Apollo Hospitals was established in 1983, renowned as the architect of modern healthcare in India. As the nation's first corporate hospital, Apollo Hospitals is acclaimed for pioneering the private healthcare revolution in the country.

As a data scientist working at Apollo 24/7, the ultimate goal is to tease out meaningful and actionable insights from Patient-level collected data.

You can help Apollo hospitals to be more efficient, to influence diagnostic and treatment processes, to map the spread of a pandemic.

One of the best examples of data scientists making a meaningful difference at a global level is in the response to the COVID-19 pandemic, where they have improved information collection, provided ongoing and accurate estimates of infection spread and health system demand, and assessed the effectiveness of government policies.

Lets Analyse the data

In [ ]:
```
## import libraries

import warnings
warnings.filterwarnings("ignore")
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from IPython.core.display import display, HTML
display(HTML("<style>.container { width:100% !important; }</style>"))
from IPython.core.display import display, HTML
display(HTML("<style>.container { width:100% !important; }</style>"))
import matplotlib_inline
matplotlib_inline.backend_inline.set_matplotlib_formats('svg')
from scipy.stats import ttest_ind
from scipy.stats import levene, ks_2samp
from sklearn.preprocessing import LabelEncoder
from scipy import stats
from statsmodels.graphics.gofplots import qqplot_2samples
plot_cmap = 'gist_heat'
plot_color = '#EE3C26'
```

In [64]:
```
## reading dataset
df = pd.read_csv(r"C:\Users\sahil.bansal\Desktop\scaler_apollo_hospitals.csv")
```

In [3]:
```
## getting the shape of dataset
df.shape
```

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

There are 1338 rows and 7 columns in the data

In [ ]:

In [4]:
```
## checking info of the data
df.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   smoker                   1338 non-null   object
 3   region                   1338 non-null   object
 4   viral load               1338 non-null   float64
 5   severity level           1338 non-null   int64
 6   hospitalization charges  1338 non-null   int64
dtypes: float64(1), int64(3), object(3)
memory usage: 73.3+ KB
```

We can see that there are 1338 data points and 7 features and there are no null values and there are both categorical and numerical values in the data

In [ ]:

In [5]:
```
## describe the data
df.describe()
```

Out[5]:

	age	viral load	severity level	hospitalization charges
count	1338.000000	1338.000000	1338.000000	1338.000000
mean	39.207025	10.221233	1.094918	33176.058296
std	14.049960	2.032796	1.205493	30275.029296
min	18.000000	5.320000	0.000000	2805.000000
25%	27.000000	8.762500	0.000000	11851.000000
50%	39.000000	10.130000	1.000000	23455.000000
75%	51.000000	11.567500	2.000000	41599.500000
max	64.000000	17.710000	5.000000	159426.000000

we can see there are not much outliers there are some outliers in hospitalization charges that we will handle in further analysis.

In [ ]:

In [6]:
```
## checking for null values
df.isnull().sum()
```

Out[6]:
```
age                        0
sex                        0
smoker                     0
region                     0
viral load                 0
severity level             0
hospitalization charges    0
dtype: int64
```

We can see there are no null values in the data

In [ ]:

In [7]:
```
## value count of data
cat_cols = ['sex','smoker','region','severity level']
df[cat_cols].apply(lambda x: x.value_counts()).T.stack()
```

Out[7]:
```
sex             female        662.0
                male          676.0
smoker          no           1064.0
                yes           274.0
region          northeast     324.0
                northwest     325.0
                southeast     364.0
                southwest     325.0
severity level  0             574.0
                1             324.0
                2             240.0
                3             157.0
                4              25.0
                5              18.0
dtype: float64
```

data looks good there are no miscellaneous values

In [ ]:

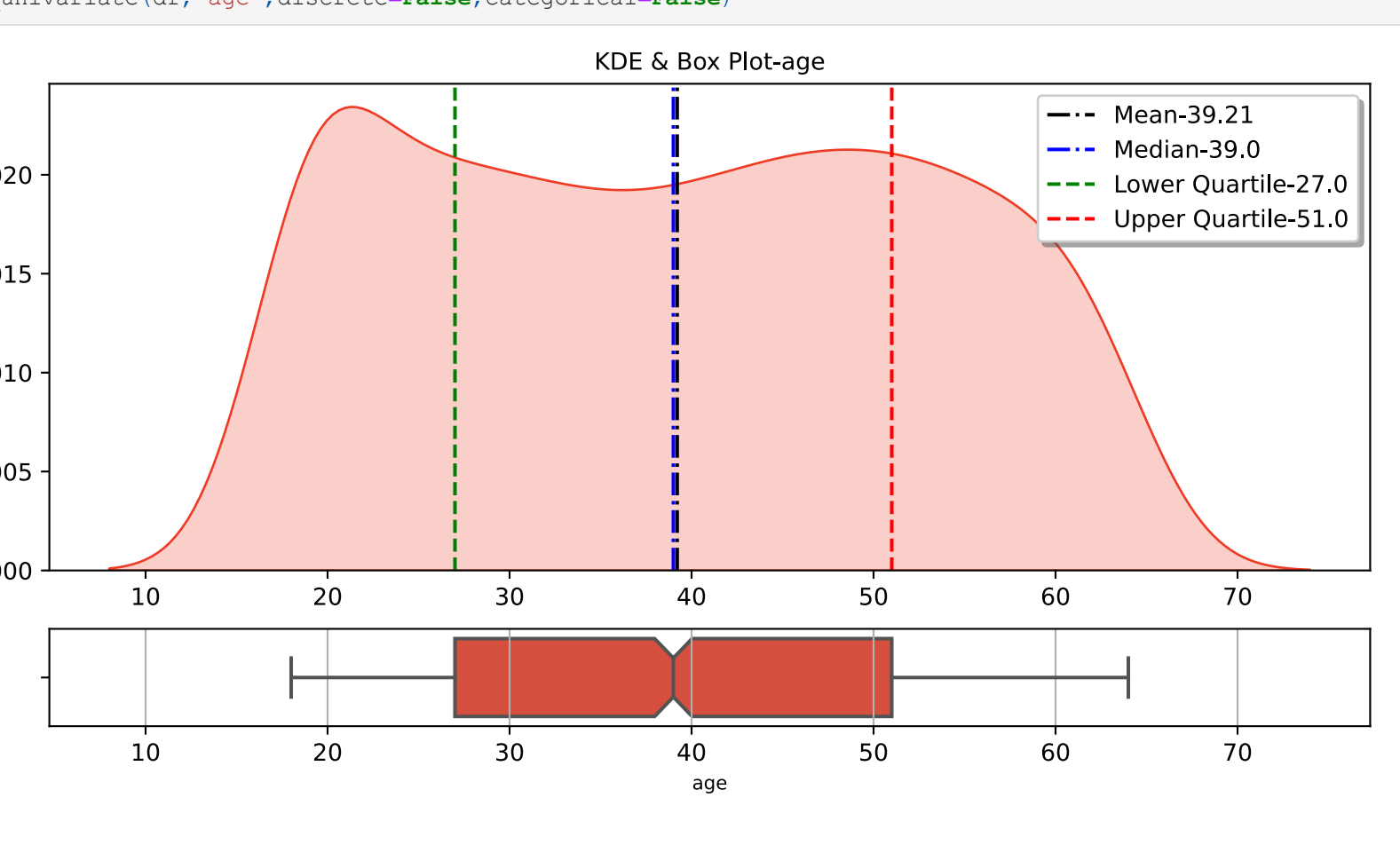
Univariate Analysis

In [8]:
```
## generic function to plot the graphs

def plot_univariate(df,column,discrete=False,categorical=False,top ='all'):

    plot_cmap = 'gist_heat'
    plot_color = '#EE3C26'
    if categorical:
        # get value counts
        df_value_count = df[column].value_counts()

        # if all, set top as length of df
        if top =='all':
            top= len(df_value_count)
        df_value_count= df_value_count.iloc[:top]

        #draw count plot
        plt.figure(figsize= (10,5))
        ax= sns.countplot(x= column, data= df, palette= plot_cmap,order= df_value_count.index)

        #annotate values
        for p in ax.patches:
            #if value of bar is less than 10% of max value, change the alignment
            ax.annotate(f'\n{p.get_height()}', (p.get_x()+p.get_width()*0.4, p.get_height()*1.01),
                        ha= 'center', va= 'bottom', color= p.get_facecolor(), rotation= 90, size= 8)

        #increase ylim to accomodate annotations
        ax.set_ylim([0, ax.get_ylim()[1]*1.11])

        #add colorbar manually
        norm= plt.Normalize(df_value_count.min(), df_value_count.max())
        cmap = plt.get_cmap(f'{plot_cmap}_r')
        sm= plt.cm.ScalarMappable(cmap= cmap, norm= norm)
        sm.set_array([])
        ax.figure.colorbar(sm)
        ax.set_xticklabels(ax.get_xticklabels(),rotation=90)

        #add title and x-axis grid
        title = f"Counts of Top {top} {column}"
        plt.title(title, fontsize=8)
        ax.xaxis.grid()
        plt.show()

    else:
        # get statistical parameters
        stat_params = df[column].describe().round(2)

        fig,axes = plt.subplots(2, 1,figsize=(10,5), gridspec_kw={'height_ratios': [5, 1]})

        #plot only histogram if discrete variable else histogram with kde
        if discrete:
            cm = plt.cm.get_cmap(f'{plot_cmap}_r')
            bins = np.histogram_bin_edges(df[column], bins='auto')
            n, bins, patches = axes[0].hist(df[column], bins = bins, linewidth=1.2,edgecolor='black')

            #https://stackoverflow.com/questions/23061657/plot-histogram-with-colors-taken-from-colormap
            bin_centers = 0.5 * (bins[:-1] + bins[1:])
            col = bin_centers - min(bin_centers)
            col /= max(col)

            for c, p in zip(col, patches):
                plt.setp(p, 'facecolor', cm(c))
        else:
            sns.kdeplot(data=df, x=column, ax=axes[0],color=plot_color,shade=True)

        # add lines showing stat params mean, median, 25% quant, 75% quant
        axes[0].axvline(x=stat_params['mean'], color='k',
                    linestyle='-.',label = f"Mean-{stat_params['mean']}")
        axes[0].axvline(x=stat_params['50%'], color='b',
                    linestyle='-.',label = f"Median-{stat_params['50%']}")
        axes[0].axvline(x=stat_params['25%'], color='g',
                    linestyle='--',label = f"Lower Quartile-{stat_params['25%']}")
        axes[0].axvline(x=stat_params['75%'], color='r',
                    linestyle='--',label = f"Upper Quartile-{stat_params['75%']}")
        axes[0].legend(fancybox= True, shadow= True, prop={'size':10})

        #titles and labels
        axes[0].set_xlabel('',fontsize = 8)
        axes[0].set_ylabel('Density',fontsize = 8)
        axes[0].set_title(f'KDE & Box Plot-{column}',fontsize = 10)
#          axes[0].xaxis.grid(True)

        #plot box plot
        sns.boxplot(x=column, data=df, ax=axes[1],color=plot_color,notch= True)

        #set edge and face colours
        for i,box in enumerate(axes[1].artists):
            fc = box.get_facecolor()
            box.set_edgecolor(plot_color)
            box.set_facecolor(mpl.colors.to_rgba(fc, 0.3))

            for j in range(6*i,6*(i+1)):
                axes[1].lines[j].set_color(plot_color)

        #set xlables of boxplot same as KDE plot
        axes[1].set_xlim(axes[0].get_xlim())
        axes[1].set_xlabel(column,fontsize = 8)
        axes[1].xaxis.grid(True)

        plt.show()
```

In [ ]:

1. Analysis of Age

In [9]:
```
plot_univariate(df,'age',discrete=False,categorical=False)
```

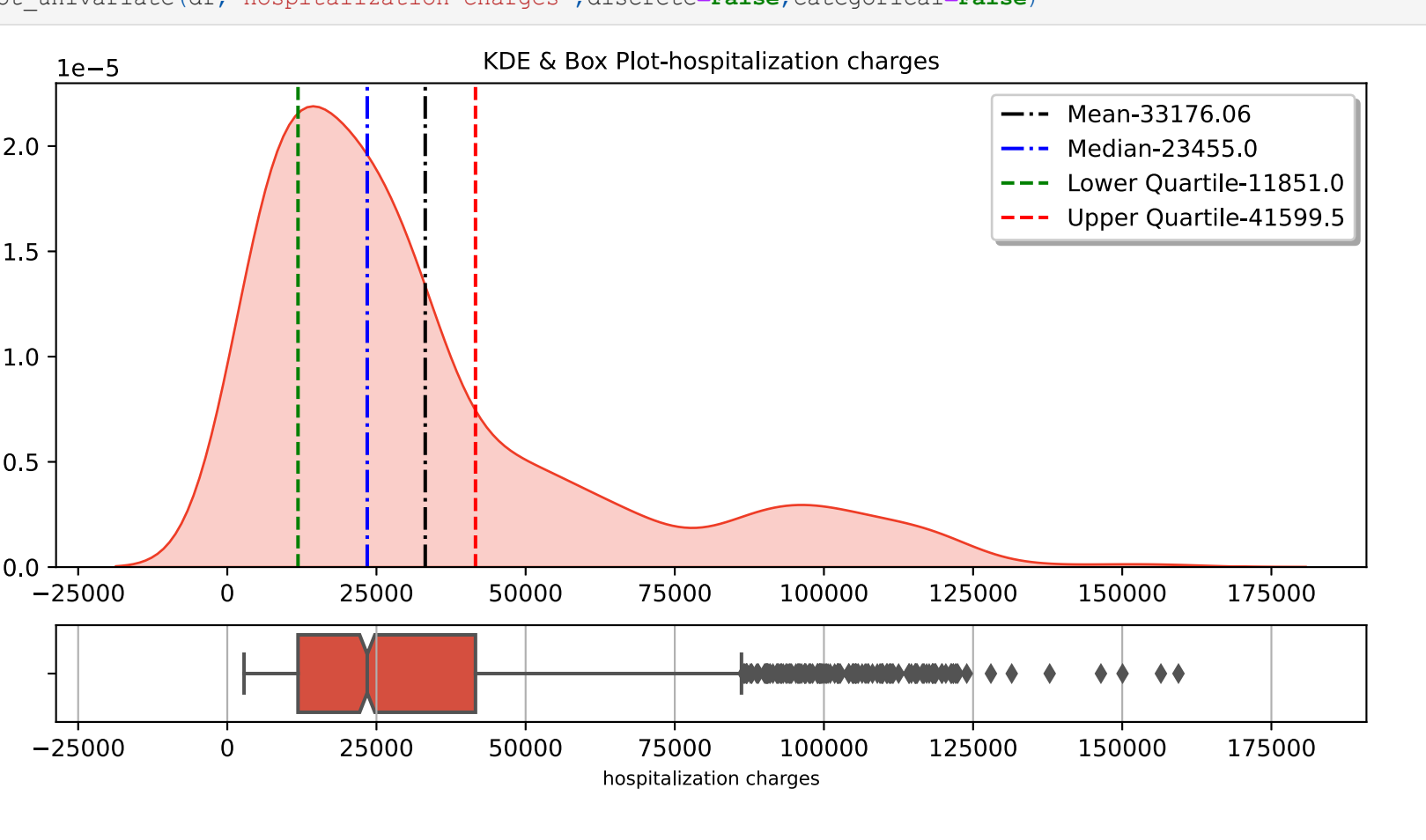


1. Mean value of age in the data is around 39.21 and it is almost normally distributed.
2. There are no outliers in the data as shown in the box plot.

