EDA on Insurance data

By Sandesh Balyan

This notebook will analyse data in Insurance.csv file and answer some critical questions about this data

1. Importing Required Libraries

In [51]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import scipy.stats as stats
from scipy.stats import ttest_ind,levene
from statsmodels.stats.proportion import proportions_ztest
import statsmodels.api         as     sm
from   statsmodels.formula.api import ols
```

2. Importing csv file in a dataframe

In [3]:
```
data = pd.read_csv('Insurance.csv')
```

In [4]:
```
data.head()
```

Out[4]:

	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

3. Basic EDA

3.a Shape of the data

In [5]:
```
data.shape
```

Out[5]: (1338, 7)

There are total 1338 rows in the csv and 7 columns

3.b Datatype of each attribute

In [6]:
```
data.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
```

In [62]:
```
# We can also verify datatype of each attribute using the function 'dtype' for each a
ttribute of the dataframe individually.

data.dtypes
```

Out[62]:
```
age           int64
sex          object
bmi         float64
children      int64
smoker       object
region       object
charges     float64
dtype: object
```

Conclusion

1. There are total 7 attributes int he dataset
2. 4 attributes are quantitative/numerical
3. 3 attributes are Qualitative/categorical in nature

3.c Checking the presence of missing values

If some attributes have null values for any record, the same shall be handled befor analysing the data. Although from section 4 usinf df.info() we can see that none of the attribute has any null value in this dataset, however we can still make individual assessment of the columns as follows

In [8]:
```
data.isnull().sum()

#data.isna() can also be used. Both returns the same results.
```

Out[8]:
```
age         0
sex         0
bmi         0
children    0
smoker      0
region      0
charges     0
dtype: int64
```

Checking duplicate values

```
In [9]: duplicates = data.duplicated()
        duplicates.sum()
```

```
Out[9]: 1
```

We can see that there is one duplicate values in the dataset. we can drop this duplicate value

```
In [10]: data = data.drop_duplicates()
```

```
In [11]: #checking the shape of the dataset again
         data.shape
```

```
Out[11]: (1337, 7)
```

Conclusion:

1. We can conclude that there are no missing values/null values in any of the attribute in the given dataset
2. There was only 1 duplicate value and same has been dropped from the dataset
3. After dropping the values new number of rows/records has become 1337

3.d Five points summary of numerical attributes

```
In [13]: data.describe()
```

Out[13]:

	age	bmi	children	charges
count	1337.000000	1337.000000	1337.000000	1337.000000
mean	39.222139	30.663452	1.095737	13279.121487
std	14.044333	6.100468	1.205571	12110.359656
min	18.000000	15.960000	0.000000	1121.873900
25%	27.000000	26.290000	0.000000	4746.344000
50%	39.000000	30.400000	1.000000	9386.161300
75%	51.000000	34.700000	2.000000	16657.717450
max	64.000000	53.130000	5.000000	63770.428010

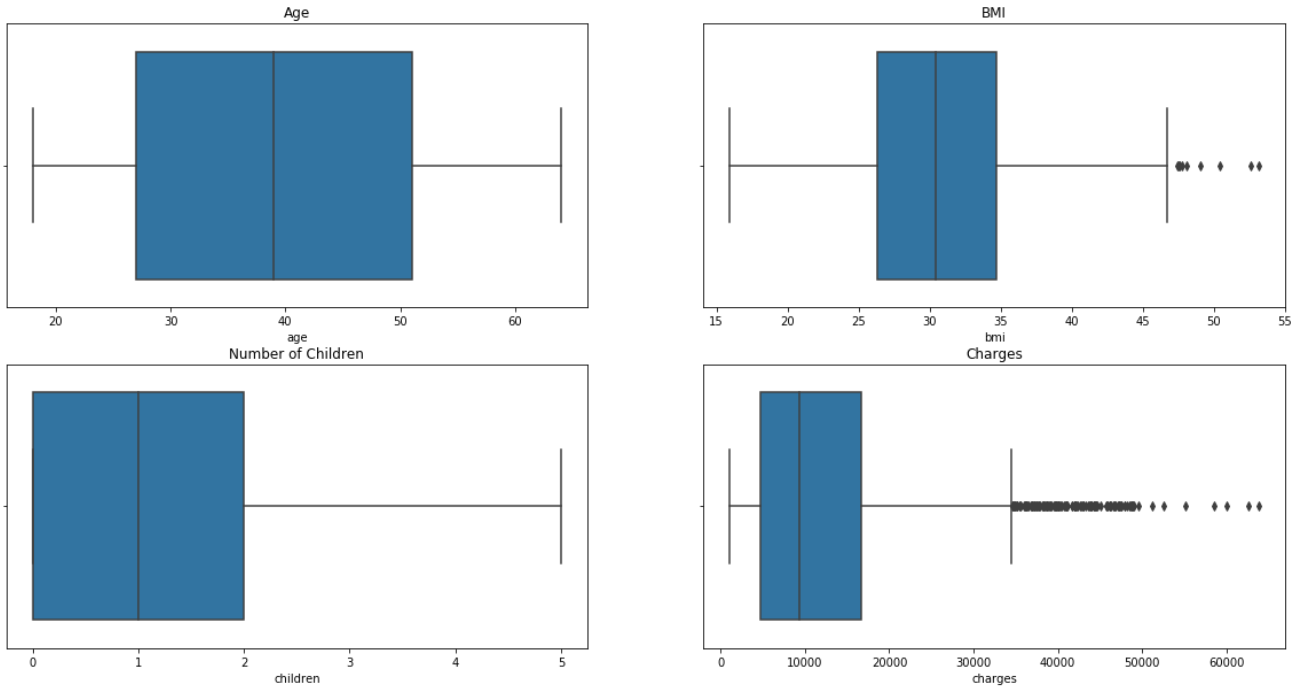
From the above figure we can see the 5 numbers summary for numerical variables along with mean and standard deviation

1. Mean and median of attributes 'Age', 'BMI' and 'Children' are almost same hence no skewness in distribution
2. Mean and max value of attribute 'charges' are significantly more than median(9382) hence there is positive skewness in the distribution for Charges

We can also interpret 5 number summary for dataset using box plots

In [14]:
```
fig, ((ax1, ax2), (ax3, ax4)) = plt.subplots(nrows=2,ncols=2,figsize=(20,10))

sns.boxplot(data['age'],ax=ax1,)
sns.boxplot(data['bmi'],ax=ax2)
sns.boxplot(data['children'],ax=ax3)
sns.boxplot(data['charges'],ax=ax4)

