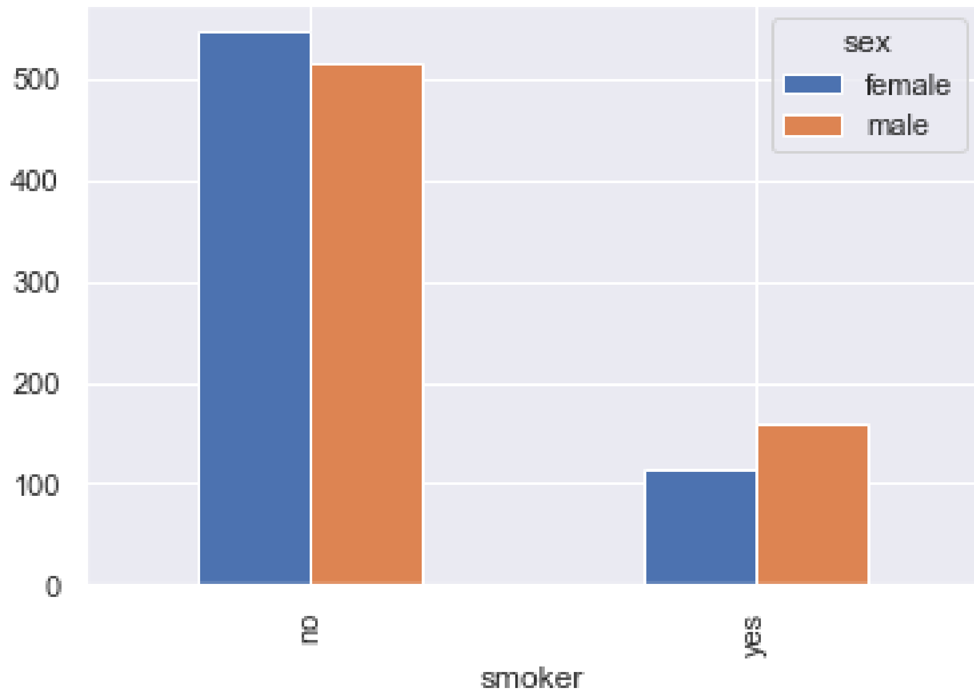
0.4.3 c. Is the proportion of smokers significantly different in different genders?

[23]:
```
#Referred pd.crosstab from https://adataanalyst.com/data-analysis-resources/
↪visualise-categorical-variables-in-python/

smoker_gender = pd.crosstab(index = df["smoker"], columns = df["sex"])
smoker_gender
```

[23]:
```
sex     female  male
smoker
no         547   517
yes        115   159
```

[24]:
```
#Nice visualization to get the proportion of smokers from genders

smoker_gender.plot(kind="bar")
```

[24]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x124cc6400>
```



Observation: No, the proportion of smokers doe not significantly differ in different genders, still there is a slight deviation but not significantly

0.4.4 d. Is the distribution of bmi across women with no children, one child and two children, the same ?

[25]:
```
#In order for smooth operations converting str to int, where assigning 'female'␣
↪to 0 and 'male' to 1

def convert(x):
    if x == "female":
        return 0
    else:
        return 1

df1 = df
gender = df1['sex']
women = gender.map(convert)
df1['sex'] = women

df1.head()
```

[25]:

	age	sex	bmi	children	smoker	region	charges
0	19	0	27.900	0	yes	southwest	16884.92400
1	18	1	33.770	1	no	southeast	1725.55230

```
2   28    1   33.000          3      no   southeast    4449.46200
3   33    1   22.705          0      no   northwest   21984.47061
4   32    1   28.880          0      no   northwest    3866.85520
```

[26]:
```
#Extracting the data where 'sex' = 0 , i.e, Female

women_a = df1[df1['sex'] == 0]
women_a.head()
```

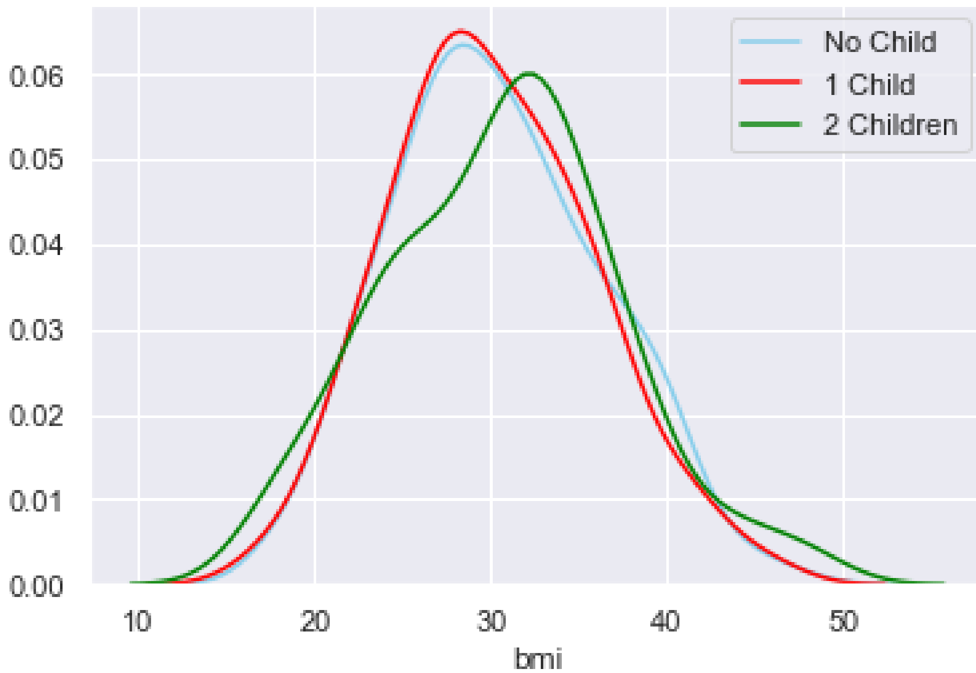
[26]:

	age	sex	bmi	children	smoker	region	charges
0	19	0	27.90	0	yes	southwest	16884.92400
5	31	0	25.74	0	no	southeast	3756.62160
6	46	0	33.44	1	no	southeast	8240.58960
7	37	0	27.74	3	no	northwest	7281.50560
9	60	0	25.84	0	no	northwest	28923.13692

[27]:
```
#Extracting the data where 'children' = 0, 1 & 2 respectively

women_no_child = women_a[women_a['children'] == 0]
women_one_child = women_a[women_a['children'] == 1]
women_two_child = women_a[women_a['children'] == 2]
```

[28]:
```
#Generating the distribution between bmi across three different data of women
 ↪with children count

sns.distplot( women_no_child["bmi"] , color="skyblue", label="No Child", hist =
 ↪False)
sns.distplot( women_one_child["bmi"] , color="red", label="1 Child", hist =
 ↪False)
sns.distplot( women_two_child["bmi"] , color="green", label="2 Children", hist
 ↪= False)
```

[28]: <matplotlib.axes._subplots.AxesSubplot at 0x124e40080>



Observation: We can see that distribution of data with 'No Child' and '1 Child' is almost same but we can see some changes in the data with '2 Children', where the mean is moving away from those of 'No Child' and '1 Child'