In [ ]:

2. Analysis of Hospitalization Charges

In [10]:
```
plot_univariate(df,'hospitalization charges',discrete=False,categorical=False)
```

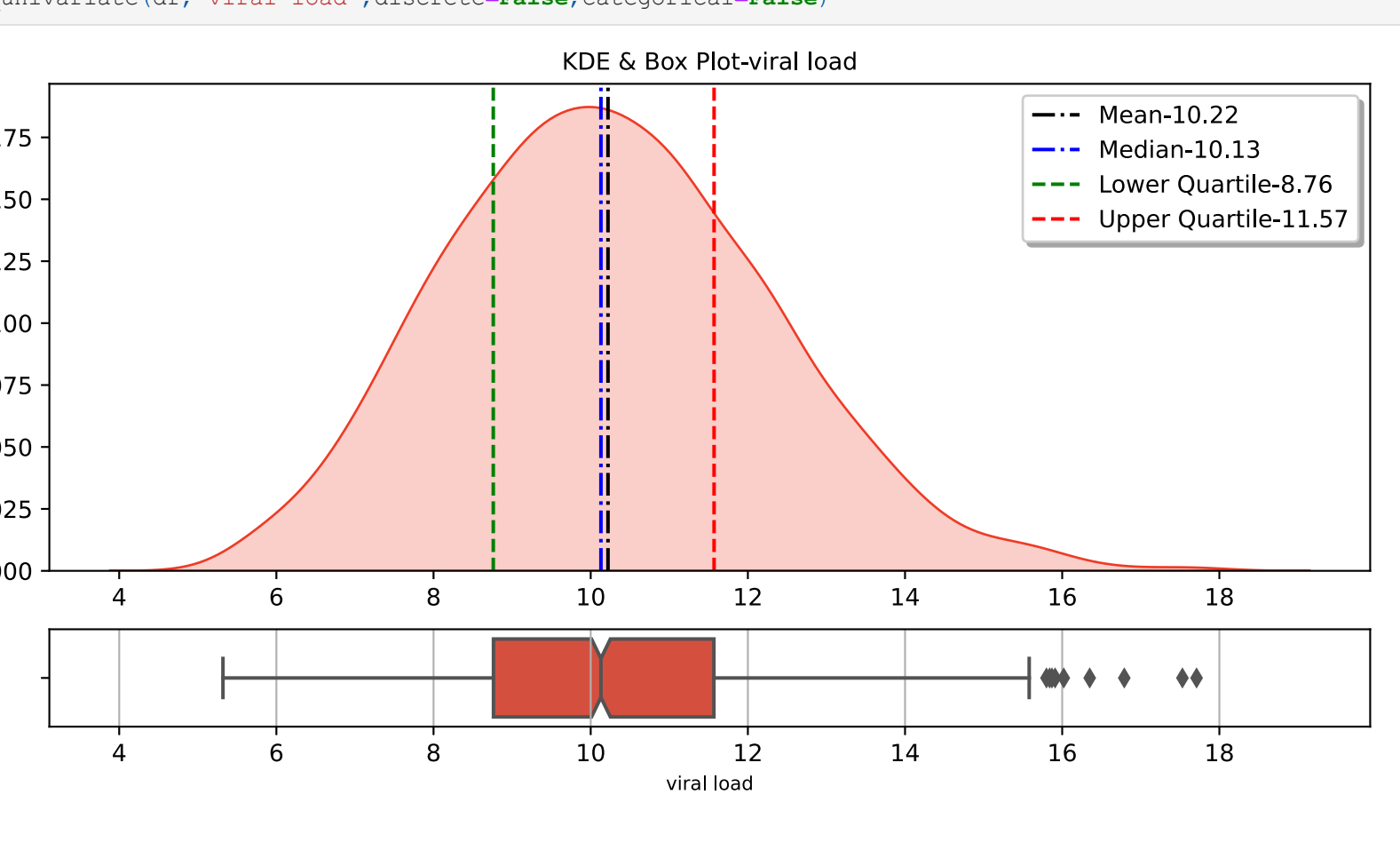


1. Mean value of Hospitalization charges is 33176 and there are outliers in the data that we will handle.
2. Data is not normally distributed, it is left skweed.

In [ ]:

3. Analysis of Viral Load

In [11]:
```
plot_univariate(df,'viral load',discrete=False,categorical=False)
```

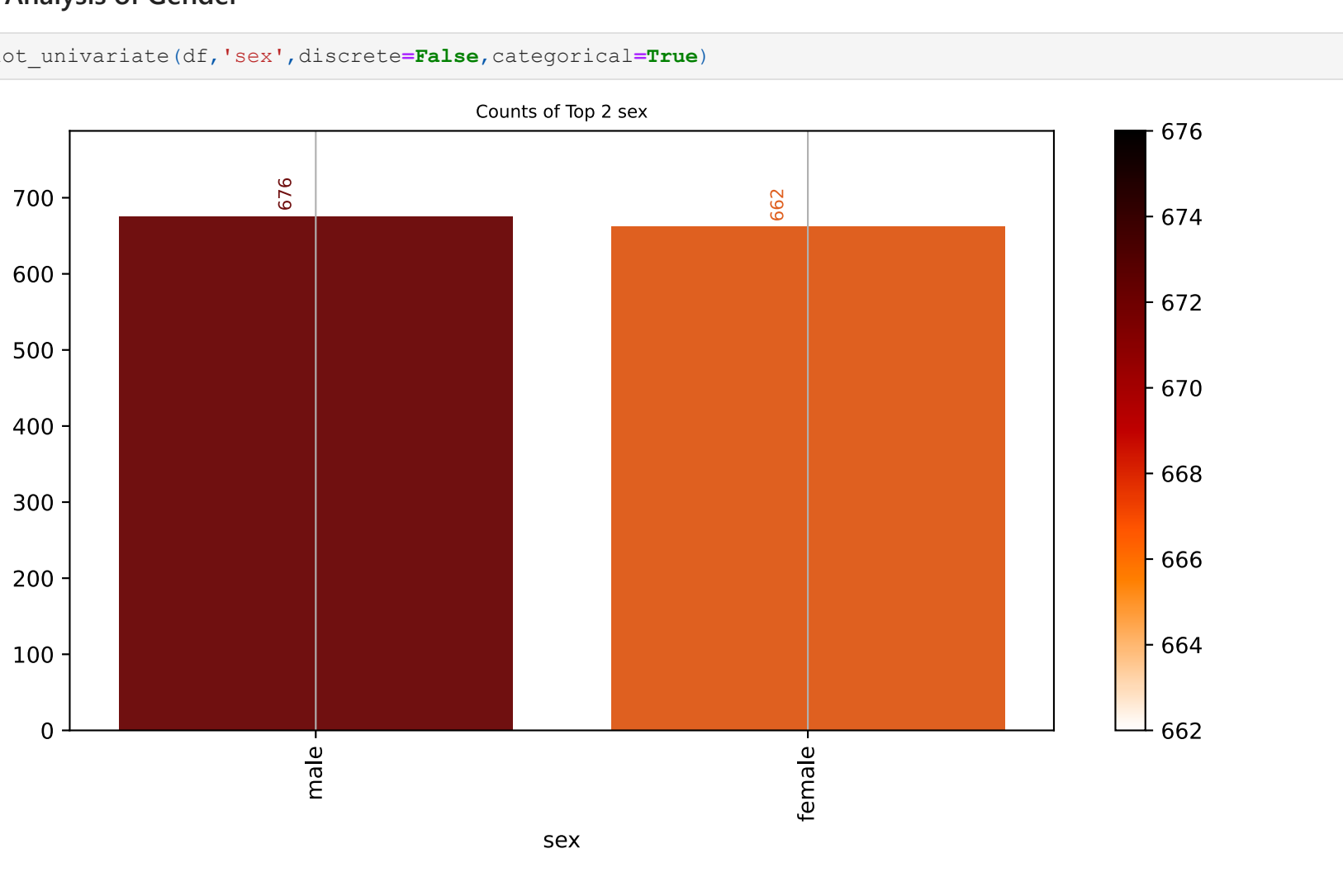


1. Mean value of viral load is 10.22 and data is almost normally distributed.
2. As we can see in the box plot there are some outliers in the data.

In [ ]:

4. Analysis of Gender

In [12]:
```
plot_univariate(df,'sex',discrete=False,categorical=True)
```

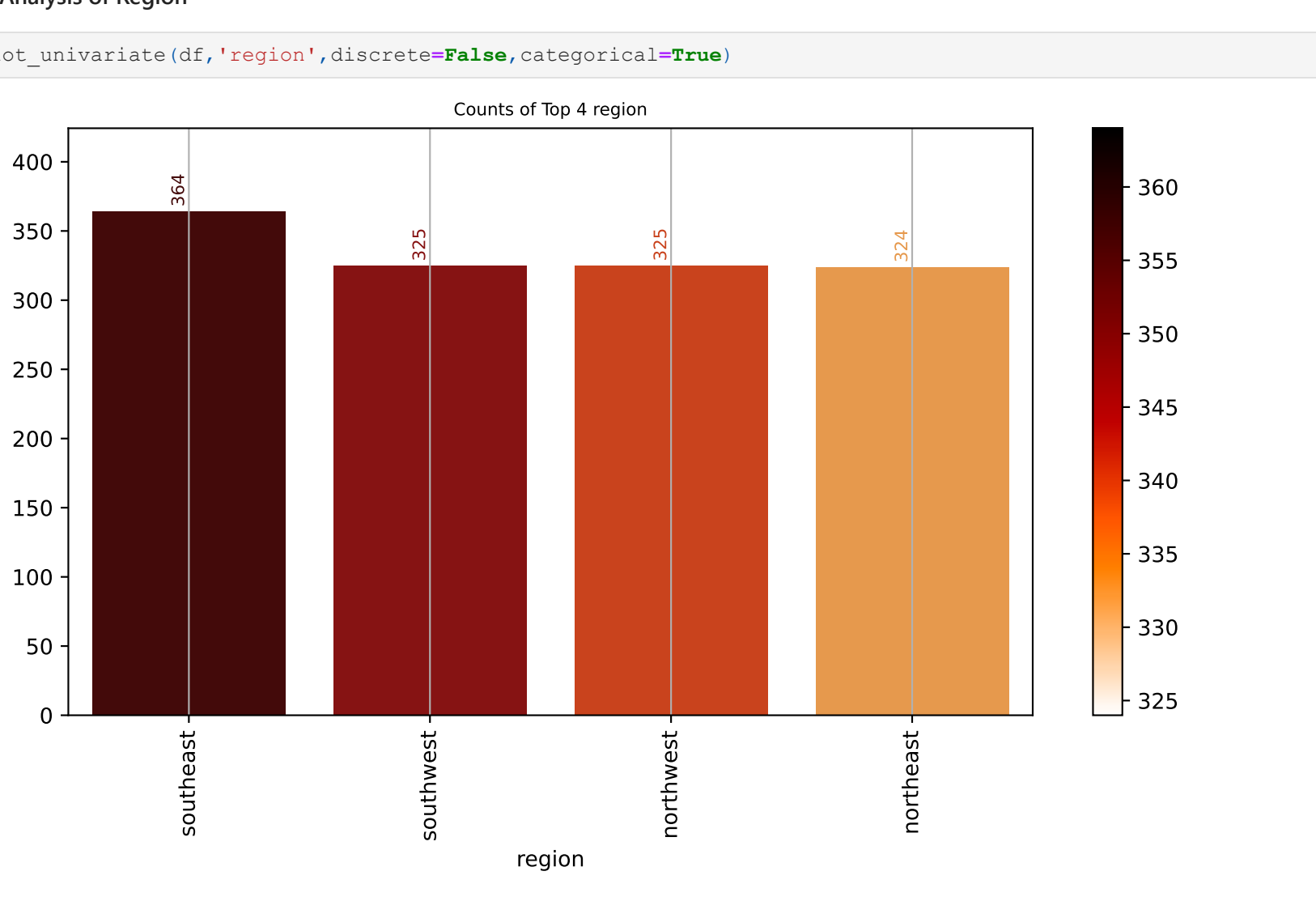


1. There are 676 Male and 662 Female data points, we can say that data is balanced.

In [ ]:

5. Analysis of Region

In [13]:
```
plot_univariate(df,'region',discrete=False,categorical=True)
```

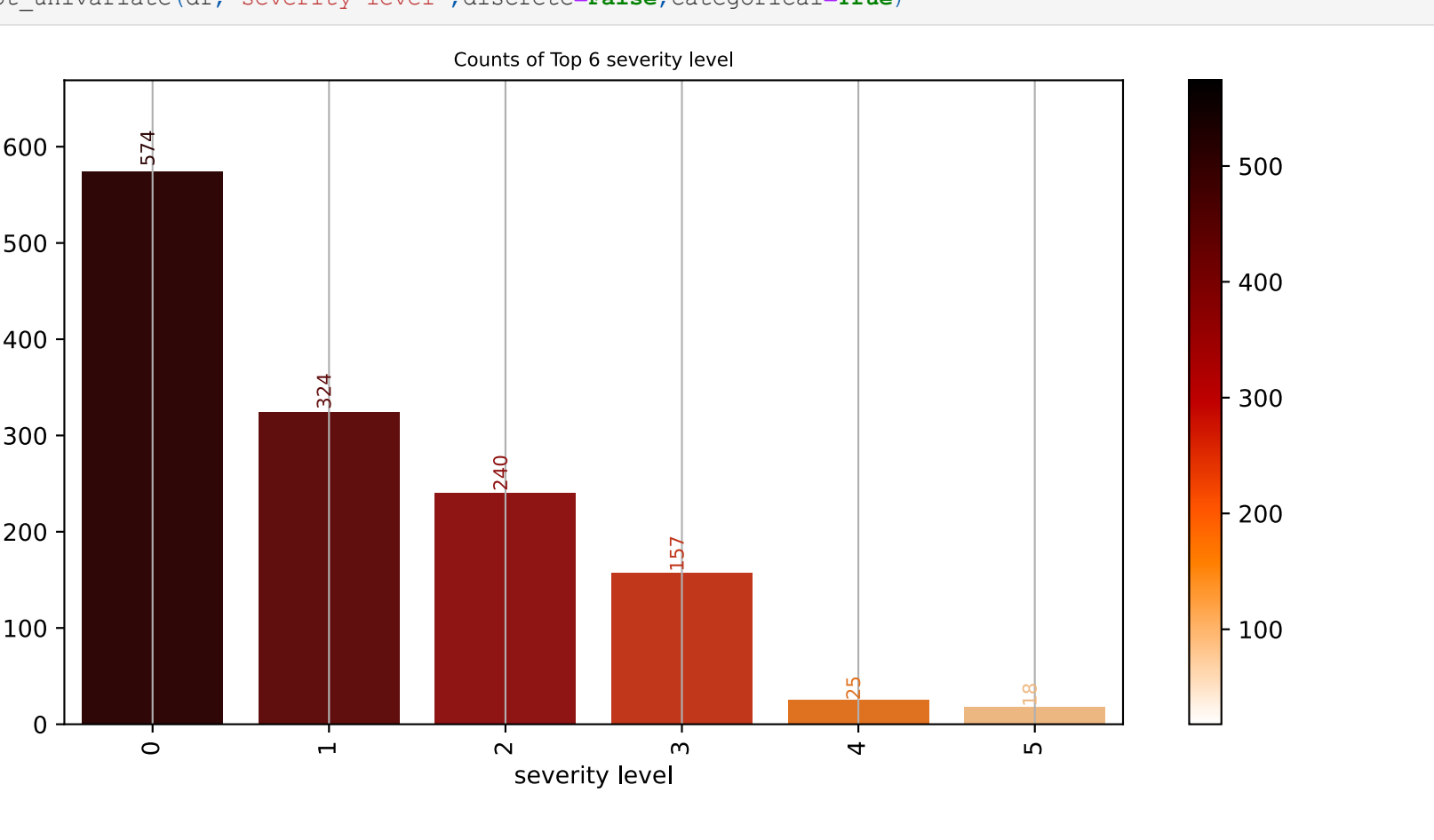


1. In southwest region there are most number of people i.e. 364.
2. In rest of the regions there is not much difference between the data.