#ax1.text(data['age'].min(),2,data['age'].min())


ax1.set_title('Age')
ax2.set_title('BMI')
ax3.set_title('Number of Children')
ax4.set_title('Charges')
```

Out[14]: Text(0.5,1,'Charges')



Observations

1. Age is evenly distributed in sample and there are no outliers inn the age attributes. Age distribution doesnot seemed to be skewed in any direction
2. bmi has few outliers on the higher end and data is slightly positively skewed
3. min children are 0 and 75% of people have 2 or less than 2 children.
4. Charges: Data is heavely positively skewed and there are huge number of outliers.

3.e Distribution of columns 'age', 'bmi' and 'charges'

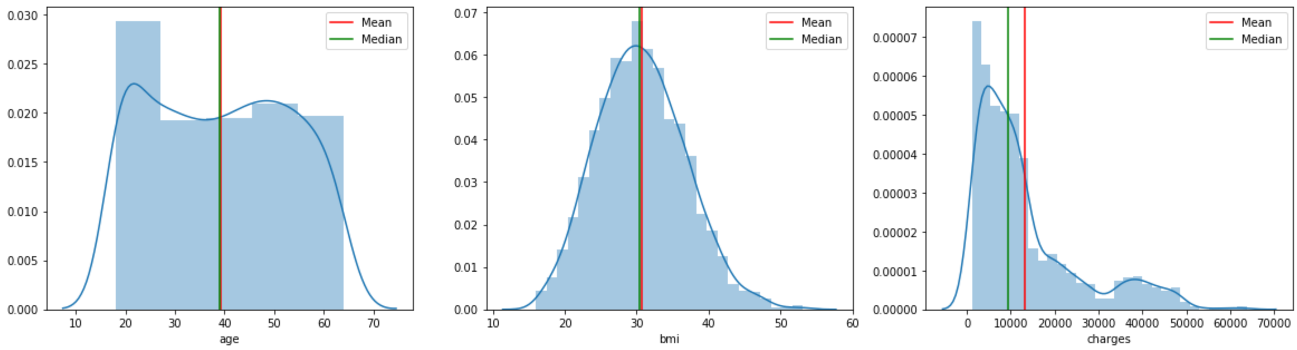
To check the distribution of numerical columns we can:

1. plot distribution plots using seaborn
2. plot histograms using matplotlib.

```
In [15]: fig, (ax1,ax2,ax3) = plt.subplots(1,3,figsize=(20,5))
sns.distplot(data['age'],bins=5,ax=ax1)
sns.distplot(data['bmi'],ax=ax2)
sns.distplot(data['charges'],ax=ax3)

ax1.axvline(data['age'].mean(),c='r',label='Mean')
ax1.axvline(data['age'].median(),c='g',label='Median')

ax2.axvline(data['bmi'].mean(),c='r',label='Mean')
ax2.axvline(data['bmi'].median(),c='g',label='Median')

ax3.axvline(data['charges'].mean(),c='r',label='Mean')
ax3.axvline(data['charges'].median(),c='g',label='Median')

ax1.legend(loc="upper right")
ax2.legend(loc="upper right")
ax3.legend(loc="upper right")
```

```
C:\Users\sande\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:6462: UserWarnin
g: The 'normed' kwarg is deprecated, and has been replaced by the 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
C:\Users\sande\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:6462: UserWarnin
g: The 'normed' kwarg is deprecated, and has been replaced by the 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
C:\Users\sande\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:6462: UserWarnin
g: The 'normed' kwarg is deprecated, and has been replaced by the 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
```

```
Out[15]: <matplotlib.legend.Legend at 0x175d61f0da0>
```



Observations for distributions

Age:

1. Data is not completely but approximately symmetrically distributed with centre of distribution being mean
2. Mean and Median are approximately same and are at the centre of the distribution Mean = Median = 40
3. Data is equally distributed on both the sides of the curve, implying no skewness in distribution
4. Distribution of age is almost equal among all age group except 20-25 where there are maximum number of observations and there is very less variability in number of people in different age group #### bmi
5. BMI is fairly symmetrical and follows approximately normal distribution with Mean = 30 = Median
6. There seems to be a little skewness in the right tail but is ignorable
7. There are very less number of people with bmi>50 and majority of people have abmi between 25 and 35 #### Charges
8. Data is not symmetrical at all
9. Data is right skewed and large values are less in number Hence positively skewed
10. Mean(15000)> Median(10000)
11. This also indicates presence of potential outliers at the higher side i.e. There are some charges which are very high

3.f Measure of skewness of 'age', 'bmi' and 'charges' column

Skewness is measure of assymetry in the distribution This can be infererd from the distribution plots mentioned in section 3.e This can also be interpreted using values of function .skew()

```
In [16]: data[['age','bmi','charges']].skew()
```

```
Out[16]: age        0.054781
         bmi        0.283914
         charges    1.515391
         dtype: float64
```

Observations

1. Age: skewness value is positive but very small close to zero indicating no skewness

2. bmi: skewness value is positive and also not significant. This indicates that there is skewness but very small.

3. charges: skewness values is positive and very large. This indicates that the distribution is right tailed or positively skewed.

Skewness values are in sync with histograms in the above section.

3.g Checking the presence of outliers in 'age', 'bmi' and 'charges'

Outliers in an attribute can bde detected using

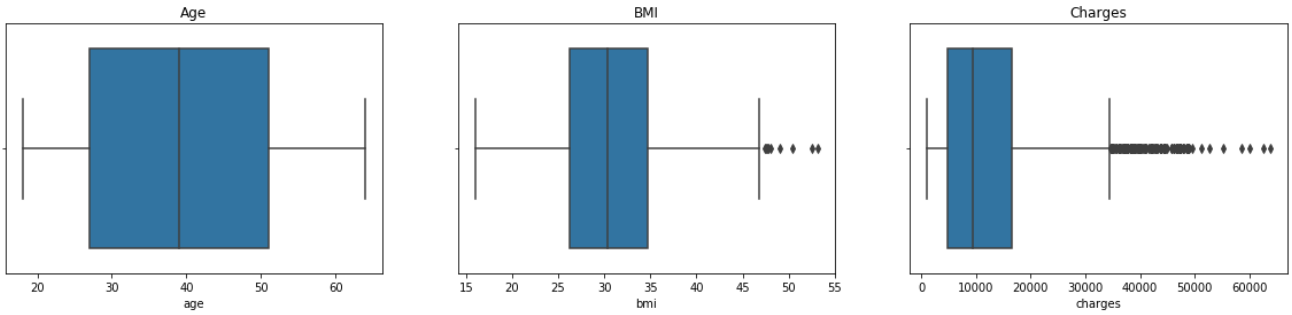
1. box plots
2. z scores

Outliers using box plots

We will plot boxplots for age, bmi and charges attributes

In [17]:
```
fig, (ax1, ax2, ax3) = plt.subplots(nrows=1,ncols=3,figsize=(20,4))

sns.boxplot(data['age'],ax=ax1)
sns.boxplot(data['bmi'],ax=ax2)
sns.boxplot(data['charges'],ax=ax3)