In [ ]:

6. Analysis of Severity Level

In [14]:
```
plot_univariate(df,'severity level',discrete=False,categorical=True)
```

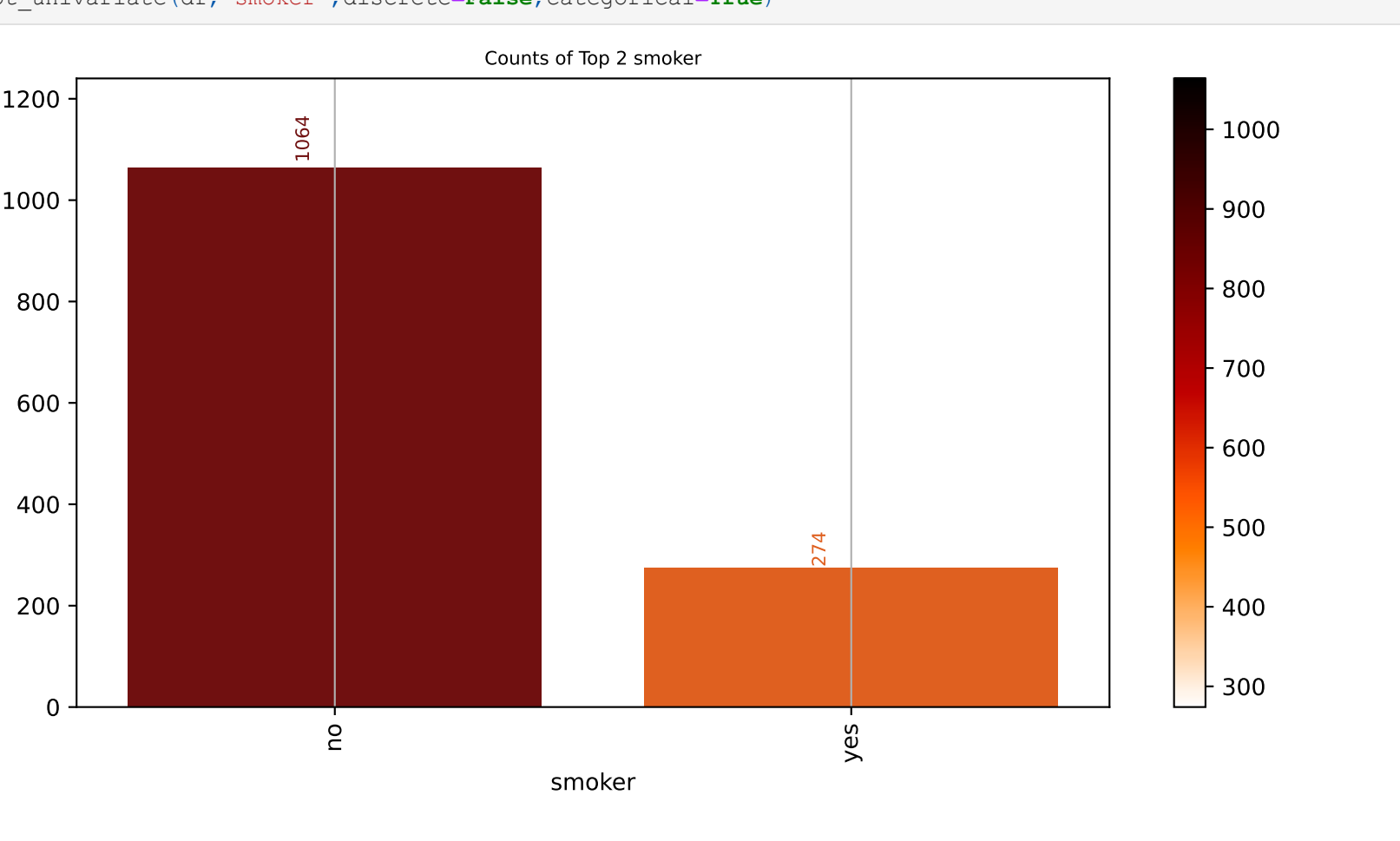


1. It is shown in the data the most of the people has severity level of 0 and least is 5 among the peoples.

In [ ]:

7. Analysis of Smoker

In [15]:
```
plot_univariate(df,'smoker',discrete=False,categorical=True)
```

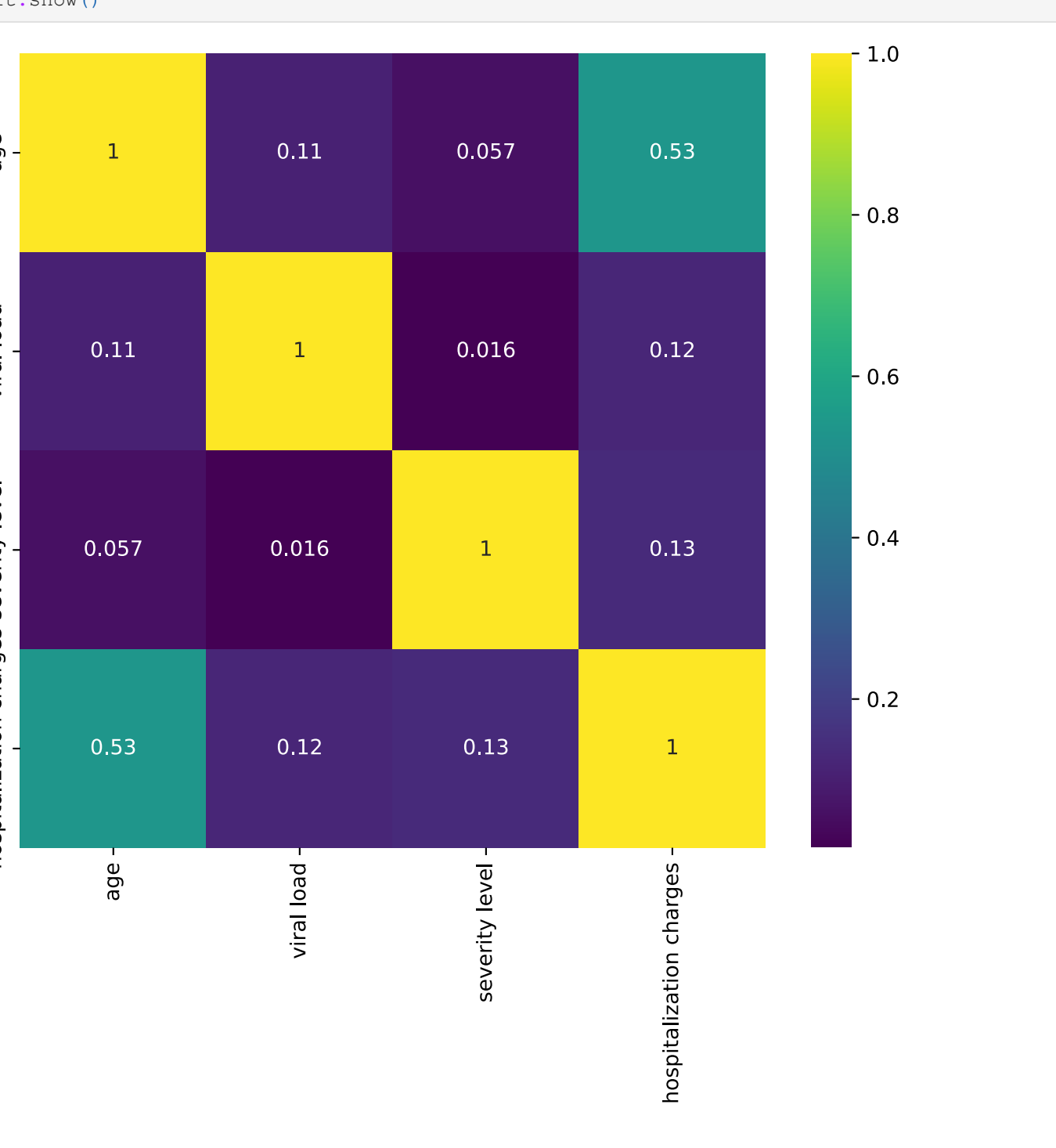


1. It is shown that most of the people are non smoker.

In [ ]:

Bivariate Analysis

In [16]:
```
## Heatmap of corelation
plt.figure(figsize=(8, 7))
sns.heatmap(df.corr(method='spearman'),
        annot=True, cmap='viridis')
plt.show()
```



We can see that there is a strong positive corelation between age and hospitalization charges, and some relation between severity level and hospitalization charges and viral load and hospitalization charges

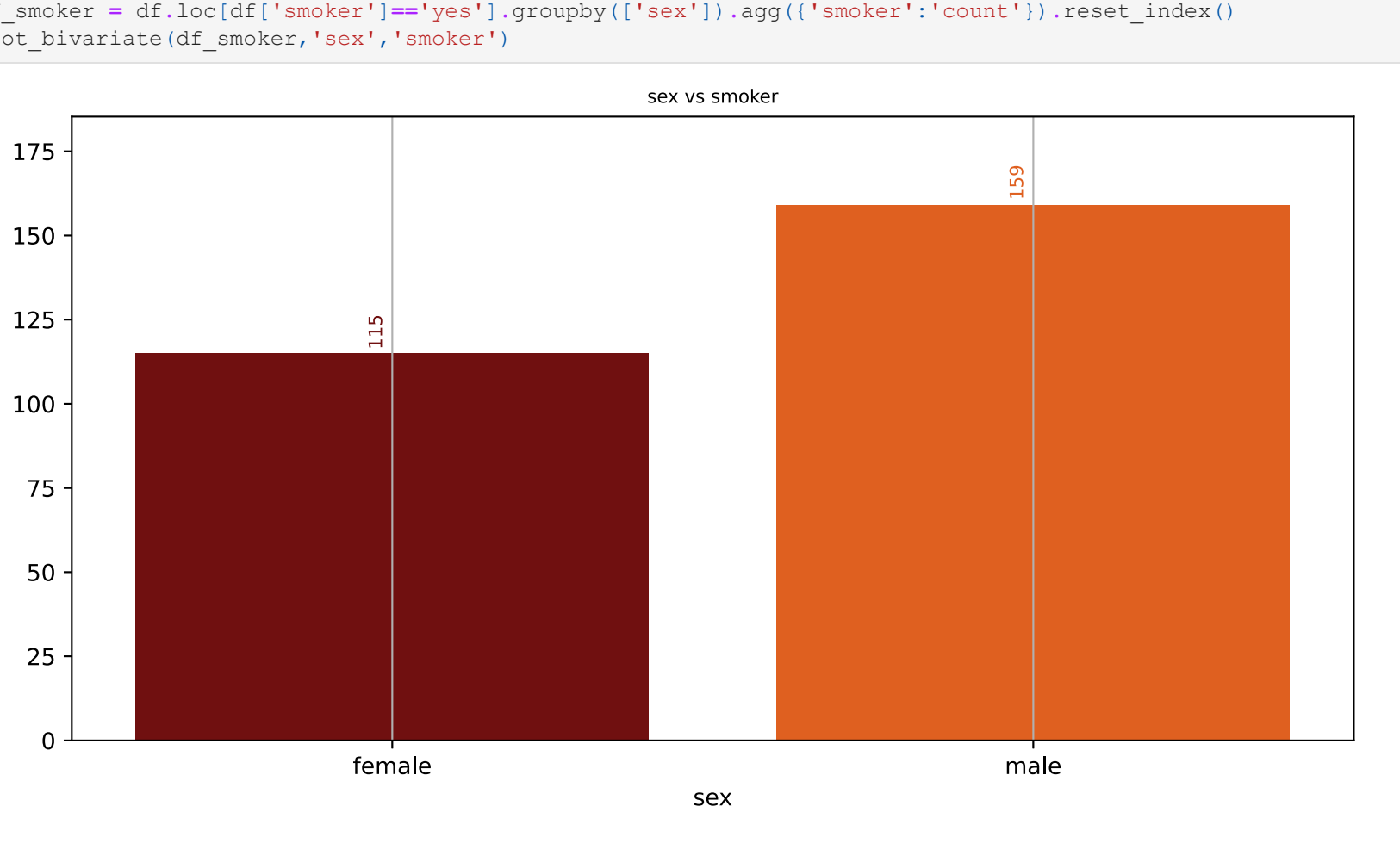
In [ ]:

In [54]:
```
## generic function to plot bivariate plots

def plot_bivariate(df,col1,col2,scatter=False):

    if not scatter:
        i=0
        plt.figure(figsize= (10,5))
        ax=sns.barplot(data= df,x=col1,y=col2, palette= plot_cmap)

        #annotate values
        for p in ax.patches:
            #if value of bar is less than 10% of max value, change the alignment
            ax.annotate(f'\n{np.round(p.get_height(),2)}', (p.get_x()+p.get_width()*0.45, p.get_height()*1.01),
                        ha= 'center', va= 'bottom', color= p.get_facecolor(), rotation= 90, size= 8)

        #increase ylim to accomodate annotations
        ax.set_ylim([0, ax.get_ylim()[1]*1.11])


        #add title and x-axis grid
        title = f"{col1} vs {col2}"
        plt.title(title, fontsize=8)
        ax.xaxis.grid()
        plt.show()

    else:
        ax = sns.scatterplot(data=df,x=col1,y=col2,hue=col2,palette= plot_cmap)
        title = f"{col1} vs {col2}"
        plt.title(title, fontsize=8)
        ax.xaxis.grid()
        plt.show()
```

In [ ]:

1. Smoker VS Gender

In [18]:
```
## plot of smoker vs gender

df_smoker = df.loc[df['smoker']=='yes'].groupby(['sex']).agg({'smoker':'count'}).reset_index()
plot_bivariate(df_smoker,'sex','smoker')
```

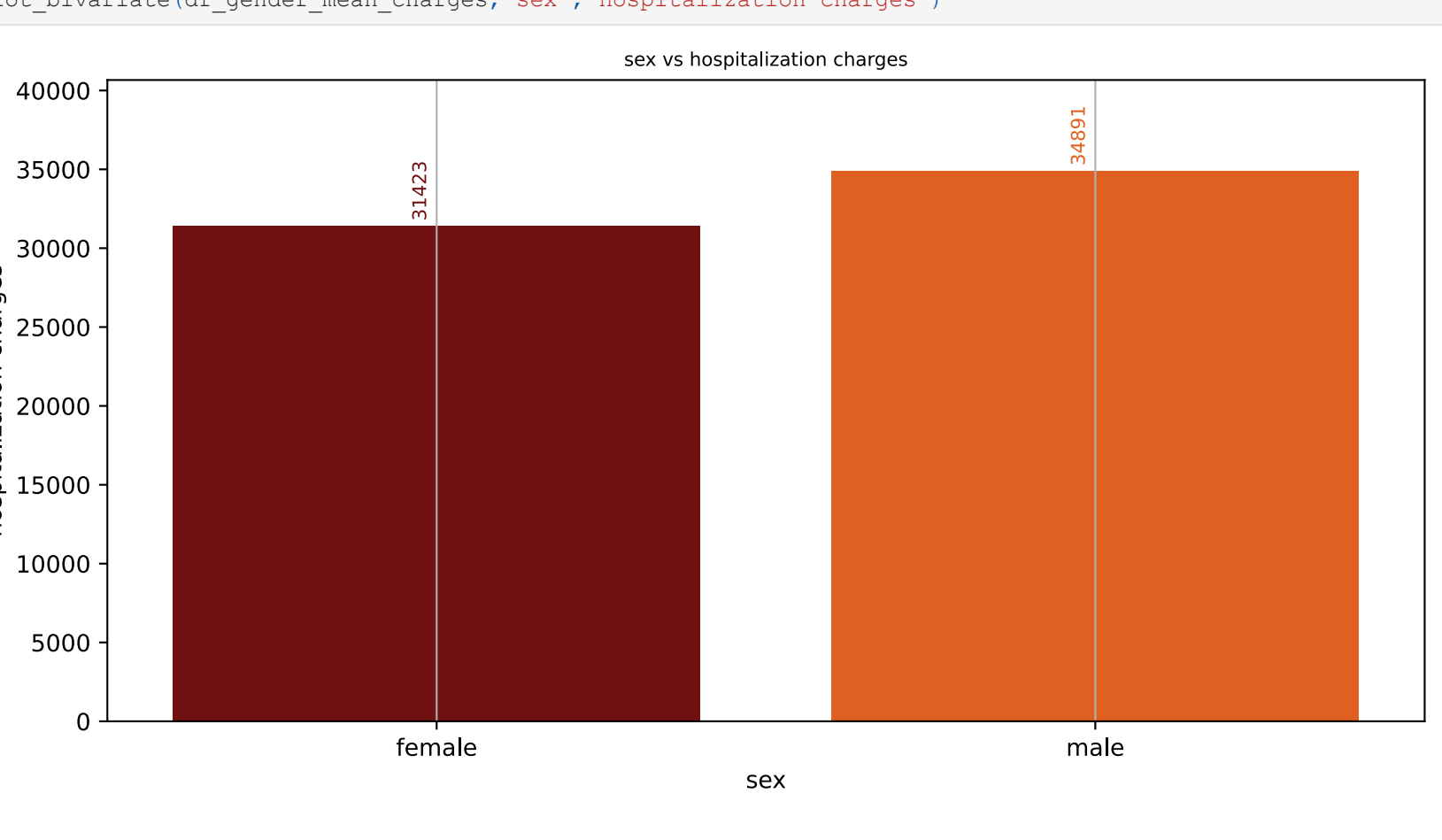


There are 115 females who smoke and 159 males who smoke

In [ ]:

2. Gender vs hospitalization charges

In [19]:
```
## plot for Gender vs hospitalization charges

df_gender_mean_charges = df.groupby(['sex']).agg({'hospitalization charges':'mean'}).reset_index()
plot_bivariate(df_gender_mean_charges,'sex','hospitalization charges')
```

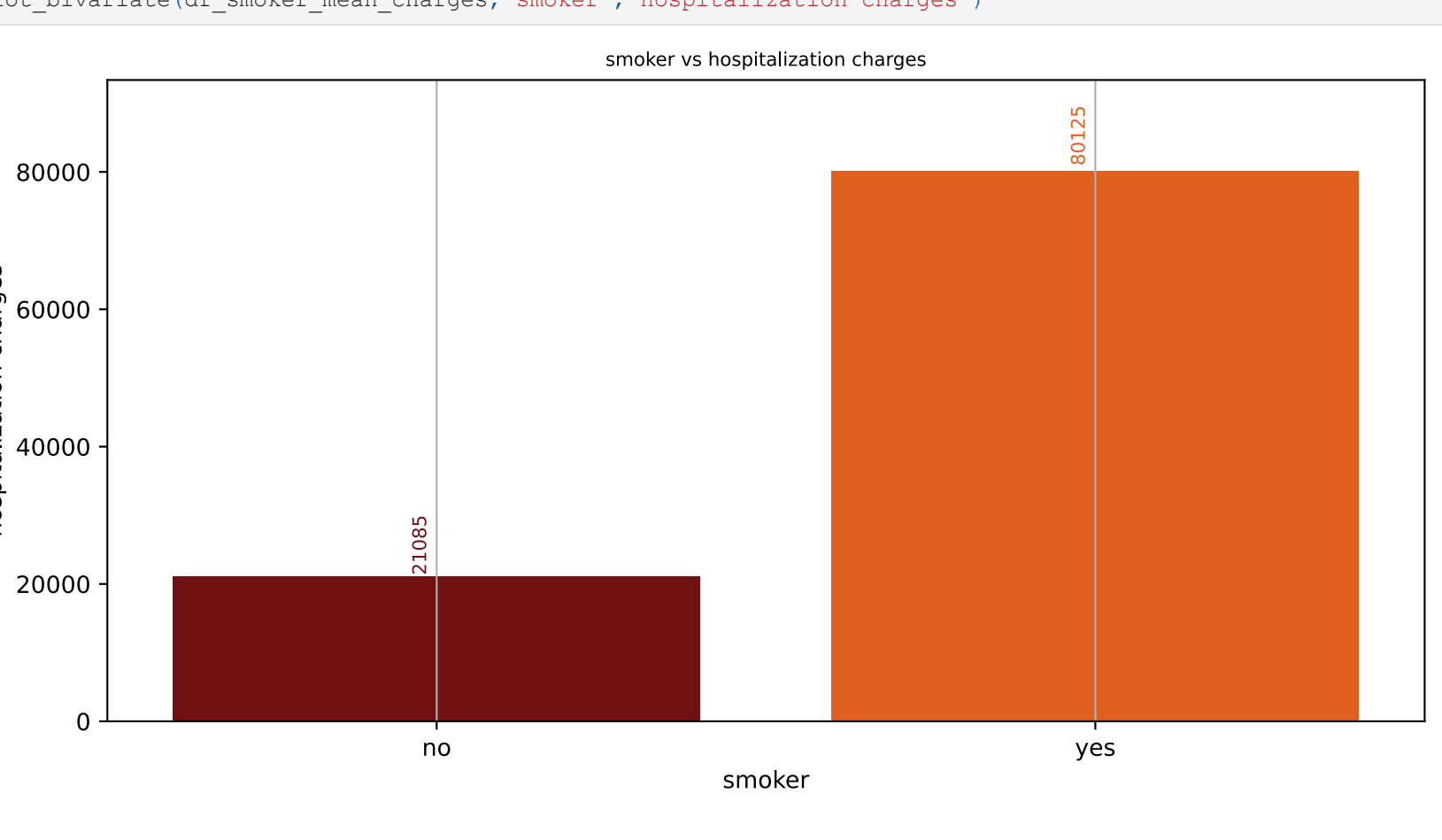


Mean value of hospitalization charges of females is 31423 and that of male is 34891. We can see that there is little difference between the means of charges for male and females.

In [ ]:

3. Smoker VS Hospitalization charges

In [20]:
```
## Smoker VS Hospitalization charges

df_smoker_mean_charges = df.groupby(['smoker']).agg({'hospitalization charges':'mean'}).reset_index()
plot_bivariate(df_smoker_mean_charges,'smoker','hospitalization charges')
```

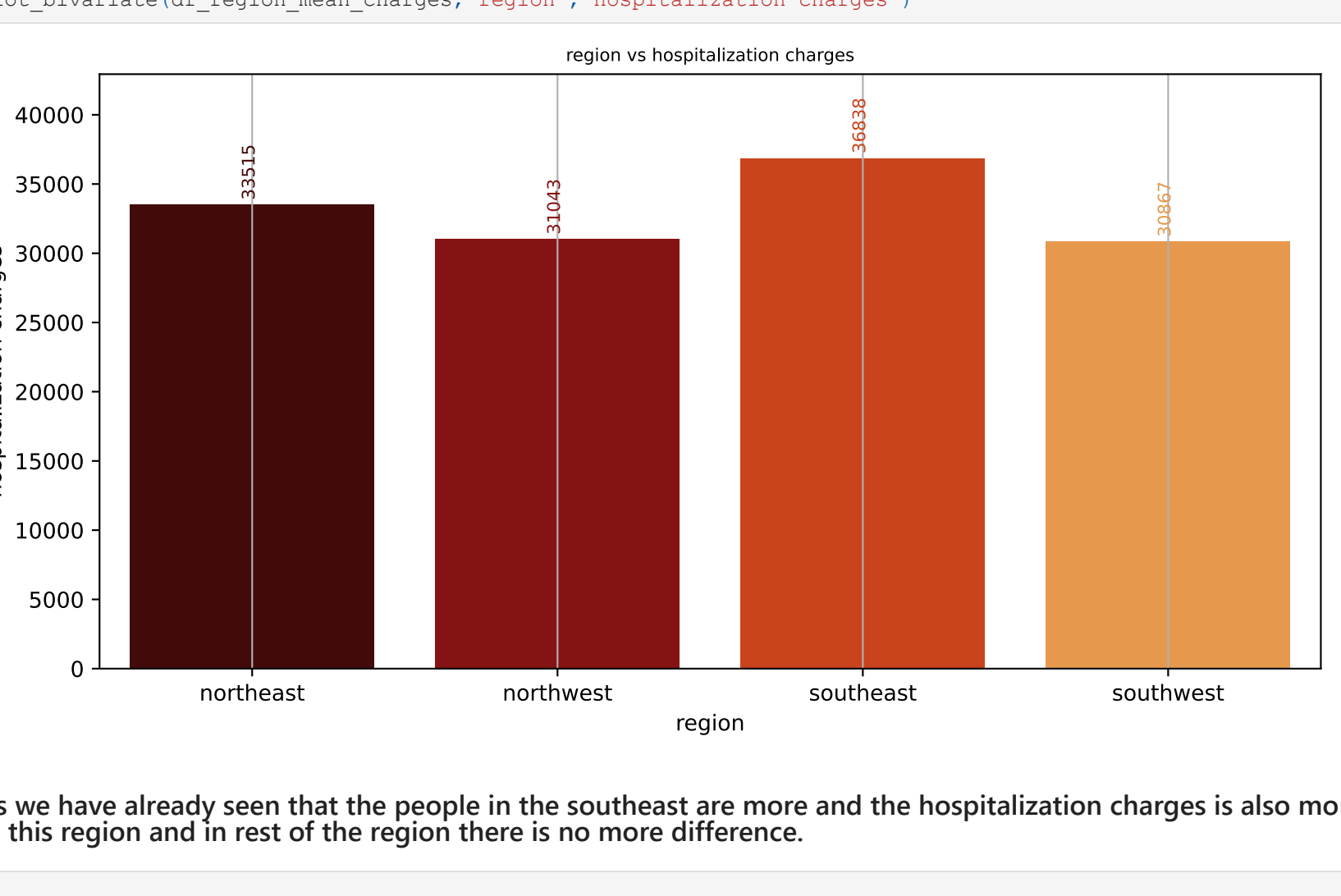


We can see that the mean charges of a smoker has more then the non smoker, because of the region that the people who smoke are more infected to the virus due to which the cost is high.

In [ ]:

4. Region VS Hospitalization charges

In [21]:
```
## Region VS Hospitalization charges

df_region_mean_charges = df.groupby(['region']).agg({'hospitalization charges':'mean'}).reset_index()
plot_bivariate(df_region_mean_charges,'region','hospitalization charges')
```

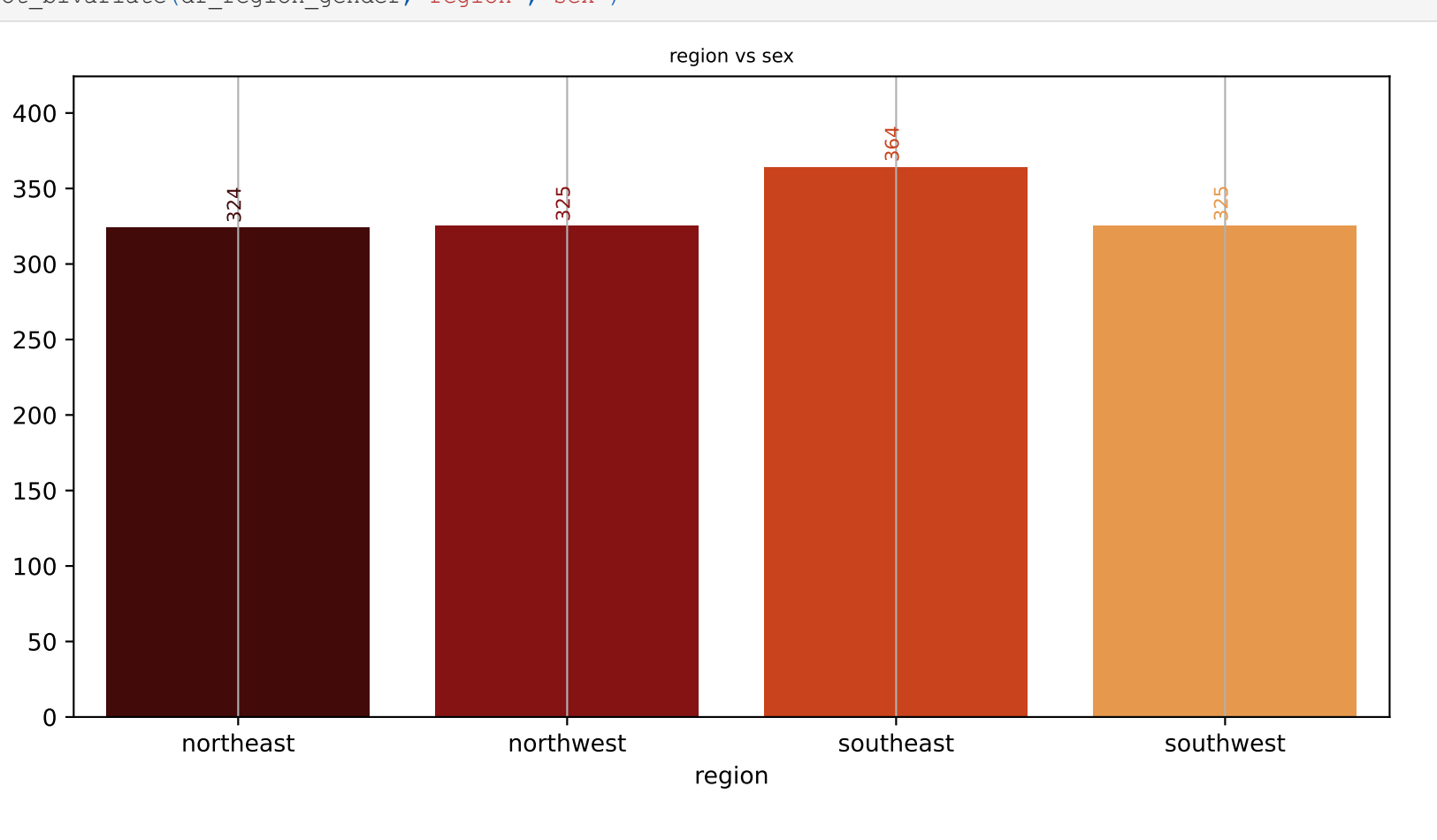


As we have already seen that the people in the southeast are more and the hospitalization charges is also more in this region and in rest of the region there is no more difference.