#ax1.text(data['age'].min(),2,data['age'].min())

ax1.set_title('Age')
ax2.set_title('BMI')
ax3.set_title('Charges')
```

Out[17]:
```
Text(0.5,1,'Charges')
```



Observations:

1. Age : No presence of outliers
2. BMI : Outliers on the higher end of the data.
3. Charges: Huge number of outliers on the higher end of the data

Boxplots can be used for detection of outliers but how many outliers and which values are the outliers cannot be detected. This can be achieved using z scores method as mentioned below

Z Score for outliers

In [18]:
```
z = stats.zscore(data[['age','bmi','charges']])
z
```

Out[18]:
```
array([[-1.44041773, -0.45315959,  0.2978567 ],
       [-1.51164747,  0.50942165, -0.9543806 ],
       [-0.79935006,  0.3831546 , -0.72937251],
       ...,
       [-1.51164747,  1.01448983, -0.96228744],
       [-1.29795825, -0.79752426, -0.9310536 ],
       [ 1.55123139, -0.26129928,  1.31029752]])
```

z scores returns a normalised values of each column. z Score is the measure of how many standard deviation away is particular observation So we can define a threshold and if a particular observation is farther away than the threshold it will be an outlier

In [19]:
```
th = 3
np.where(z>3)
#this returns an array of rows and columns with indices of the values violating this
 rule. Hence ouliers
```

Out[19]:
```
(array([  34,  116,  543,  577,  818,  846, 1046, 1145, 1229, 1299, 1316],
       dtype=int64), array([2, 1, 2, 2, 2, 1, 1, 2, 2, 2, 1], dtype=int64))
```

```
In [20]: z[34][2]
```

```
Out[20]: 3.1319982217432654
```

1. This indicates that value in 34th row and 2nd column is an outlier
2. There are 12 outliers combined in column 1 and 2 , i.e bmi and charges
3. There are no outliers in column 'age'

3.h Distribution of categorical variables (including children)

```
In [21]: data.info()
```

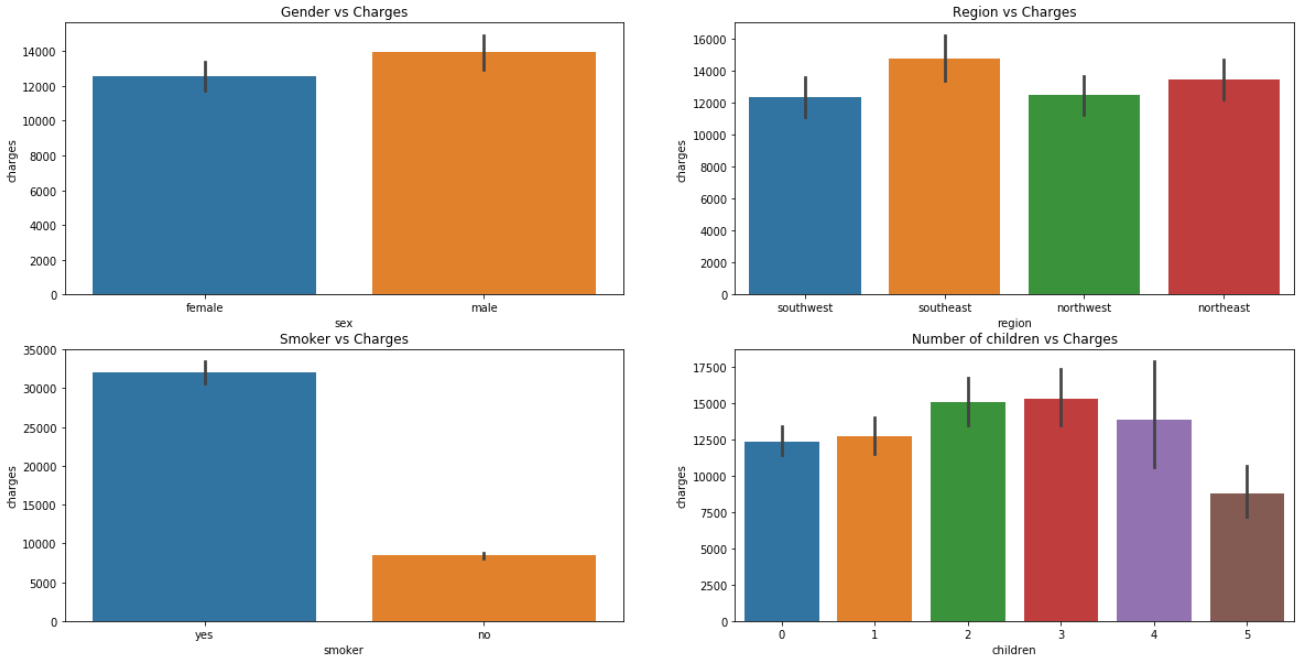
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1337 entries, 0 to 1337
Data columns (total 7 columns):
age         1337 non-null int64
sex         1337 non-null object
bmi         1337 non-null float64
children    1337 non-null int64
smoker      1337 non-null object
region      1337 non-null object
charges     1337 non-null float64
dtypes: float64(2), int64(2), object(3)
memory usage: 83.6+ KB
```

From above information

1. There are 3 non numeric or categorical variables in the dataset
2. these are sex, smoker and region
3. We will treat children or number of children as the categorical variable as well

In [22]:
```
fig, ((ax1, ax2), (ax3,ax4)) = plt.subplots(nrows=2,ncols=2,figsize=(20,10))

sns.barplot(x=data['sex'],y=data['charges'],ax=ax1)
sns.barplot(x=data['region'],y=data['charges'],ax=ax2)
sns.barplot(x=data['smoker'],y=data['charges'],ax=ax3)
sns.barplot(x=data['children'],y=data['charges'],ax=ax4)
#ax1.text(data['age'].min(),2,data['age'].min())

ax1.set_title('Gender vs Charges')
ax2.set_title('Region vs Charges')
ax3.set_title('Smoker vs Charges')
ax4.set_title('Number of children vs Charges')
```

Out[22]:
```
Text(0.5,1,'Number of children vs Charges')
```



Observations

Gender vs charges:

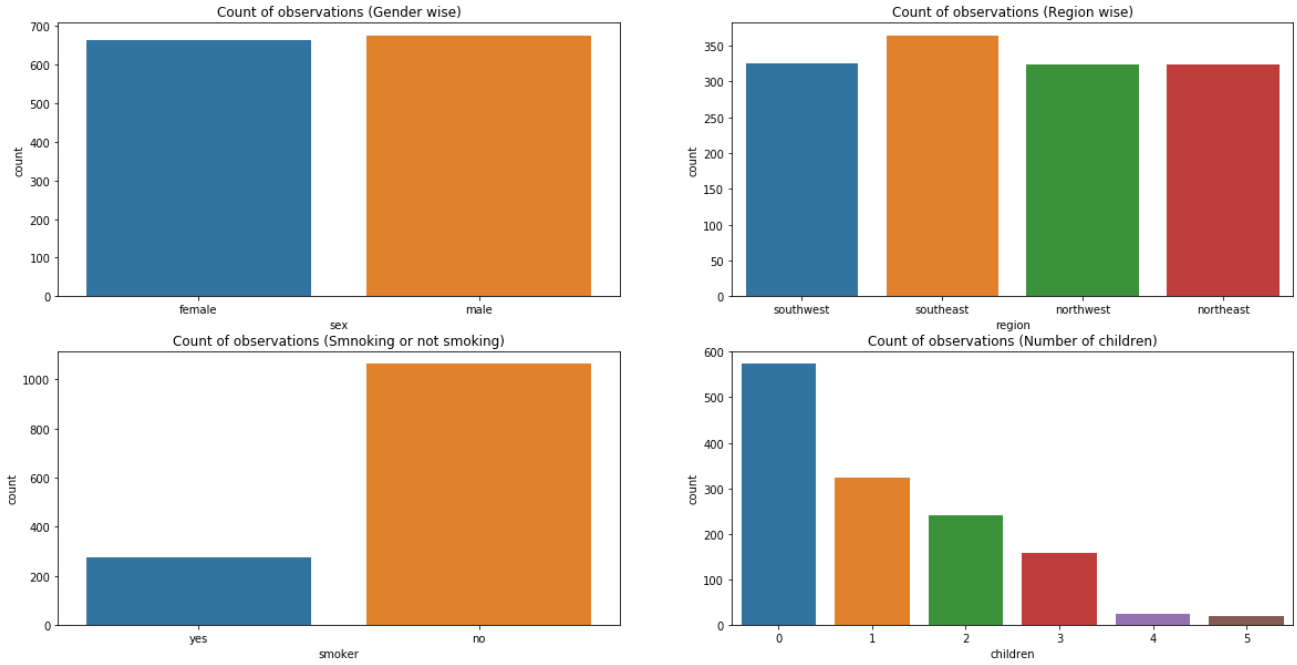
1. Charges for males and females are almost the same with a marginal difference #### Region vs charges:
2. Region 'southeast' pays a little bit higher charges compared to other regions #### Smoker vs charges:
3. There are huge difference in the insurance charges between snokers and non smokers.
4. Smokers pay more than non smokers #### Number of children vs charges:
5. There are slight higher insurance charges if number of children is 2 or 3
6. There is no significant difference in the charges if number if children are 0,1 or 5
7. Insurance charges considerably reduce when number of children are 5