In [ ]:

5. Region VS Gender

In [22]:
```
## Region VS Gender

df_region_gender = df.groupby(['region']).agg({'sex':'count'}).reset_index()
plot_bivariate(df_region_gender,'region','sex')
```

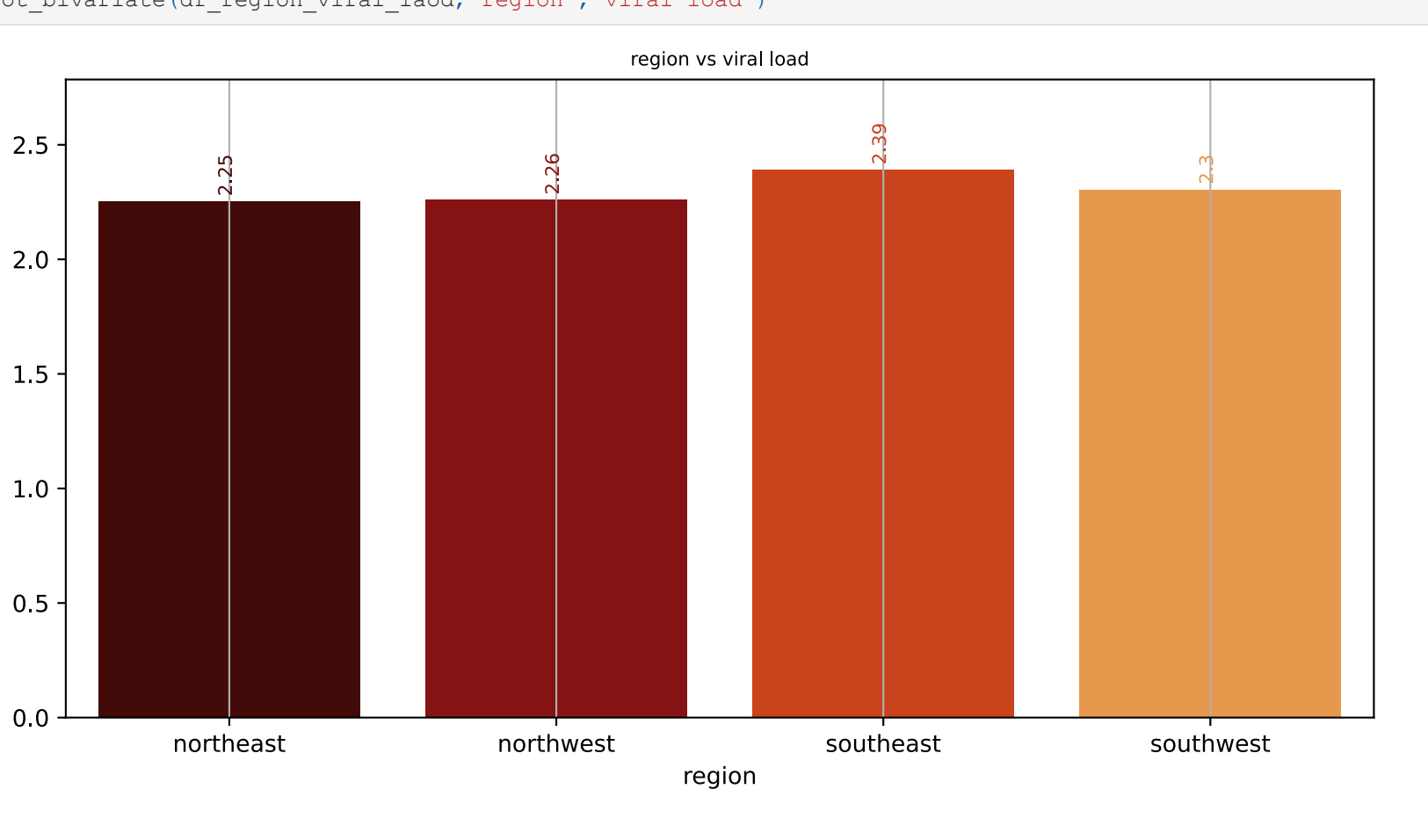


In southeast region there are more number of people and in rest of the regions data is equally divided.

In [ ]:

6. Region VS Viral Load

In [55]:
```
## Region Vs Viral Load

df_region_viral_laod = df.groupby(['region']).agg({'viral load':'mean'}).reset_index()
plot_bivariate(df_region_viral_laod,'region','viral load')
```

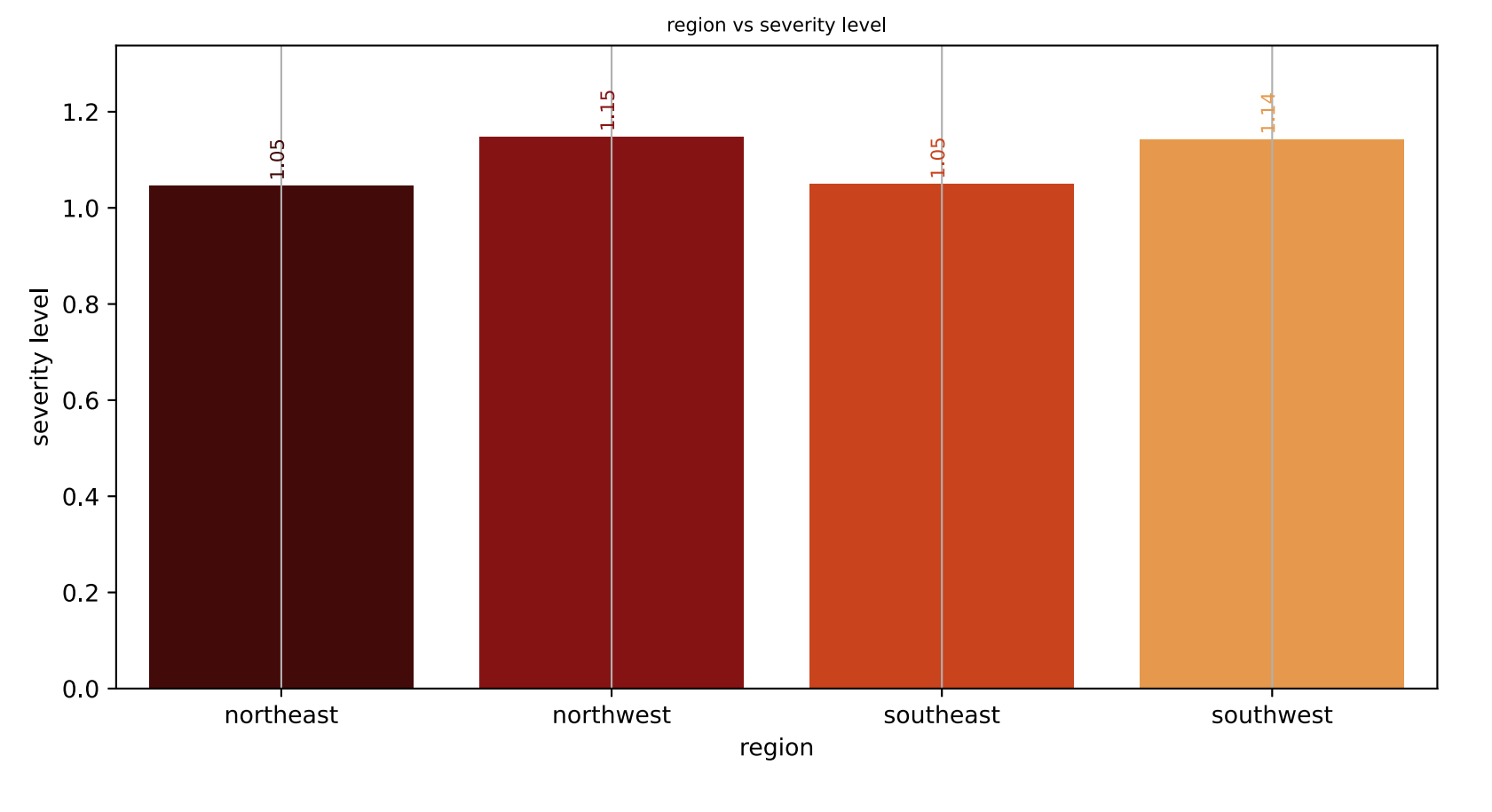


We can see that in southeast region there is more viral load then of others due to which people get hospitalized.

In [ ]:

7. Region VS Severity Level

In [57]:
```
## Region VS Severity Level

df_region_severity_level = df.groupby(['region']).agg({'severity level':'mean'}).reset_index()
plot_bivariate(df_region_severity_level,'region','severity level')
```



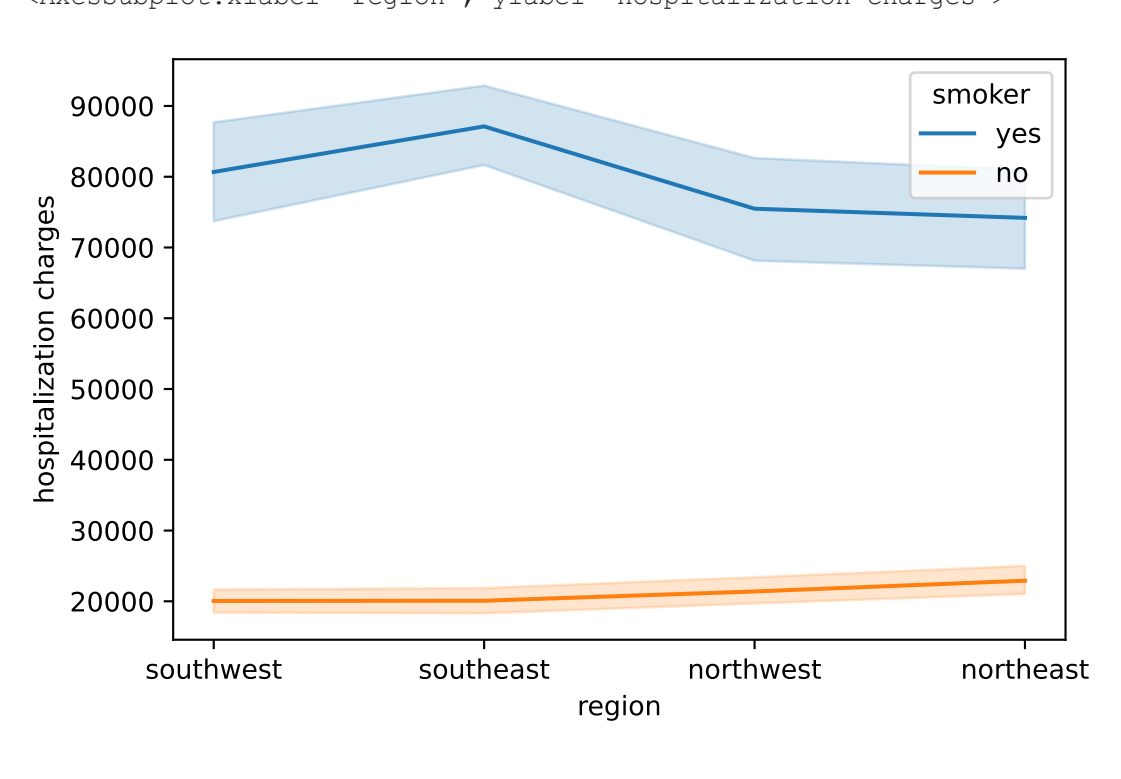
We can see that most of the people who hospitalized in northwest region has high severity level then rest of the regions.

In [ ]:

8. Region VS Hospitalization Charges for smoker and non smoker

In [67]:
```
## Region VS Hospitalization Charges for smoker and non smoker

sns.lineplot(data = df, x='region', y='hospitalization charges',hue='smoker')
```

Out[67]: <AxesSubplot:xlabel='region', ylabel='hospitalization charges'>



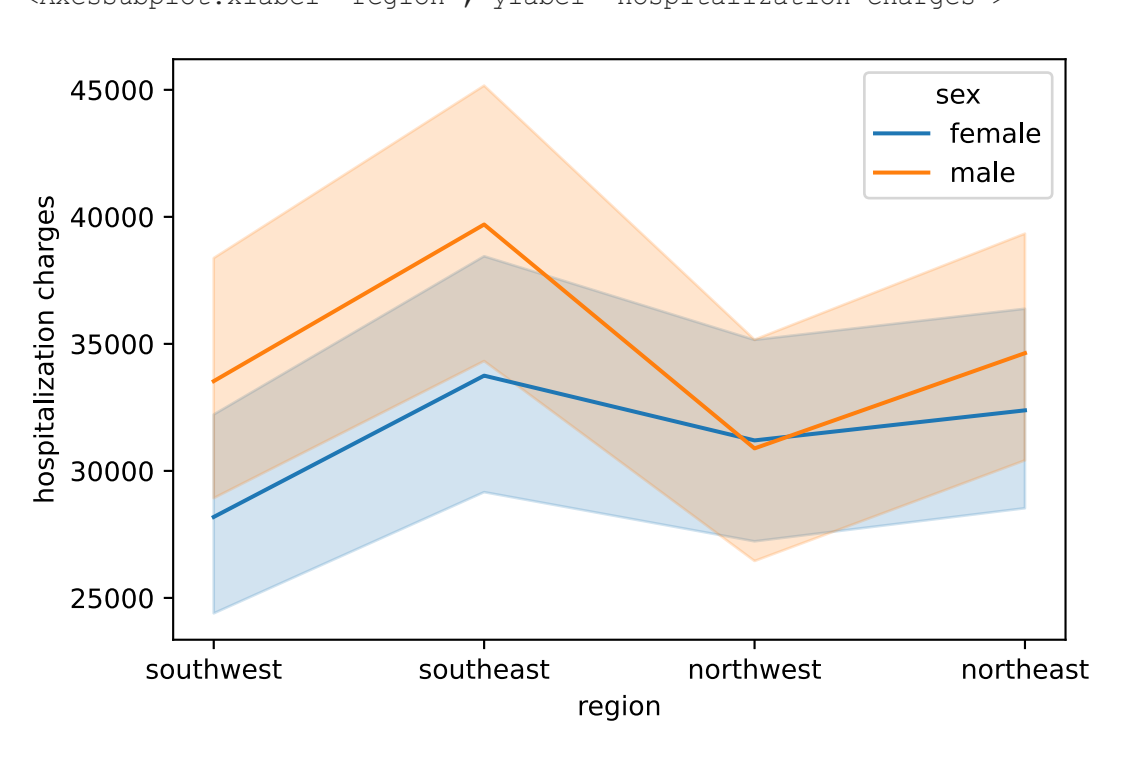
We can see that the charges for smoker is more in all the regions then the non somker but it is maximum in southeast region

In [ ]:

9. Region VS hospitalization charges for male or Female

In [71]:
```
## Region VS hospitalization charges for male or Female

sns.lineplot(data = df, x='region', y='hospitalization charges',hue='sex')
```

Out[71]: <AxesSubplot:xlabel='region', ylabel='hospitalization charges'>

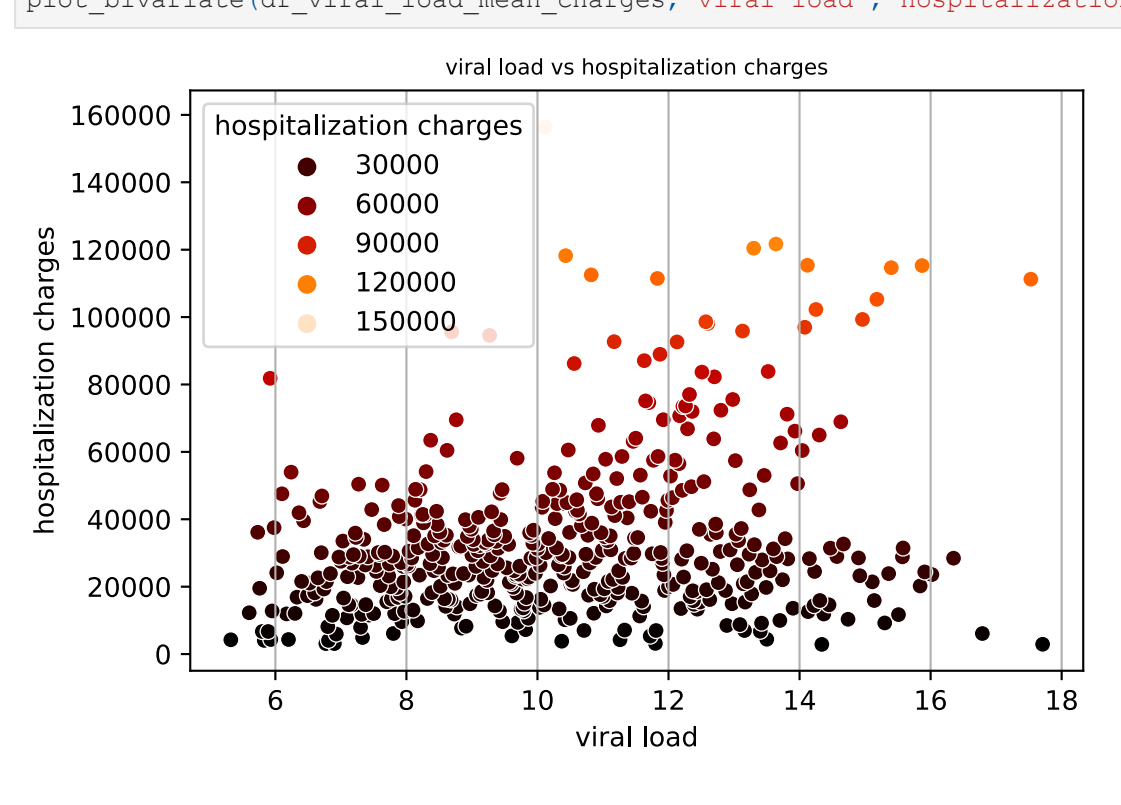


There are more charges of male then female in all the regions except nothwest, means in this region more female members get hospitalized

In [ ]:

10. Viral Load VS Hospitalization charges

In [23]:
```
#### Viral Load VS Hospitalization charges

df_viral_load_mean_charges = df.groupby(['viral load']).agg({'hospitalization charges':'mean'}).sort_values(by=

plot_bivariate(df_viral_load_mean_charges,'viral load','hospitalization charges',scatter=True)
```

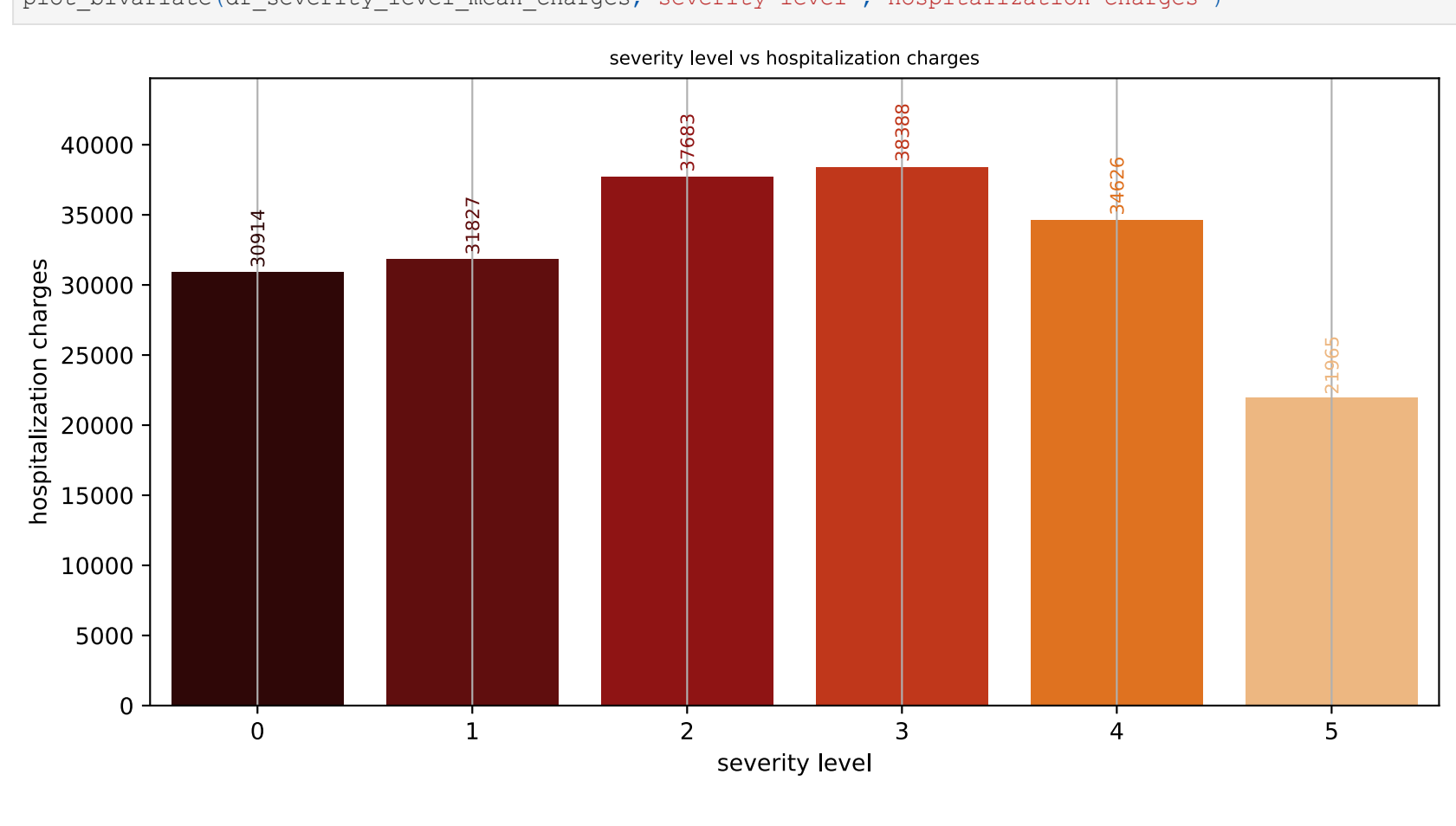


We can see that as viral load increases there are more numbers of people with high hospitalization charges.

In [ ]:

11. Severity level VS Hospitalization Charges

In [24]:
```
## Severity level VS Hospitalization Charges

df_severity_level_mean_charges = df.groupby(['severity level']).agg({'hospitalization charges':'mean'}).sort_va
plot_bivariate(df_severity_level_mean_charges,'severity level','hospitalization charges')
```

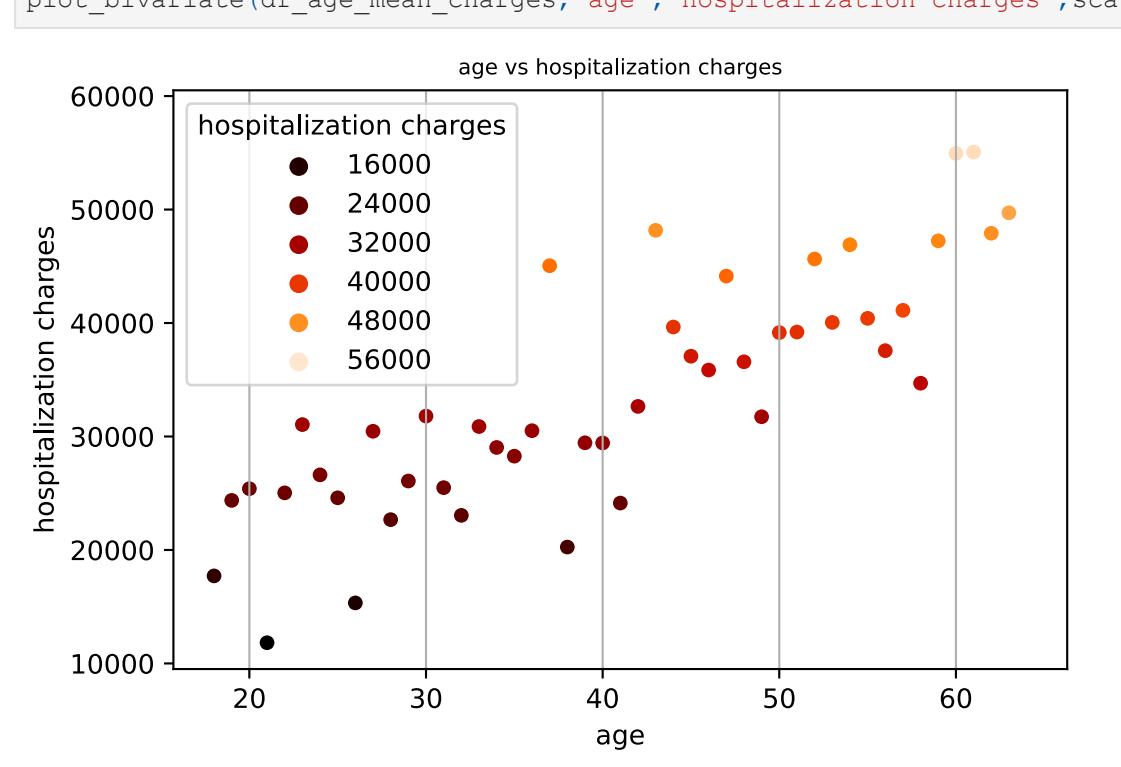


Hospitalization charges for individual is more in level 2 and 3 and least in level 5

In [ ]:

12. Age VS Hospitalization Charges

In [25]:
```
## Age VS Hospitalization Charges

df_age_mean_charges = df.groupby(['age']).agg({'hospitalization charges':'mean'}).sort_values(by=['hospitalizat
plot_bivariate(df_age_mean_charges,'age','hospitalization charges',scatter=True)
```



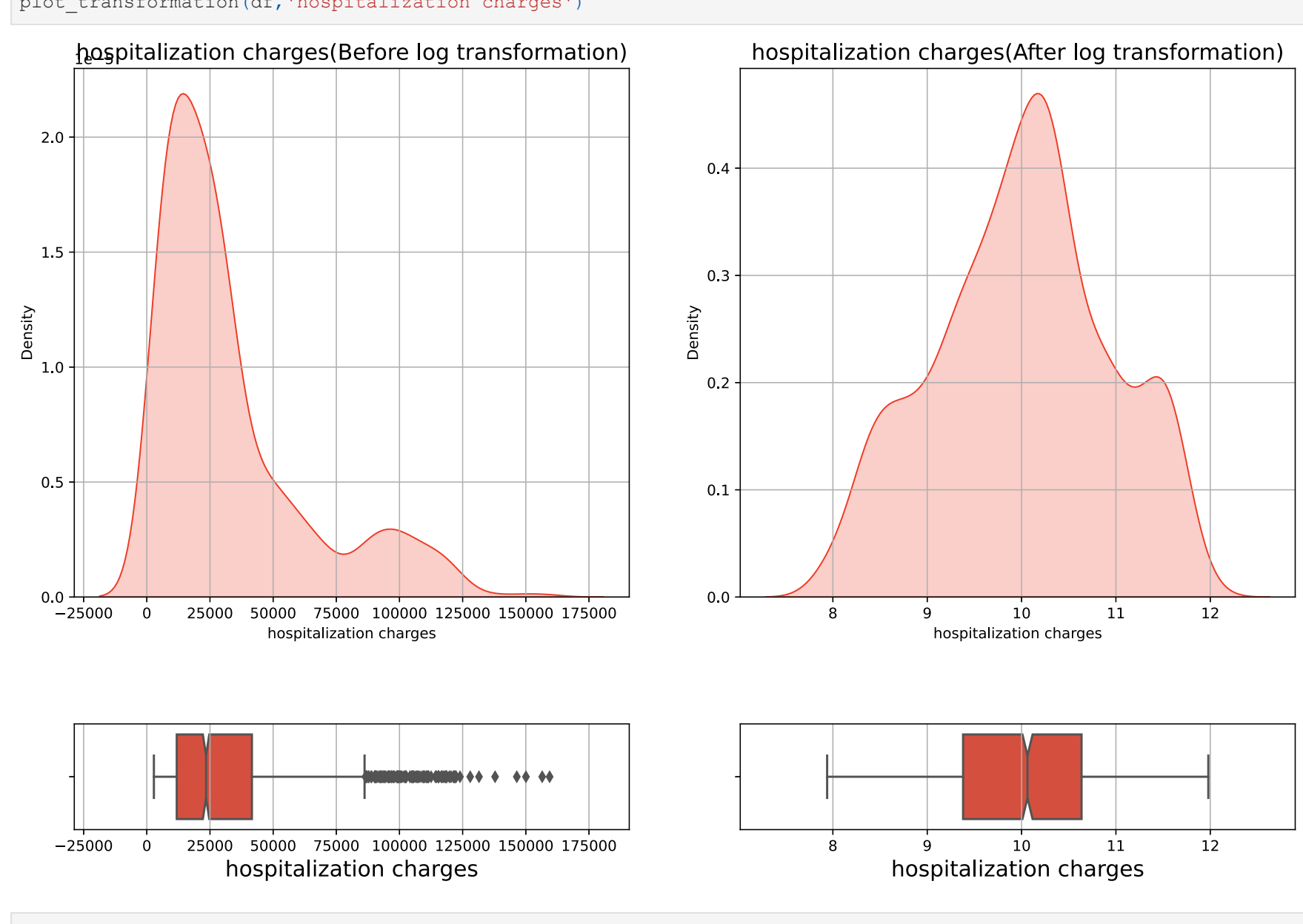
We can see a liner increasing trend in age and hospitalization charges