Countplots can be used to plot number of observations per category

Count plot provides us with number of observation in each category in each categorical variable This will help us know if there is any imbalance in recorded data

In [23]:
```
fig, ((ax1, ax2), (ax3,ax4)) = plt.subplots(nrows=2,ncols=2,figsize=(20,10))

sns.countplot(x=data['sex'],ax=ax1)
sns.countplot(x=data['region'],ax=ax2)
sns.countplot(x=data['smoker'],ax=ax3)
sns.countplot(x=data['children'],ax=ax4)


ax1.set_title('Count of observations (Gender wise)')
ax2.set_title('Count of observations (Region wise)')
ax3.set_title('Count of observations (Smnoking or not smoking)')
ax4.set_title('Count of observations (Number of children)')
```

Out[23]:
```
Text(0.5,1,'Count of observations (Number of children)')
```



Observations

1. There are almost equal number of observations for males and females and all the regions
2. Numbere of observations for non smokers are more than that of smokers. hence data set is imabalanced from point of view of this attributte
3. Imbalance is also there for observations for number of children. As the number of children increases number of observations decreases. But this is also possible as higher number of children is generally less obseravable in today's world

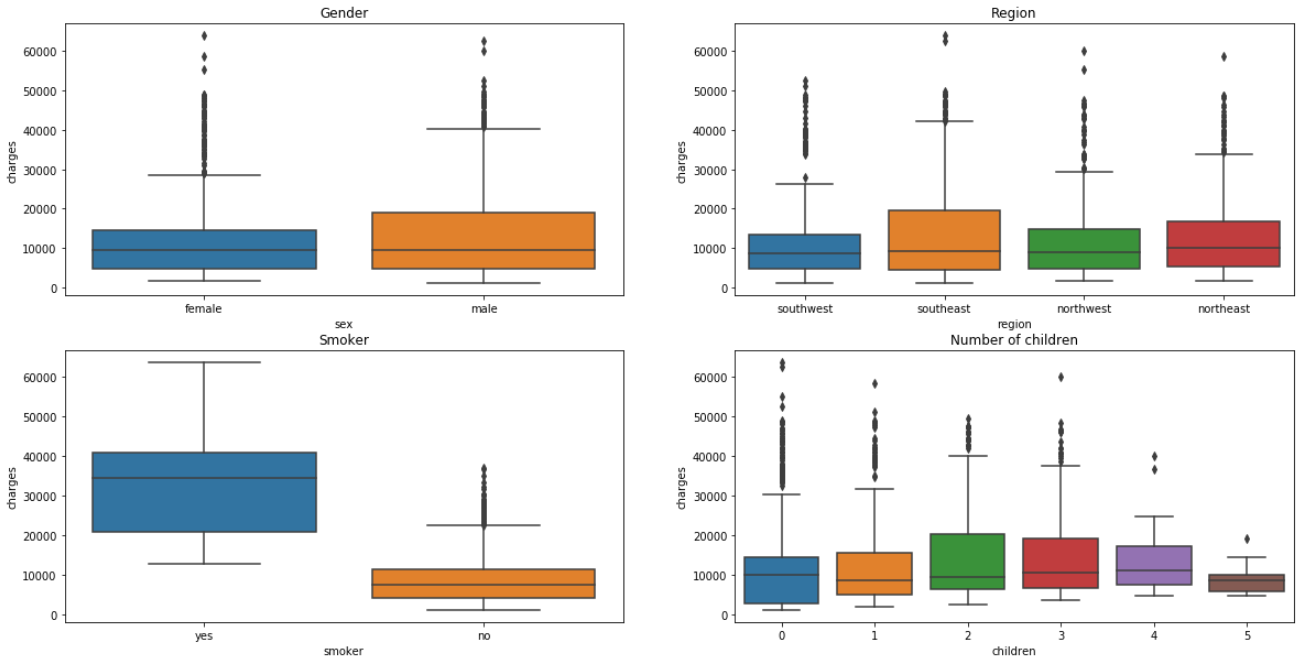
So dataset is fairly balanced except for attribute 'Smoker'

Box Plots

Box plots can also be used for representing categorical data

In [24]:
```
fig, ((ax1, ax2), (ax3,ax4)) = plt.subplots(nrows=2,ncols=2,figsize=(20,10))

sns.boxplot(x="sex", y="charges",  data=data,ax=ax1)
sns.boxplot(x="region", y="charges",  data=data,ax=ax2)
sns.boxplot(x="smoker", y="charges",  data=data,ax=ax3)
sns.boxplot(x="children", y="charges",  data=data,ax=ax4)

ax1.set_title('Gender')
ax2.set_title('Region')
ax3.set_title('Smoker')
ax4.set_title('Number of children')
```

Out[24]: Text(0.5,1,'Number of children')



Observations

Gender

1. Median charges are same for males and females.
2. Maximum charges for male is higher than that of females
3. There are outliers in distribution of charges for both males as well as females.
4. from #3, distribution of charges is positively/ right skewed for males as well as females

Region

1. Median charges for all regions are same
2. There are lot of outliers in distribution of charges for all regions
3. Distribution of charges is not symmetrical and is positively skewed for all regions
4. Maximum charges paid in southeast is more than that of southwest or any other region

Smoker

1. Sokers have very high median charges than n on smokers
2. min and max charges for smokers are very large compared to non smokers
3. There are few outliers in the distribution of charges for non smokers. This implies that some non smokers are paying higher charges than the rest of the non smoker population. This could be because of some other factors
4. SMOKING IS INJURIOUS TO POCKET