Handling Outliers

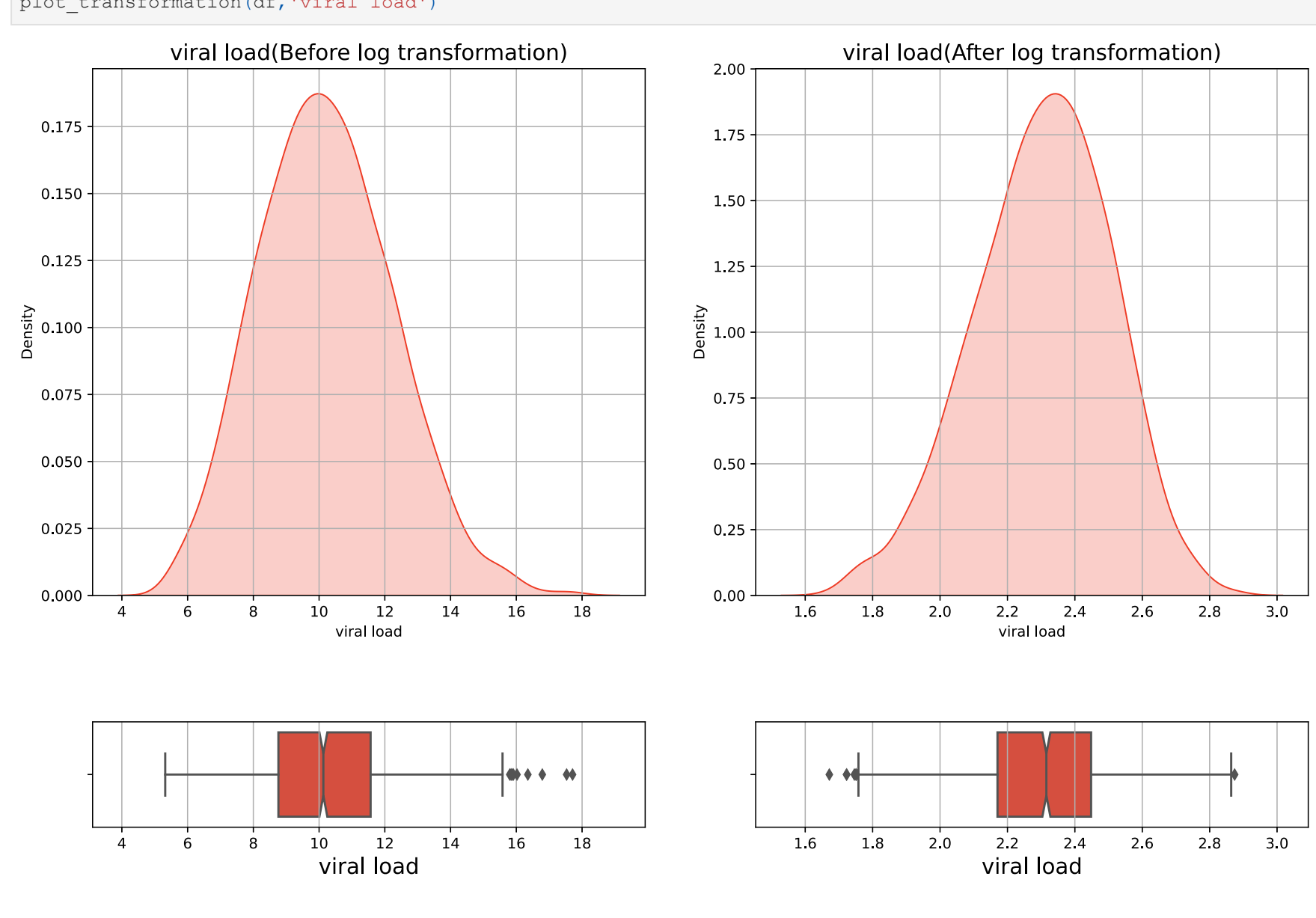
In [73]:
```
## Generic function to plot graph before and after handling outliers

def plot_transformation(df,column):
    fig, axes = plt.subplots(nrows=2, ncols=2, figsize=(15, 10), gridspec_kw={'height_ratios': [5, 1]})

    fig.subplots_adjust(hspace=0.4, top=0.85)

    sns.kdeplot(data=df, x=column, ax=axes[0,0],color=plot_color,shade=True)

    #plot box plot
    sns.boxplot(x=column, data=df, ax=axes[1,0],color=plot_color,notch= True)

    #set edge and face colours
    for i,box in enumerate(axes[1,0].artists):
        fc = box.get_facecolor()
        box.set_edgecolor(plot_color)
        box.set_facecolor(mpl.colors.to_rgba(fc, 0.3))

        for j in range(6*i,6*(i+1)):
            axes[1,0].lines[j].set_color(plot_color)

    #set xlables of boxplot same as KDE plot
    axes[1,0].set_xlim(axes[0,0].get_xlim())
    axes[1,0].set_xlabel(column,fontsize = 15)
    axes[1,0].xaxis.grid(True)

    df[column] = np.log(df[column])

    sns.kdeplot(data=df, x=column, ax=axes[0,1],color=plot_color,shade=True)

    axes[0,0].grid()
    axes[0,1].grid()

    axes[0,0].set_title(f"{column}(Before log transformation)",fontsize = 15)
    axes[0,1].set_title(f"{column}(After log transformation)",fontsize = 15)

    #plot box plot
    sns.boxplot(x=column, data=df, ax=axes[1,1],color=plot_color,notch= True)

    #set edge and face colours
    for i,box in enumerate(axes[1,1].artists):
        fc = box.get_facecolor()
        box.set_edgecolor(plot_color)
        box.set_facecolor(mpl.colors.to_rgba(fc, 0.3))

        for j in range(6*i,6*(i+1)):
            axes[1,1].lines[j].set_color(plot_color)

    #set xlables of boxplot same as KDE plot
    axes[1,1].set_xlim(axes[0,1].get_xlim())
    axes[1,1].set_xlabel(column,fontsize = 15)
    axes[1,1].xaxis.grid(True)

    plt.show()
```

In [ ]:

In [28]:
```
plot_transformation(df,'hospitalization charges')
```



In [29]:
```
plot_transformation(df,'viral load')
```



We can see that there are some outliers in hospitalization charges and viral load so we handled them using log transformation, and outliers is handled now we can see in the graph that outliers are not there

In [ ]:

Hypothesis Testing

In [ ]:

In [34]:
```
## Genric function for hypothesis testing

def display_two_kde_plots(data1, data2 , col1, col2, xlabel=None):
    sns.kdeplot(data=data1, x=col1, label=col1)
    sns.kdeplot(data=data2, x=col2, label=col2)
    if xlabel:
        plt.xlabel(xlabel)
#     plt.legend()
    plt.grid()
    plt.show()

def qq_plot(df,df2,col1,col2):
    qqplot_2samples(df[col1], df2[col2], line="r")
    plt.show()

def check_for_equal_variances(sample_1, sample_2, significance_level=0.05):

    print(f"Variance of sample 1 = {round(sample_1.var(), 2)}")
    print(f"Variance of sample 2 = {round(sample_2.var(), 2)}")
    p_val = levene(sample_1, sample_2)[1]
    print(f"Levene test of equal variance's p-value = {p_val}")
    if p_val > significance_level:
        print(f"Since p-value is greater than significance level ({significance_level}) ,we fail to reject the
    else:
        print(f"Since p-value is lesser than significance level ({significance_level}) ,we can reject null hypo

def perform_independent_ttest(sample_1, sample_2, alternate_hyp='two-sided', equal_var=True, significance_level

    p_val = ttest_ind(sample_1, sample_2, alternative=alternate_hyp)[1]
    print(f"Two sampled indepedent t-test's p-value = {p_val}")
    if p_val > significance_level:
        print(f"Since p-value is greater than significance level ({significance_level}) ,we fail to reject the
    else:
        print(f"Since p-value is lesser than significance level ({significance_level}) ,we can reject null hypo


def perform_two_sampled_ks_test(sample1, sample2, significance_level=0.05):

    p_val = ks_2samp(sample1, sample2)[1]
    print(f"KS test's p-value = {p_val}")
    if p_val > significance_level:
        print(f"Since p-value is greater than significance level ({significance_level}) ,we fail to reject the
    else:
        print(f"Since  p-value is not greater than significance level ({significance_level}) ,we can reject nul


def chi_square_test(sample1, sample2, significance_level=0.05):
    table_1 = pd.crosstab(sample1, sample2)
    print("Observed values:")
    print(table_1)
    p_val = stats.chi2_contingency(table_1)[1]
    print(f"Chi square test's p-value = {p_val}")
    if p_val > significance_level:
        print(f"Since p-value is greater than significance level ({significance_level}) ,we fail to reject the
    else:
        print(f"Since  p-value is not greater than significance level ({significance_level}) ,we can reject nul
```

In [ ]:

1. Hospitalization of people who do smoking is greater than those who don't (T-test Right tailed)

Null Hypothesis(H0) : Mean of Hospitalization charge of people who do smoking is equal to those who don't.

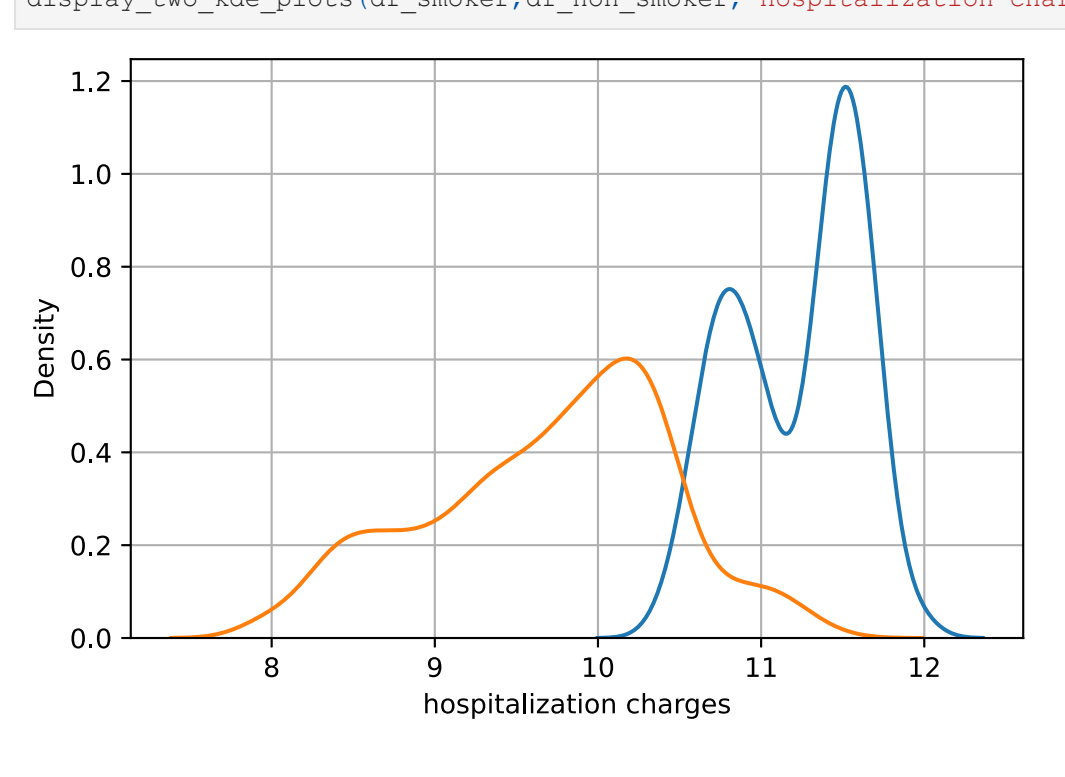
Alternate Hypothesis(H1) : Mean of Hospitalization charge of people who do smoking is greater then those who don't

Assumptions of t-test

1. Data values are continuous
2. Observations are taken through independent and random sampling
3. Data in each group is normally distributed
4. Variances of both groups is the same (1) is true, we assume (2) to be true, (3) can anyways be bypassed because CLT anyways gurantees sample means follow the normal distribution when sample size > 30 althought we will check it using QQ plot but we have to check for (4)

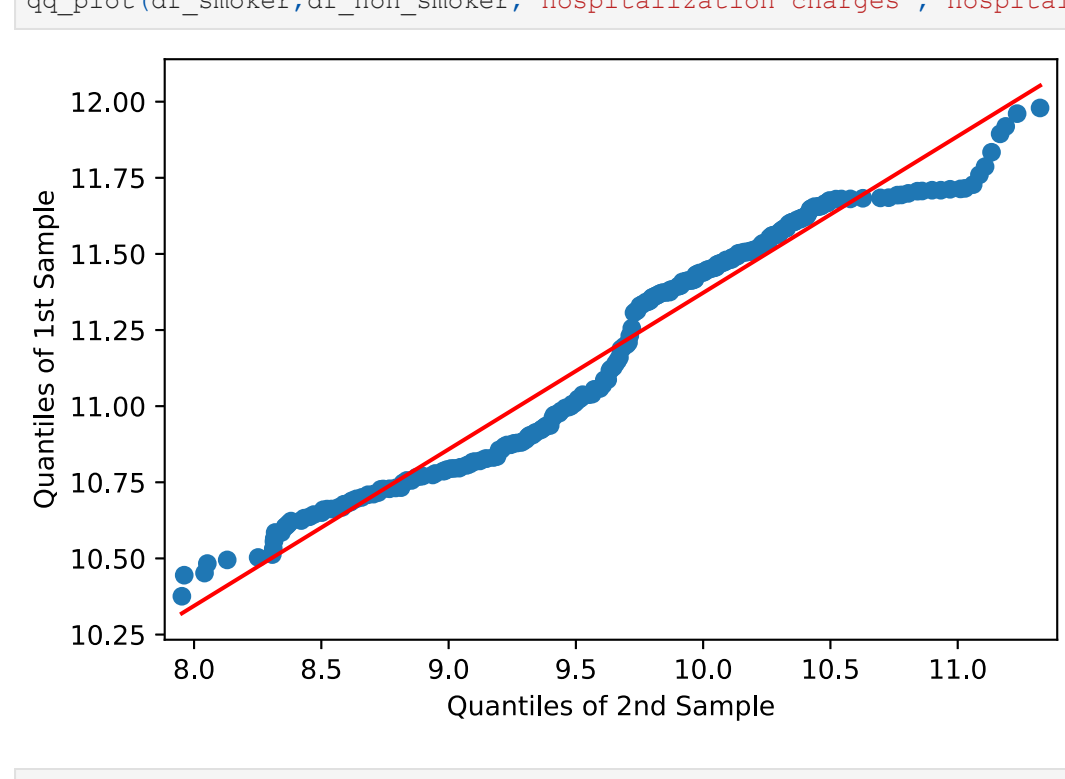
In [35]:
```
df_smoker = df.loc[df['smoker']=='yes']
df_non_smoker = df.loc[df['smoker']=='no']
```

In [36]:
```
display_two_kde_plots(df_smoker,df_non_smoker,'hospitalization charges','hospitalization charges')
```



QQ plot

In [37]:
```
qq_plot(df_smoker,df_non_smoker,'hospitalization charges','hospitalization charges')
```