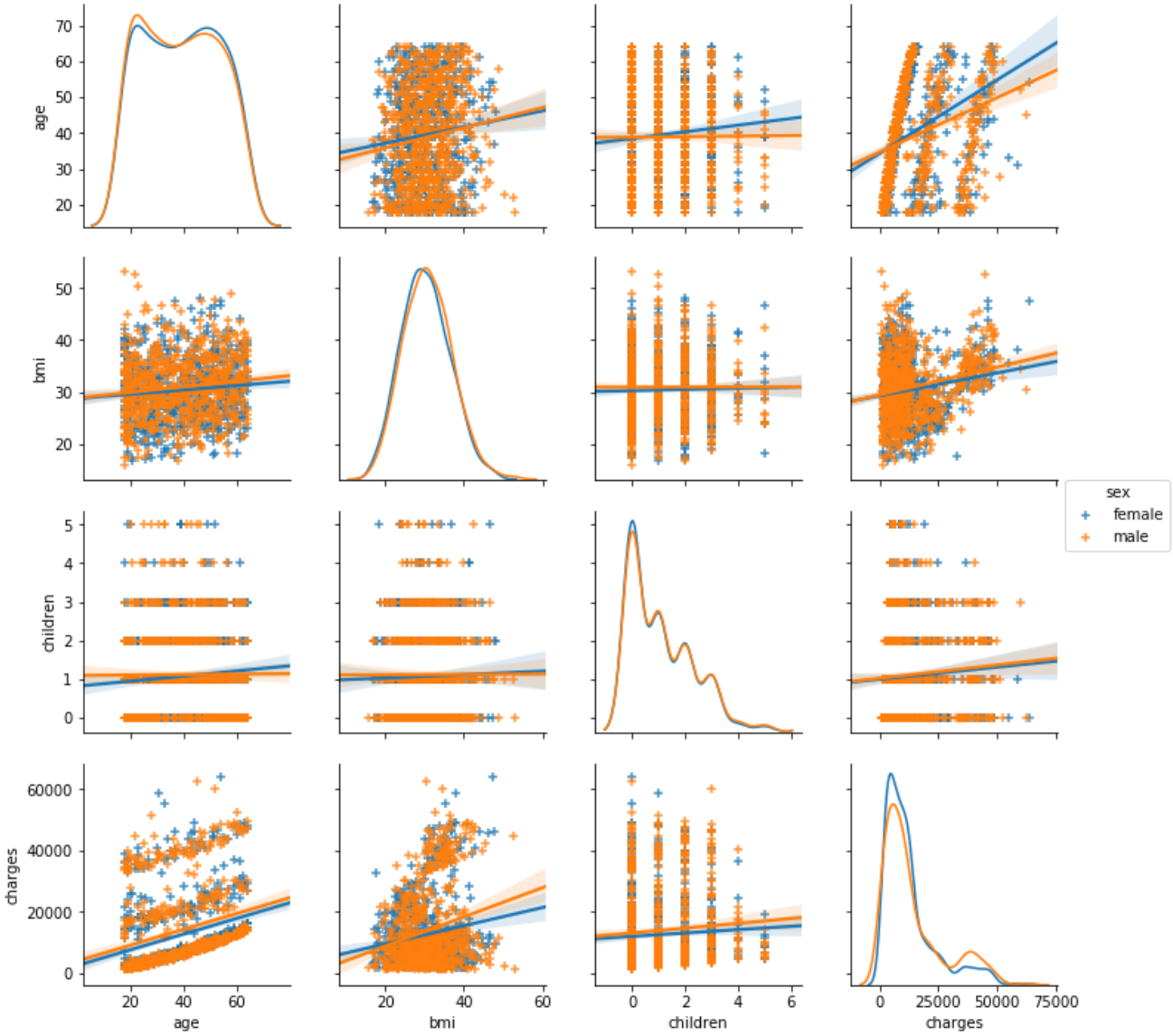
Number of children

1. Median charges are close to each other irrespective of number of children
2. maximum charges is for people with 2 children
3. There is no considerable difference between min charges for any number of children
4. There are lot of outliers across all categories of number of children implying that the data is positively skewed

3.i Pair plots that includes all the columns of the dataframe

In [64]:
```
sns.pairplot(data,kind='reg',hue='sex',diag_kind='kde',markers='+')
#kind='reg' is used to get some kind of trendline to judge the correlation between at
tributes
```

Out[64]: <seaborn.axisgrid.PairGrid at 0x175de0cbc88>

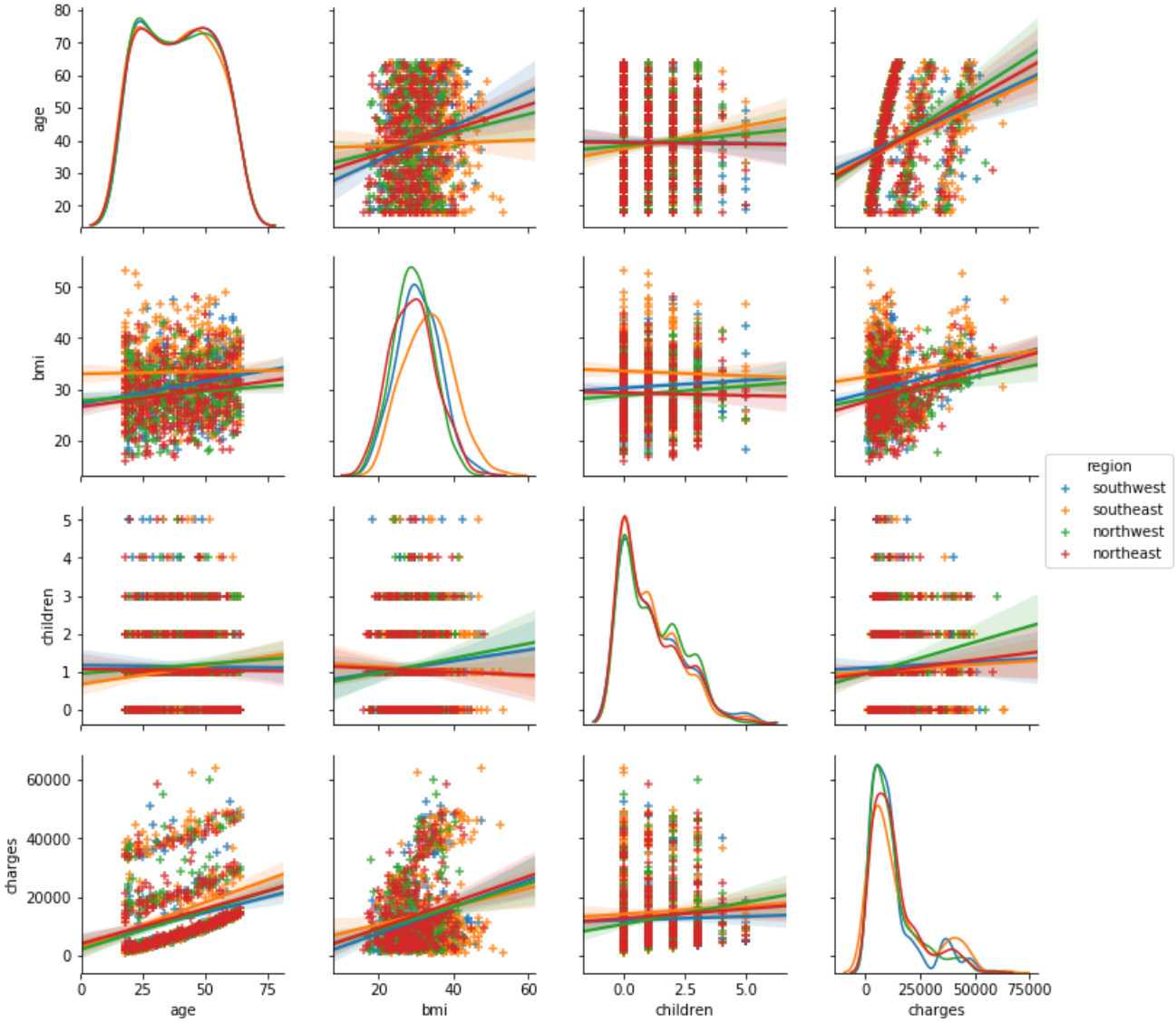


Observations

1. There is a positive correlation between age and charges.
2. There is a seemingly positive correlation between charges and bmi however the spread of charges is more at all bmi values.
3. There seems to be no difference in spread of charges when number of children is upto 3. However, charges drops for number of children 4 or 5
4. Also,spread of BMI decreases a little bit with increase in the number of children however its almost similar for any number of children
5. There is no visual evidence of any correlation between age and bmi. The spread of bmis is almost similar across all age groups

```
In [63]: sns.pairplot(data,kind='reg',hue='region',diag_kind='kde',markers='+')
```

```
Out[63]: <seaborn.axisgrid.PairGrid at 0x175e52a6d30>
```

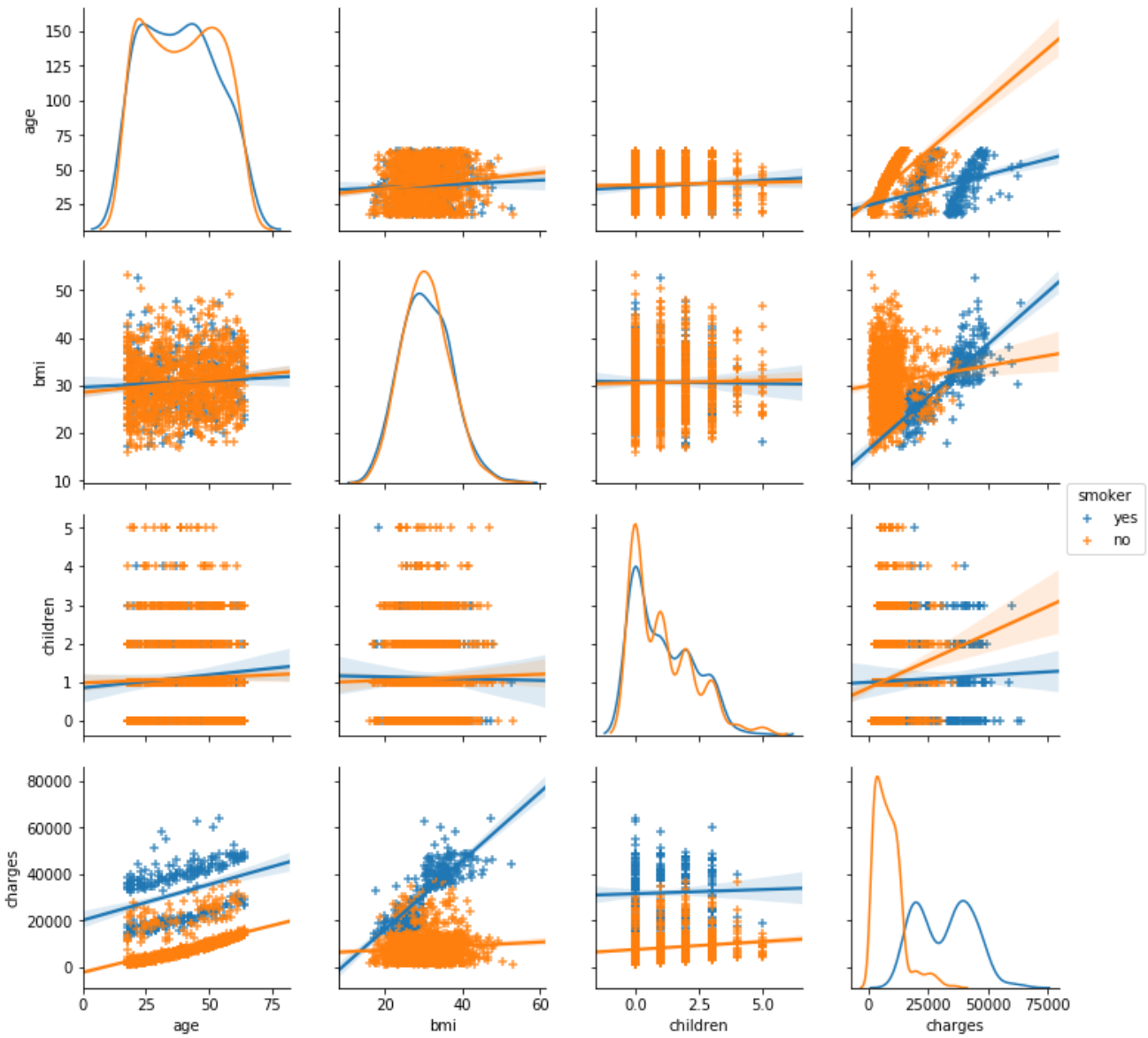


Observations

1. There is no impact of different regions on distribution of charges
2. Distribution of charges are positively skewed for all the regions
3. Regions have no impact charges vs age distributions

```
In [57]: sns.pairplot(data,kind='reg',hue='smoker',diag_kind='kde',markers='+')
```

```
Out[57]: <seaborn.axisgrid.PairGrid at 0x175e2b07c50>
```



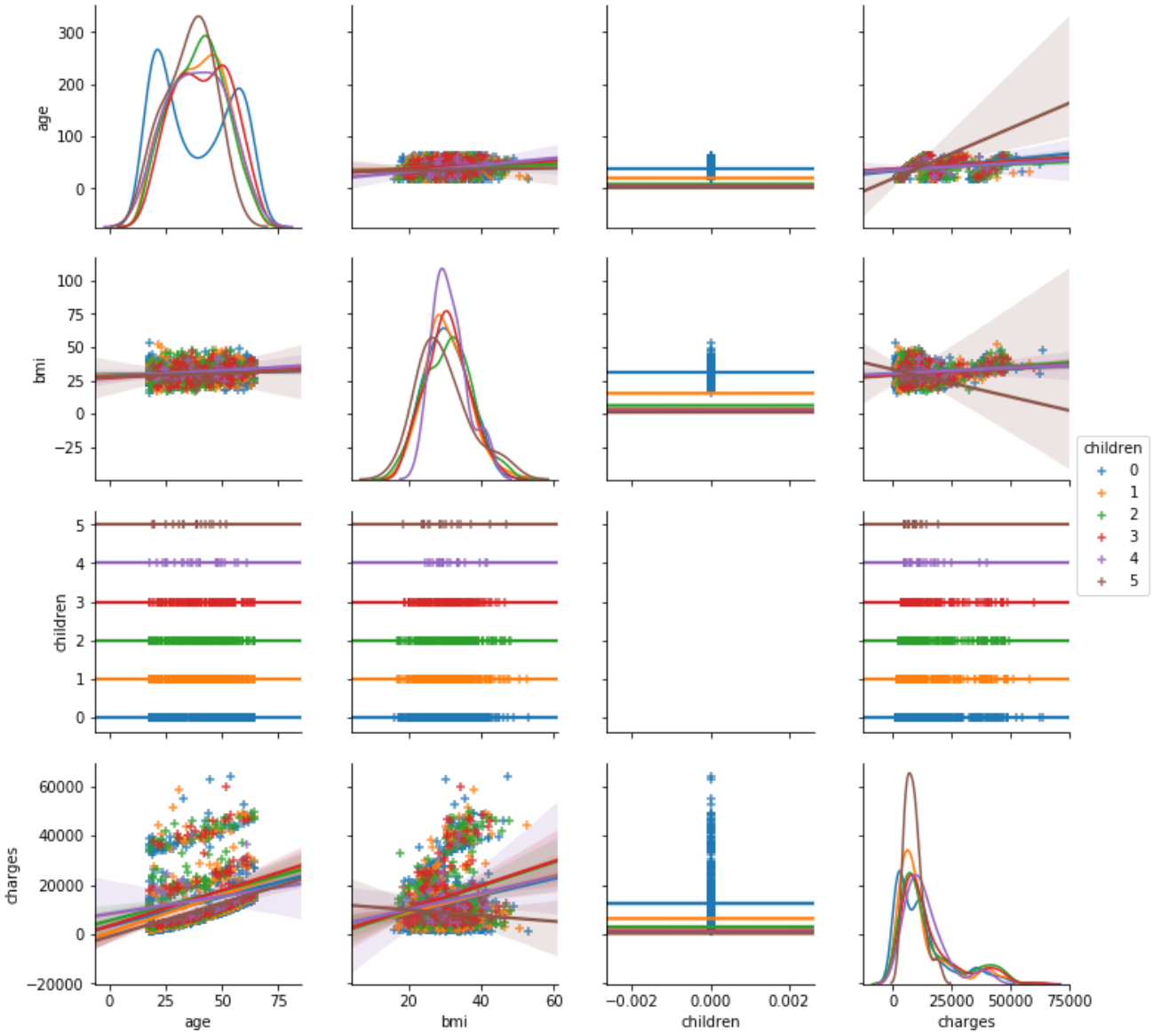
Observations

1. There is a good relationship between Charges and age for both smokers and non smokers. This can be easily distinguished and higher for smokers for every age group
2. Relationship between charges and bmi is also linear. Its low and similar for non smokers even though vmi increases. For smokers as the bmi increases the increase in charges are quite steap
3. bmi has a normal distribution for both smokers as well as non smokers
4. no stark differentioation between bmi for smokers and non smokers at every age group
5. Density distribution for charges for smokers vs non smokers displays significant difference in the distribution of charges

In [59]:
```
sns.pairplot(data,kind='reg',hue='children',diag_kind='kde',markers='+')
```

```
C:\Users\sande\Anaconda3\lib\site-packages\statsmodels\nonparametric\kde.py:488: Run
timeWarning: invalid value encountered in true_divide
  binned = fast_linbin(X, a, b, gridsize) / (delta * nobs)
C:\Users\sande\Anaconda3\lib\site-packages\statsmodels\nonparametric\kdetools.py:34:
RuntimeWarning: invalid value encountered in double_scalars
  FAC1 = 2*(np.pi*bw/RANGE)**2
```

Out[59]: <seaborn.axisgrid.PairGrid at 0x175dd960d30>



Observations

1. There is no impact of number of children on charges wrt to age. As age increases charges increases steadily irrecpective of the number of children
2. Distribution of chaereges is almost same for all categories and is positively skewed for almost all categories

Section 4

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

To check weather charges of smokers differ significantly from charges if non smokers we will perform a 2 sample t test

We will use comparison of means for this purpose

Lets assume Null Hypothesis Ho is Mean charges of smokers = Mean charges of non smokers Alternate Hypothesis Ha = Mean Charges of smokets # mean charges of non smokers

These hypothesis will be tested at the significance level of 5% i.e critical value will be 0.05

If p-value < 0.05 ==> we reject the null hypothesis and state that there are sufficient statistical evidence that mean charges for smokers are significantly different from non smokers

For this test we will breakdown our dataset into 2 separate datasets

1. Charges for smokers named 'smoker'
2. Charges for non smokers named 'non_smoker'

In [36]:
```
#division of dataset into smoker and non_smokers
smoker = data[data['smoker']=='yes']['charges']
non_smoker = data[data['smoker']=='no']['charges']
smoker.head(5)
```

Out[36]:
```
0     16884.9240
11    27808.7251
14    39611.7577
19    36837.4670
23    37701.8768
Name: charges, dtype: float64
```

In [37]:
```
non_smoker.head(5)
```

Out[37]:
```
1     1725.55230
2     4449.46200
3    21984.47061
4     3866.85520
5     3756.62160
Name: charges, dtype: float64
```

Assumption: This test assumes that the variances of the 2 samples are equal.

In [38]:
```
t_statistic, p_value = ttest_ind(smoker, non_smoker)
print(t_statistic, p_value)
```

```
46.64479459840305 1.4067220949376498e-282
```

Observations

1. pvalue is very low and < significance level of 0.05
2. Hence we reject the null Hypothesis
3. There is enough statistical evidence that Mean charges for smokers are significantly higher than mean charges for non smokers

PS: Although it was assumed that variances are equal hoever, when lavene test was conducted, the variances were not found to be equal. The pvalue of lavene test was significantly lower than 0.05 and Ho that variances are equal shall be rejected.

See the results of the test below

In [39]:
```
t_stat,p_value = levene(smoker,non_smoker)
print("The p value of the lavene test on smokers and non smokers is :",p_value)
```

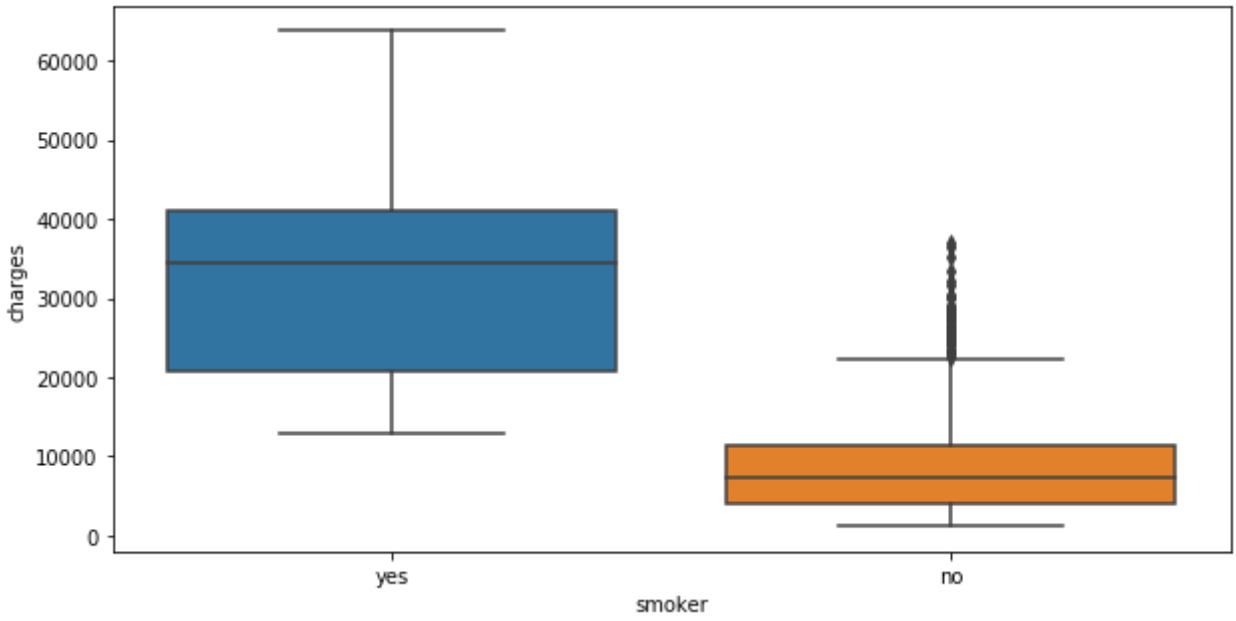
```
The p value of the lavene test on smokers and non smokers is : 1.670117565125241e-66
```

Statistical evidence above can also be validated visually using the boxplot

Details description of observations of boxplot is already available in section 3.h It can be clearly seen that there is a considerable difference in mean charges of smokers and non smokers

In [53]:
```
fig, ax = plt.subplots(figsize=(10,5))
sns.boxplot(x="smoker", y="charges",  data=data,ax=ax)
ax3.set_title('Smoker')
```

Out[53]:
```
Text(0.5,1,'Smoker')
```



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

We will again use 2 sample t test to test the similarity between bmi of males and females

lets assume:

Null Hypothesis Ho : mean bmi of male = mean bmi of female

Alternate Hypothesis Ha: mean bmi of male # mean bmi of female

These hypothesis will be tested at the significance level of 5% i.e critical value will be 0.05

If p-value < 0.05 ==> we reject the null hypothesis and state that there are sufficient statistical evidence that mean charges for smokers are significantly different from non smokers¶

For this test we will breakdown our dataset into 2 separate datasets

1. bmi of males
2. bmi of females

In [40]:
```
#division of dataset into smoker and non_smokers
bmi_male = data[data['sex']=='male']['bmi']
bmi_female = data[data['sex']=='female']['bmi']
```

In [41]:
```
#shape of 2 datasets
print(bmi_male.shape, bmi_female.shape)
```

```
(675,) (662,)
```

Assumption

This test assumes that the 2 samples have equal variances perform levene test to confirm the same

Ho : variance of sample bmi_male = variance of sample bmi_female

Ha: variance of sample bmi_male # variance of sample bmi_female

In [42]:
```
#levene test
t_stat,p_val = levene(bmi_male,bmi_female)
p_val
```

Out[42]:
```
0.9216570820140155
```

0.921 > 0.05 Hence we fail to reject the null hypothesis that the 2 samples have equal variances

we can perform 2 sample t test as below

In [43]:
```
t_statistic,p_value = ttest_ind(bmi_male,bmi_female)
print(p_value)
```

```
0.08991704324931137
```

pvalue(0.089) > significance level(0.05) ==> we fail to reject null hypothesis Ho

Hence there is no statistical evidence that the bmi of males differs significantly from bmi of females. Thus means of bmi of males and females are quite similar.

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

We will use Z test of proportions to verify that the proprtions of male smokers is equal to propoertions of female populations

proportion of male smokers = male smokers/total male populations proportion of female smokers = female smokers/total female populations

Null Hypothesis Ho : proportion of male smokers = proportion of female smokers

Alternat hypothesis Ha : proportion of male smokers # proportion of female smokers

For this test we will have to seaprate male and female smokers and calculate their proportions befre we can apply proportions_ztest

In [44]:
```
male_smokers = data[data['sex']=='male']['smoker'].value_counts()[1]
total_males = data[data['sex']=='male']['sex'].value_counts()[0]
pMaleSmoker = round(male_smokers/total_males,2)
print(f"Total proportion of male smokers is : {pMaleSmoker}%")
```

```
Total proportion of male smokers is : 0.24%
```

In [45]:
```
female_smokers = data[data['sex']=='female']['smoker'].value_counts()[1]
total_females = data[data['sex']=='female']['sex'].value_counts()[0]
pFemaleSmoker = round(female_smokers/total_females,2)
print(f"Total proportion of female smokers is : {pFemaleSmoker}%")
```

```
Total proportion of female smokers is : 0.17%
```

In [46]:
```
z_stat,p_value = proportions_ztest([male_smokers,female_smokers],[total_males,total_f
emales])
print(p_value)
```

```
0.005098746217145657
```

Observations:

Since pvalue(0.0050) < 0.05 ==> we reject the null hypothesis and can say that proportion of male smokers is different than that of female smokers

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

To comparer the bmi of women with 0,1,2 children we will use ANOVA

1. Prepare separate datasets for female with 0,1 and 2 children

Null Hypothesis:

Ho = mean bmi of women with 0,1,2 children are same

Ha = mean bmi of women with 0,1,2 children are different or mean bmi of atleast 2 groups is different

Since we need to compare 3 samples here, we can use ANOVA We will test the hypothiesis at 95% significance level i.e. critical value will be 0.05

In [47]:
```
female_0 = data[(data['sex']=='female') & (data['children'] == 0)]['bmi']
female_1 = data[(data['sex']=='female') & (data['children'] == 1)]['bmi']
female_2 = data[(data['sex']=='female') & (data['children'] == 2)]['bmi']
```

Above we have separated the series for women with 0,1 and 2 children

Below we will convert them into separate dataframes and then concatenate to create a single dataframe

In [48]:
```
mean_df = pd.DataFrame()

df1             = pd.DataFrame({'Children': '0', 'bmi':female_0})
df2             = pd.DataFrame({'Children': '1', 'bmi':female_1})
df3             = pd.DataFrame({'Children': '2', 'bmi':female_2})

mean_df = mean_df.append(df1)
mean_df = mean_df.append(df2)
mean_df = mean_df.append(df3)
```

In [52]:
```
mod = ols('bmi ~ Children', data = mean_df).fit()
aov_table = sm.stats.anova_lm(mod, typ=2)
print(aov_table)
```

```
                sum_sq     df         F    PR(>F)
Children     24.590123    2.0  0.334472  0.715858
Residual  20695.661583  563.0       NaN       NaN
```

Observations

1. Here key result is p-value = 0.715
2. significance level is 0.05
3. now since 0.715>0.05 i.e p-value > 0.05
4. hence we fail to reject null hypothesis Thus distribution of bmi is same across women with 0,1,2 children