In [74]:
```
check_for_equal_variances(df_smoker['hospitalization charges'],df_non_smoker['hospitalization charges'])
```

```
Variance of sample 1 = 0.15
Variance of sample 2 = 0.55
Levene test of equal variance's p-value = 1.332832802389931e-20
Since p-value is lesser than significance level (0.05) ,we can reject null hypothesis and say that the samples
have different variance at 95% confidence
```

In [77]:
```
perform_independent_ttest(df_smoker['hospitalization charges'],df_non_smoker['hospitalization charges'],alterna
```

```
Two sampled indepedent t-test's p-value = 3.152579721099768e-172
Since p-value is lesser than significance level (0.05) ,we can reject null hypothesis and say that the samples
have different means at 95% confidence
```

In [ ]:

We can also verify this assumption with KS Test (where null hypothesis is that CDFs of the two distributions exactly overlap) which would tell us whether the 2 distributions are identical or not

In [40]:
```
perform_two_sampled_ks_test(df_smoker['hospitalization charges'],df_non_smoker['hospitalization charges'])
```

```
KS test's p-value = 4.557845534711012e-213
Since  p-value is not greater than significance level (0.05) ,we can reject null hypothesis and say that the sa
mples have different distributions at 95% confidence
```

From visual analysis and hypothesis testing (t-test and KS-test), we can accept the alternate hypothesis and can say that the Mean of Hospitalization charge of people who do smoking is greater then those who don't

In [ ]:

2. Viral load of females is different from that of males (Two tailed T test)

Null Hypothesis(H0) : Mean of Viral load of females is same as that of males

Alternate Hypothesis(H1) : Mean of Viral load of females is different from that of males

Assumptions of t-test

1. Data values are continuous
2. Observations are taken through independent and random sampling
3. Data in each group is normally distributed
4. Variances of both groups is the same (1) is true, we assume (2) to be true, (3) can anyways be bypassed because CLT anyways gurantees sample means follow the normal distribution when sample size > 30 althought we will check it using QQ plot but we have to check for (4)

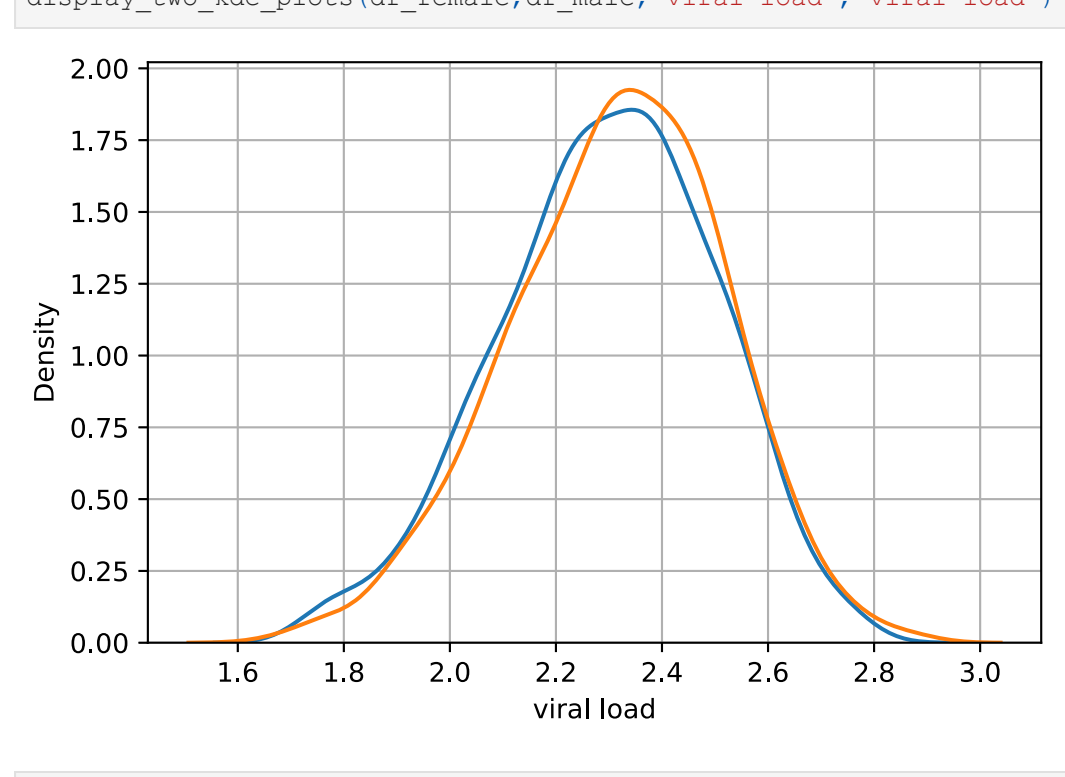
In [41]:
```
df.head()
```

Out[41]:

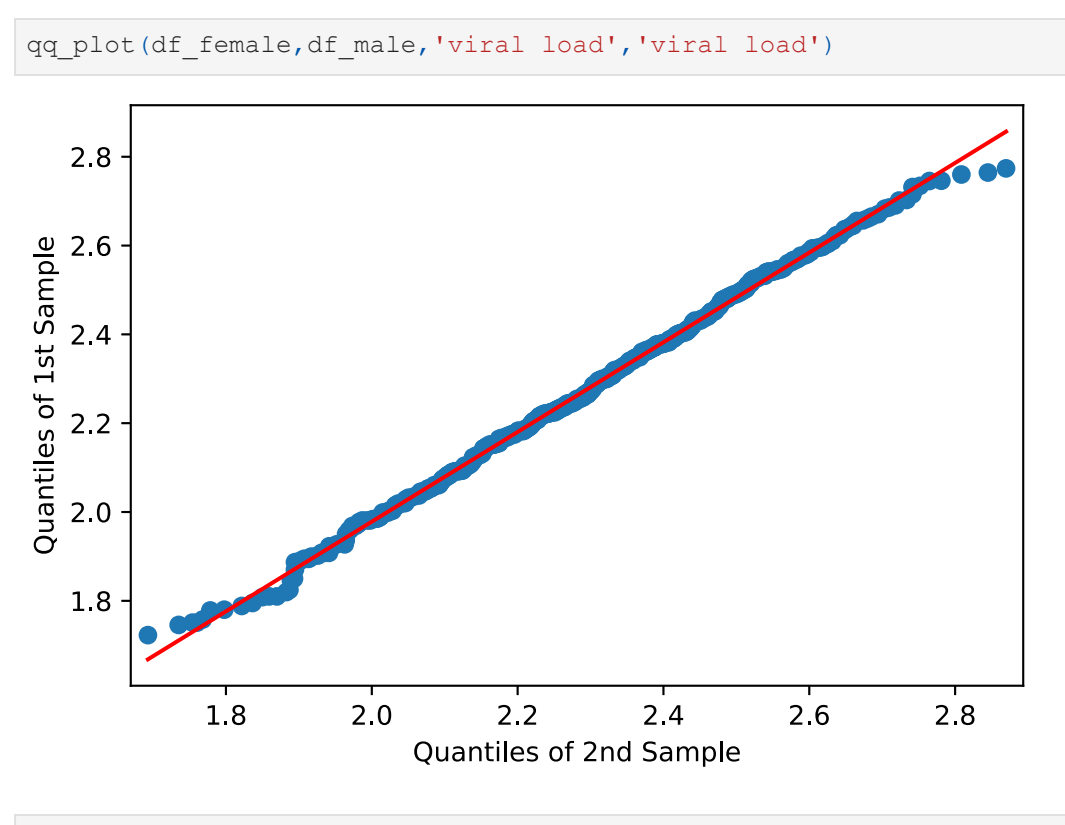
	age	sex	smoker	region	viral load	severity level	hospitalization charges
0	19	female	yes	southwest	2.230014	0	10.650460
1	18	male	no	southeast	2.421257	1	8.369621
2	28	male	no	southeast	2.397895	3	9.316860
3	33	male	no	northwest	2.024193	0	10.914379
4	32	male	no	northwest	2.264883	0	9.176473

In [42]:
```
df_female = df.loc[df['sex']=='female']
df_male = df.loc[df['sex']=='male']
```

In [43]:
```
display_two_kde_plots(df_female,df_male,'viral load','viral load')
```



In [44]:
```
qq_plot(df_female,df_male,'viral load','viral load')
```



In [45]:
```
check_for_equal_variances(df_female['viral load'],df_male['viral load'])
```

```
Variance of sample 1 = 0.04
Variance of sample 2 = 0.04
Levene test of equal variance's p-value = 0.6954736404070048
Since p-value is greater than significance level (0.05) ,we fail to reject the null hypothesis and hence have t
o say that the samples have equal variance at 95% confidence
```

In [46]:
```
perform_independent_ttest(df_female['viral load'],df_male['viral load'], equal_var=False)
```

```
Two sampled indepedent t-test's p-value = 0.09227949561258407
Since p-value is greater than significance level (0.05) ,we fail to reject the null hypothesis and hence have t
o say that the two samples have same mean at 95% confidence
```

In [ ]:

We can also verify this assumption with KS Test (where null hypothesis is that CDFs of the two distributions exactly overlap) which would tell us whether the 2 distributions are identical or not

In [47]:
```
perform_two_sampled_ks_test(df_female['viral load'],df_male['viral load'])
```

```
KS test's p-value = 0.1803102096437813
Since p-value is greater than significance level (0.05) ,we fail to reject the null hypothesis and hence have t
o say that the two distributions are the same at 95% confidence
```

In [ ]:

From visual analysis and hypothesis testing (t-test and KS-test), we fail to reject the null hypothesis and can say that Viral load of females is same that of males

In [ ]:

3. Proportion of smoking significantly different across different regions. (Chi Square Test)

Null Hypothesis(H0) : Proportion of smoking is same across different regions

Alternate Hypothesis(H1) : Proportion of smoking significantly different across different regions

Assumptions of chi square test

1. Both variables are categorical
2. All observations are independent.
3. Cells in the contingency table are mutually exclusive.
4. Expected value of cells should be 5 or greater in at least 80% of cells.

All of them are true so we are good to perform this test

In [48]:
```
chi_square_test(df['smoker'], df['region'])
```

```
Observed values:
region  northeast  northwest  southeast  southwest
smoker
no            257        267        273        267
yes            67         58         91         58
Chi square test's p-value = 0.06171954839170547
Since p-value is greater than significance level (0.05) ,we fail to reject the null hypothesis and hence have t
o say that the two distributions have same proportion at 95% confidence
```

In [ ]:

Using this test we can say that we fail to reject the null hypothesis and can say that Proportion of smoking is same across different regions

In [ ]:

4. Is the mean viral load of women with 0 Severity level , 1 Severity level, and 2 Severity level the same (One Way Annova)

Null Hypothesis(H0) : mean viral load of women with 0 Severity level , 1 Severity level, and 2 Severity level the same

Alternate Hypothesis(H1) : mean viral load of women with 0 Severity level , 1 Severity level, and 2 Severity level the not same

Assumptions of one way annova

1. Each sample is taken from a normally distributed population
2. Each sample has been drawn independently of the other samples
3. Variance of data in the different groups should be the same

All of them are true so we are good to perform this test

In [49]:
```
df_female = df.loc[df['sex']=='female']
gp1 = df_female.loc[df_female['severity level']==0]['viral load'].values
gp2 = df_female.loc[df_female['severity level']==1]['viral load'].values
gp3 = df_female.loc[df_female['severity level']==2]['viral load'].values

p_val = stats.f_oneway(gp1, gp2, gp3)[1]

significance_level = 0.05
print(f"Two sampled one way annova p-value = {p_val}")
if p_val > significance_level:
    print(f"Since p-value is greater than significance level ({significance_level}) ,we fail to reject the null
else:
    print(f"Since  p-value is not greater than significance level ({significance_level}) ,we can reject null hy
```

```
Two sampled one way annova p-value = 0.832166188721299
Since p-value is greater than significance level (0.05) ,we fail to reject the null hypothesis and hence have t
o say that the mean viral load of women with 0 Severity level , 1 Severity level, and 2 Severity level the same
95% confidence
```

In [ ]:

We can say that we fail to reject the null hypothesis , i.e. mean viral load of women with 0 Severity level , 1 Severity level, and 2 Severity level the same

In [ ]:

Useful Insights from Data

1. There are 1338 data points and 7 features and there are no null values and there are both categorical and numerical values in the data.

2. There are no null values in the data.

3. Mean value of age is 39.21 and data is normally distributed.

4. Mean value of hospitalization charges is 33176 and it contains outliers which are handled by using log transformation.

5. Mean value of viral load is 10.22 and data is almost normally distributed.

6. There are 676 Male and 662 Female data points which are divided in the 4 regions in which maximum people are there in southeast region.

7. According to the data maximum number of people who are hospitalized are not very severe their severity level is between 0-2, a smaller number of people have severity more than 2.

8. According to the data most of the people are non-smoker, 115 females who smoke and 159 males who smoke.

9. There is a strong positive corelation between age and hospitalization charges, and some relation between severity level and hospitalization charges and viral load and hospitalization charges.

10. Mean value of hospitalization charges of females is 31423 and that of male is 34891. We can see that there is little difference between the means of charges for male and females.

11. For smoker people hospitalization charges are more than the non-smoker.

12. As we have already seen that the people in the southeast are more, and the hospitalization charges is also more in this region and in rest of the region there is no more difference.

13. In southeast region there are a greater number of people and in rest of the regions data is equally divided.

14. We can see that in southeast region there is more viral load then of others due to which people get hospitalized.

15. We can see that most of the people who hospitalized in northwest region has high severity level then rest of the regions.

16. We can see that the charge for smoker is more in all the regions then the non-smoker, but it is maximum in southeast region.

17. There are more charges of male then female in all the regions except northwest, means in this region more female members get hospitalized.

18. We can see that as viral load increases there are more numbers of people with high hospitalization charges.

19. Hospitalization charges for individual is more in level 2 and 3 and least in level 5

20. We can see a liner increasing trend in age and hospitalization charges

21. From visual analysis and hypothesis testing (t-test and KS-test), we can accept the alternate hypothesis and can say that the Mean of Hospitalization charge of people who do smoking is greater than those who don't.

22. From visual analysis and hypothesis testing (t-test and KS-test), we fail to reject the null hypothesis and can say that Viral load of females is same that of males

23. Using this test, we can say that we fail to reject the null hypothesis and can say that Proportion of smoking is same across different regions

24. We can say that we fail to reject the null hypothesis, i.e. mean viral load of women with 0 Severity level, 1 Severity level, and 2 Severity level the same

In [ ]:

Business Recommendations

1. According to the data the southeast region has a greater number of people and their probability of getting hospitalized is more as there are a greater number of smokers, so hospitals should be prepared according to them.

2. In southeast region people are hospitalized more because they are smoker so hospitals should be prepared for the treatment of the diseases which occurred due to smoking,

3. More aged people have high chances to be hospitalized so hospitals should be prepared accordingly.

4. If severity level is high so there are more chances of a person to get hospitalized, so all the treatment process should be made ready.

5. Viral load is not dependent on males or females, there both can be infected with the viral load and should be treated accordingly.

6. There is equal chance of having Viral load on difference severity level for females so it should also be treated accordingly.

In [ ]: