In [1]:
```
import pandas as pd
import numpy as np
from scipy import stats
import matplotlib.pyplot as plt
import seaborn as sns
import statsmodels.api as sm
from matplotlib import pyplot
import pylab as py

import warnings
warnings.filterwarnings("ignore")
```

In [2]:
```
df = pd.read_csv('apollo_hospitals.csv')
df
```

Out[2]:

	Unnamed: 0	age	sex	smoker	region	viral load	severity level	hospitalization charges
0	0	19	female	yes	southwest	9.30	0	42212
1	1	18	male	no	southeast	11.26	1	4314
2	2	28	male	no	southeast	11.00	3	11124
3	3	33	male	no	northwest	7.57	0	54961
4	4	32	male	no	northwest	9.63	0	9667
...	...	...	...	...	...	...	...	...
1333	1333	50	male	no	northwest	10.32	3	26501
1334	1334	18	female	no	northeast	10.64	0	5515
1335	1335	18	female	no	southeast	12.28	0	4075
1336	1336	21	female	no	southwest	8.60	0	5020
1337	1337	61	female	yes	northwest	9.69	0	72853

1338 rows × 8 columns

1. Define Problem Statement and perform Exploratory Data Analysis

1.a. Definition of problem (as per given problem statement with additional views).
=> Which variables are significant in predicting the reason for hospitalization for different regions.
=> How well some variables like viral load, smoking, Severity Level describe the hospitalization charges.
=> Check if there is any relation between age and hospital charges.
1.b. Observations on shape of data, data types of all the attributes, conversion of categorical attributes to 'category' (If required) , missing value detection, statistical summary.

In [3]:
```
df.columns
```

Out[3]:
```
Index(['Unnamed: 0', 'age', 'sex', 'smoker', 'region', 'viral load',
       'severity level', 'hospitalization charges'],
      dtype='object')
```

In [4]:
```
# Let's remove 'Unnamed: 0' column as it is not useful.
df = df.drop('Unnamed: 0', axis=1)
```

In [5]:
```
df.head()
```

Out[5]:

	age	sex	smoker	region	viral load	severity level	hospitalization charges
0	19	female	yes	southwest	9.30	0	42212
1	18	male	no	southeast	11.26	1	4314
2	28	male	no	southeast	11.00	3	11124
3	33	male	no	northwest	7.57	0	54961
4	32	male	no	northwest	9.63	0	9667

In [6]:
```
df.shape
```

Out[6]:
```
(1338, 7)
```

In [7]:
```
df.columns
```

Out[7]:
```
Index(['age', 'sex', 'smoker', 'region', 'viral load', 'severity level',
       'hospitalization charges'],
      dtype='object')
```

In [8]:
```
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
```

In [9]:
```
# conversion of categorical attributes to 'category'.

df['sex'] = df.sex.astype('category')
df['smoker'] = df.smoker.astype('category')
df['region'] = df.region.astype('category')
df['severity level'] = df['severity level'].astype('category')
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   category
 2   smoker                  1338 non-null   category
 3   region                  1338 non-null   category
 4   viral load              1338 non-null   float64
 5   severity level          1338 non-null   category
 6   hospitalization charges  1338 non-null   int64
dtypes: category(4), float64(1), int64(2)
memory usage: 37.4 KB
```

In [10]:
```
numerical_cols = ['age', 'viral load', 'hospitalization charges']
categorical_cols = ['sex', 'smoker', 'region', 'severity level']
```

In [11]:
```
df[numerical_cols].describe()
```

Out[11]:

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

Inference: I do not think there will be outliers. But in 'hospitalization charges' column, there might be outliers.

In [12]:
```
df[categorical_cols].describe()
```

Out[12]:

	sex	smoker	region	severity level
count	1338	1338	1338	1338
unique	2	2	4	6
top	male	no	southeast	0
freq	676	1064	364	574

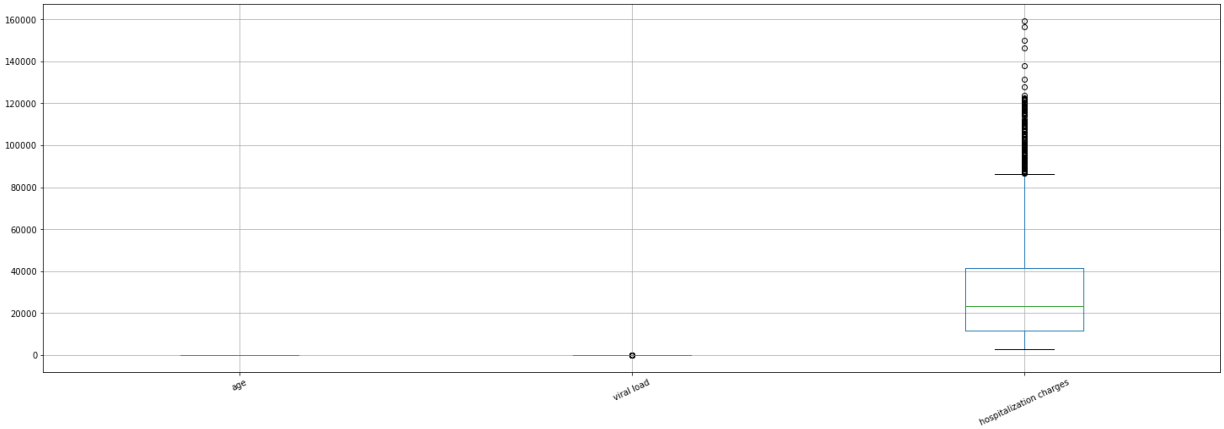
In [13]:
```
# Count the number of null values in each columns

df.isna().sum()
```

Out[13]:
```
age                        0
sex                        0
smoker                     0
region                     0
viral load                 0
severity level             0
hospitalization charges    0
dtype: int64
```

Inference: There are no missing values in the dataset.

In [14]:
```
df[numerical_cols].boxplot(rot=25, figsize=(25,8))
```

Out[14]: <AxesSubplot:>



1.c. Missing values Treatment & Outlier treatment

In [15]:
```
Q1 = df[numerical_cols].quantile(0.25)
Q3 = df[numerical_cols].quantile(0.75)
IQR = Q3 - Q1
print(IQR)
```

```
age                          24.000
viral load                    2.805
hospitalization charges    29748.500
dtype: float64
```

In [16]:
```
df = df[~((df[numerical_cols] < (Q1 - 1.5 * IQR)) |(df[numerical_cols] > (Q3 + 1.
df = df.reset_index(drop=True)
```

In [17]:
```
df
```

Out[17]:

	age	sex	smoker	region	viral load	severity level	hospitalization charges
0	19	female	yes	southwest	9.30	0	42212
1	18	male	no	southeast	11.26	1	4314
2	28	male	no	southeast	11.00	3	11124
3	33	male	no	northwest	7.57	0	54961
4	32	male	no	northwest	9.63	0	9667
...	...	...	...	...	...	...	...
1188	50	male	no	northwest	10.32	3	26501
1189	18	female	no	northeast	10.64	0	5515
1190	18	female	no	southeast	12.28	0	4075
1191	21	female	no	southwest	8.60	0	5020
1192	61	female	yes	northwest	9.69	0	72853

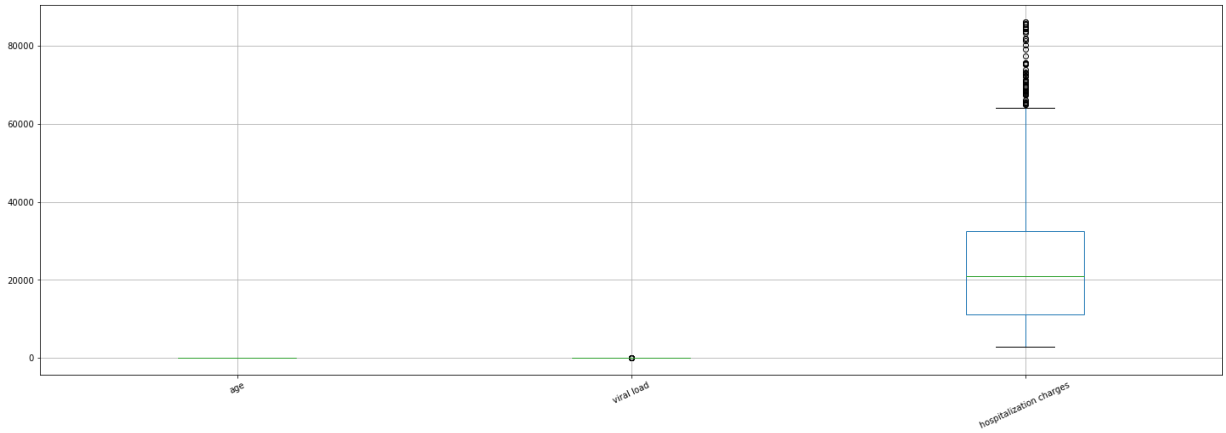
1193 rows × 7 columns

In [18]:
```
df[['age', 'viral load']].boxplot(rot=25, figsize=(25,8))
```

Out[18]: <AxesSubplot:>

In [19]:
```
df[numerical_cols].boxplot(rot=25, figsize=(25,8))
```

Out[19]: <AxesSubplot:>

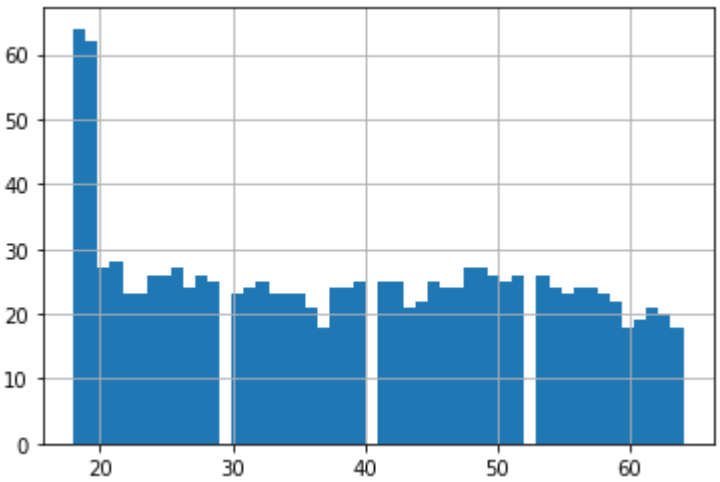


1.d. Univariate Analysis (distribution plots of all the continuous variable(s) barplots/countplots of all the categorical variables)

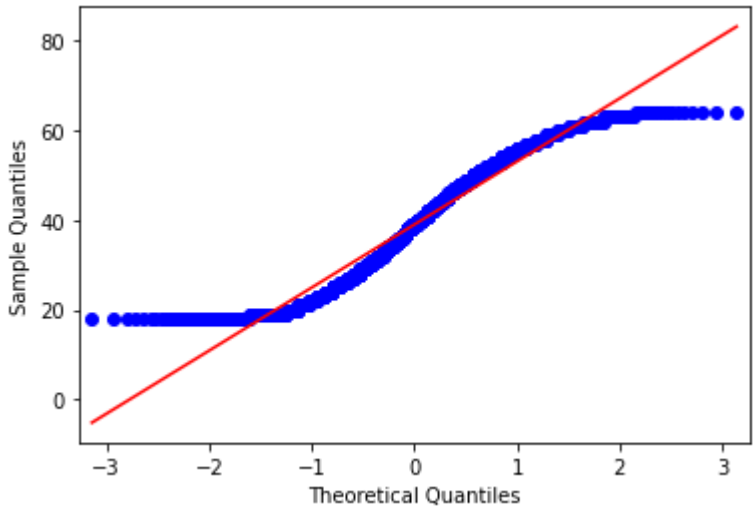
Numerical Variables

In [20]:
```
df["age"].hist(bins=50)
```

Out[20]: <AxesSubplot:>



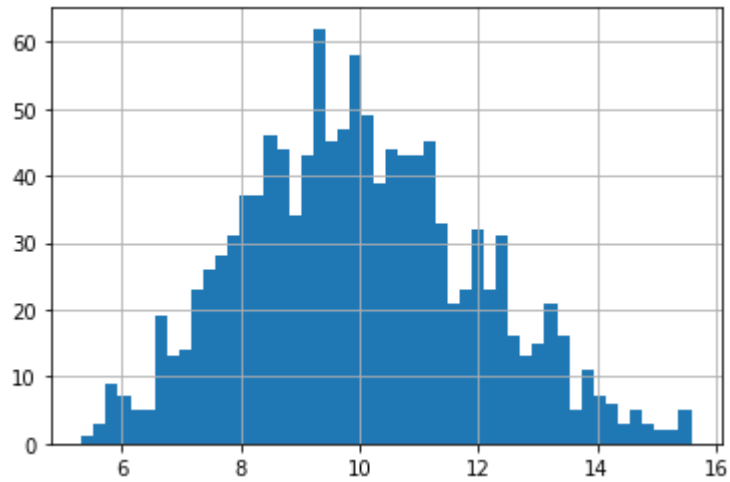
In [21]:
```
sm.qqplot(df["age"], line ='s')
py.show()
```



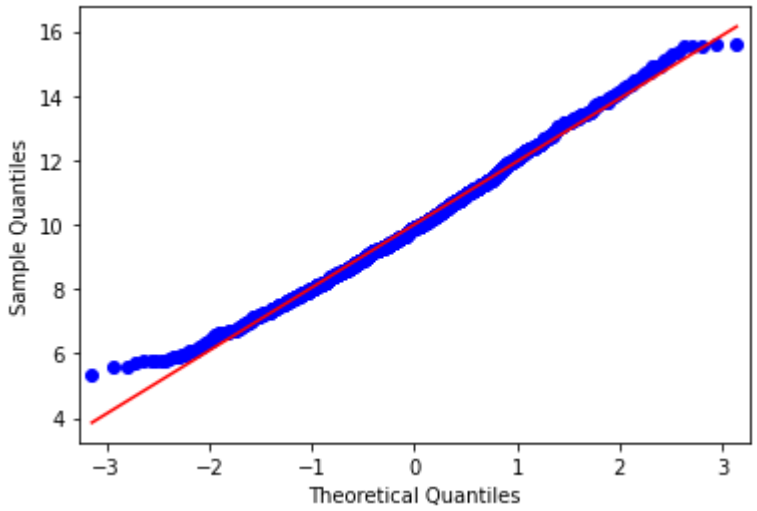
Inference: Lot of patients are children from the age 10-20 years. And it is nowhere close to Normal distribution

In [22]:
```
df["viral load"].hist(bins=50)
```

Out[22]: <AxesSubplot:>



In [23]:
```
sm.qqplot(df["viral load"], line ='s')
py.show()
```

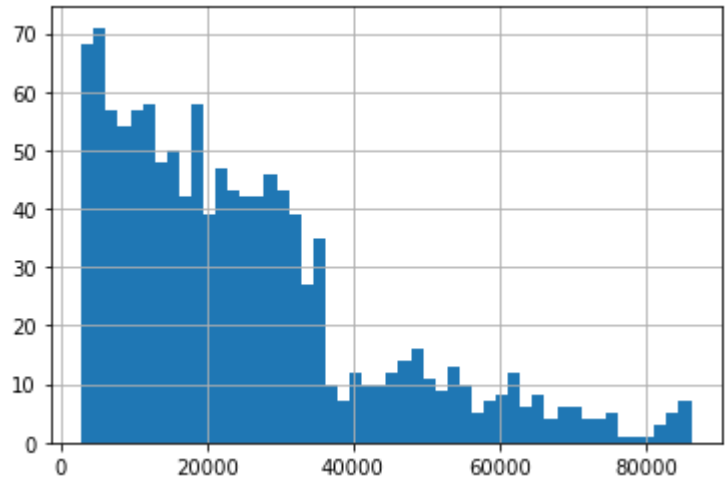


Inference: 'viral load' column is following Normal Distribution.

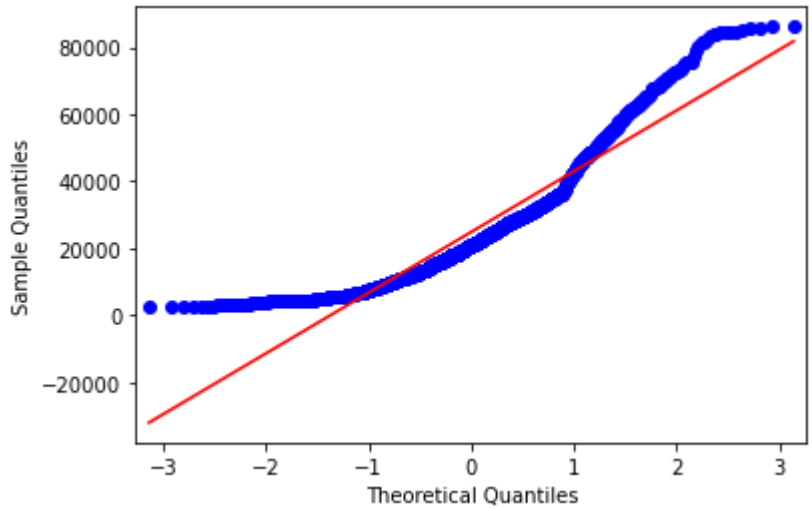
Inference: "severity level" column is not at all following Normal Distribution.

In [24]:
```
df["hospitalization charges"].hist(bins=50)
```

Out[24]: <AxesSubplot:>



In [25]:
```
sm.qqplot(df["hospitalization charges"], line ='s')
py.show()
```

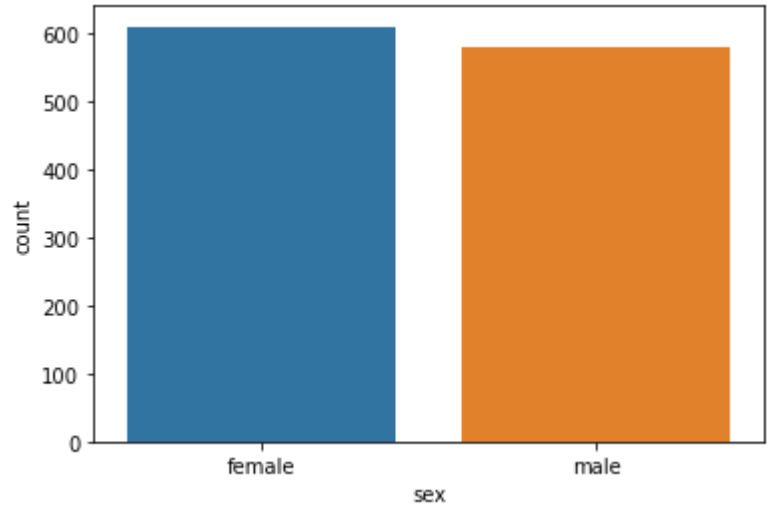


Inference: "hospitalization charges" column is not following Normal Distribution.

Categorical Columns

In [26]:
```
sns.countplot(x="sex", data=df)
```

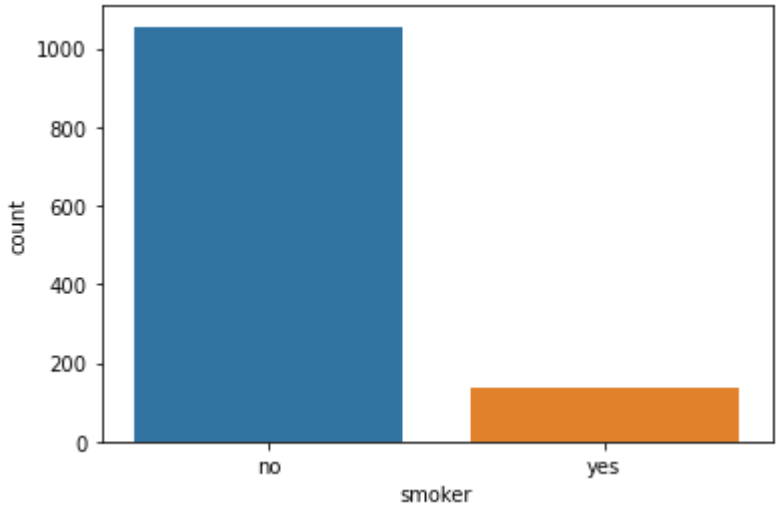
Out[26]:
```
<AxesSubplot:xlabel='sex', ylabel='count'>
```



Inference: There are equal number of male and female patients.

In [27]:
```
sns.countplot(x="smoker", data=df)
```

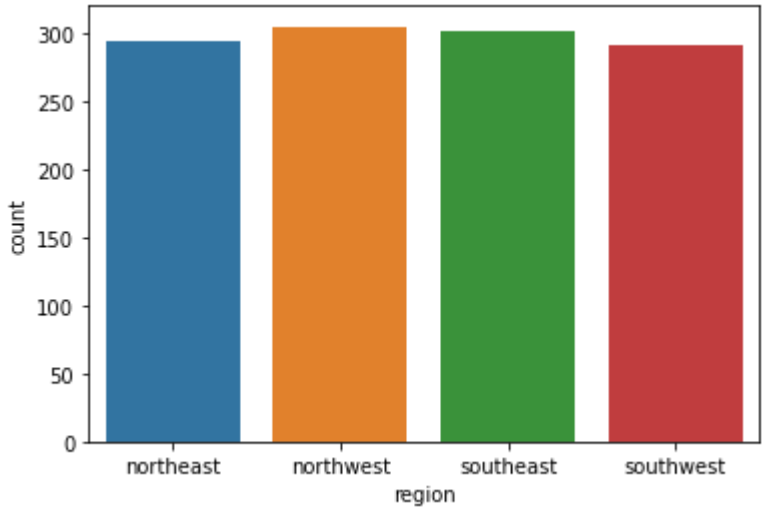
Out[27]:
```
<AxesSubplot:xlabel='smoker', ylabel='count'>
```



Inference: Smokers are very very less than Non-smokers. Only 10% of the non-smokers are smokers.

In [28]:
```
sns.countplot(x="region", data=df)
```

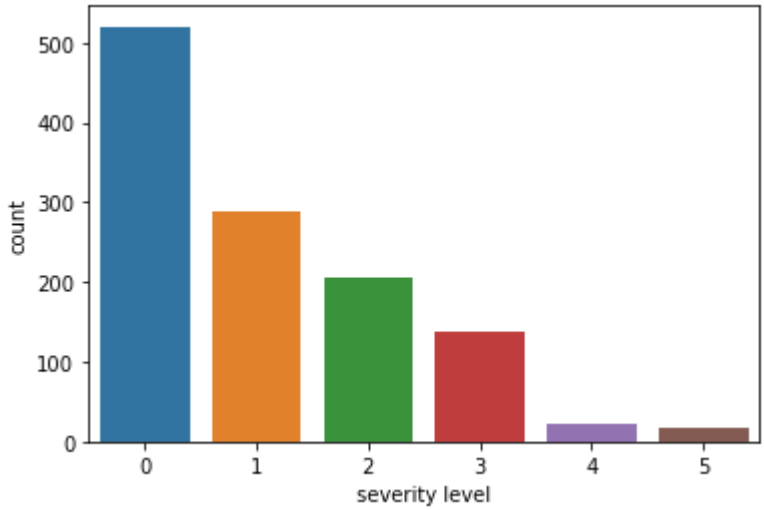
Out[28]:
```
<AxesSubplot:xlabel='region', ylabel='count'>
```



Inference: From all the regions of Delhi, there are same amount of patients.

In [29]:
```
sns.countplot(x="severity level", data=df)
```

Out[29]:
```
<AxesSubplot:xlabel='severity level', ylabel='count'>
```

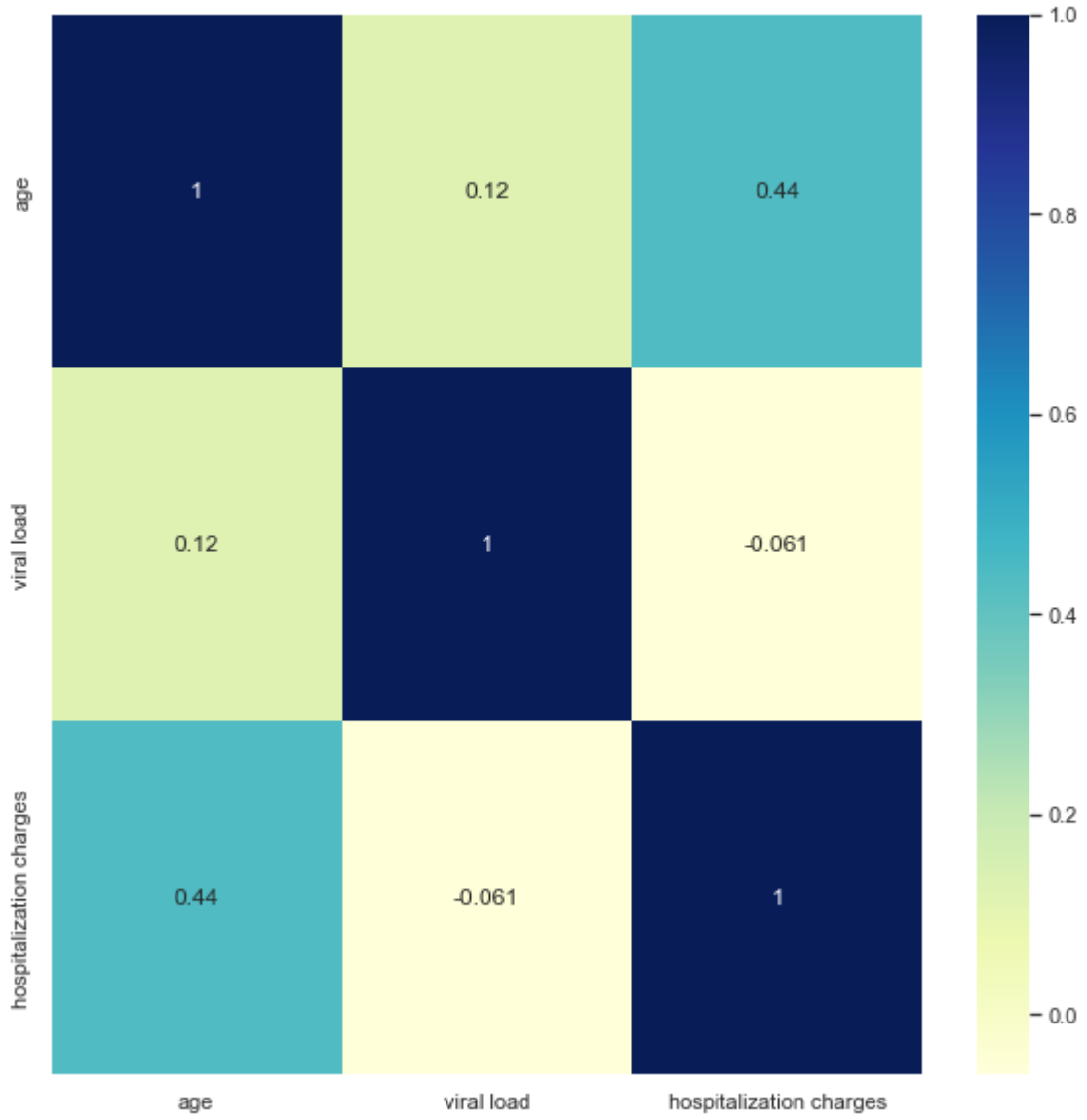


Inference: There are least number of high severity cases. As the severity increases, the number of patients belonging to that severity decreases.

1.e. Bivariate Analysis (Relationships between important variables)

In [30]:
```
sns.set(rc = {'figure.figsize':(10,10)})
sns.heatmap(df.corr(), cmap="YlGnBu", annot=True)
```

Out[30]: <AxesSubplot:>

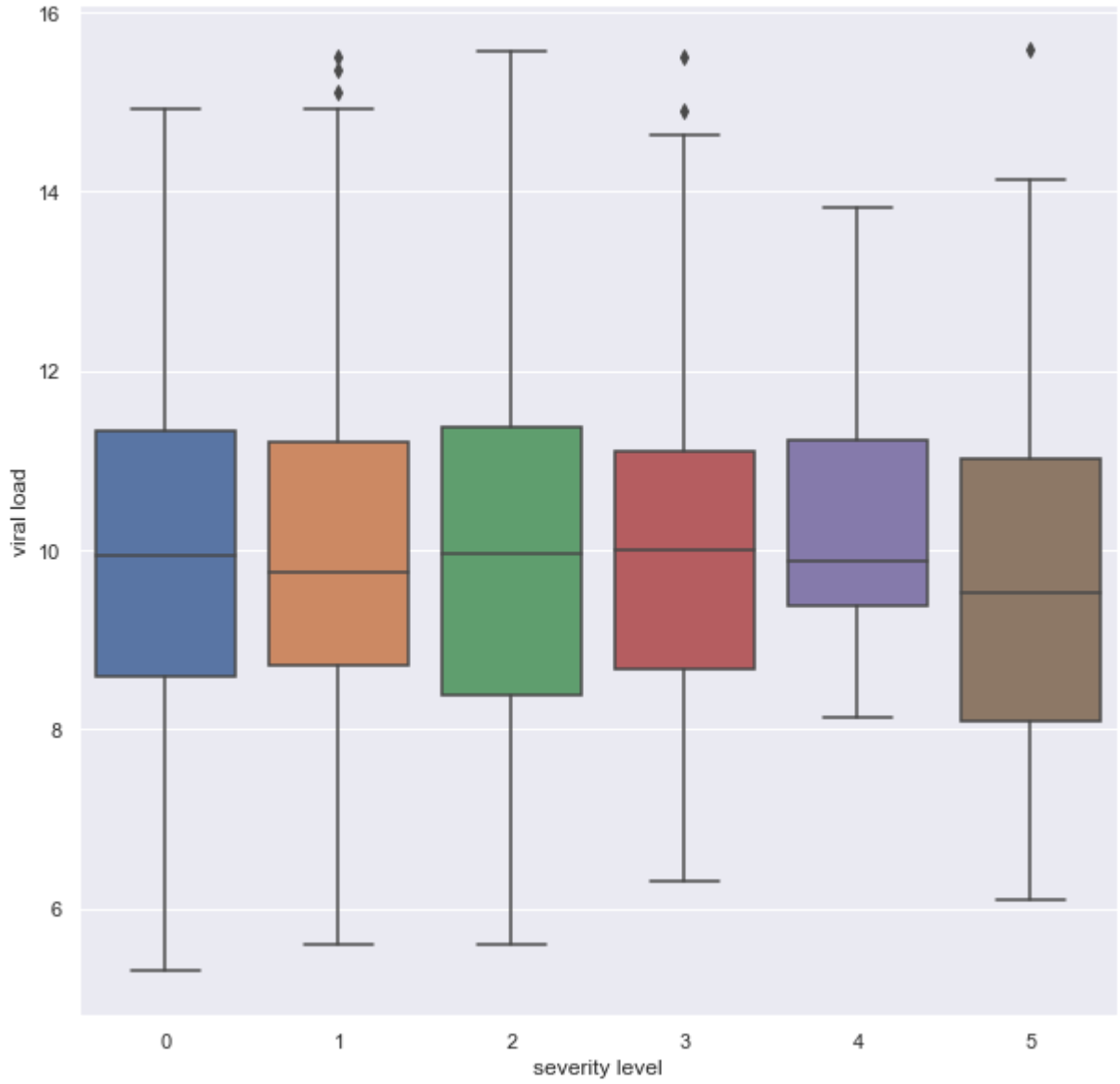


Inference: No two numerical variables are correlated to each other.

viral load v/s severity level

In [31]:
```
sns.boxplot(data = df, x = "severity level", y = "viral load")
```

Out[31]:
```
<AxesSubplot:xlabel='severity level', ylabel='viral load'>
```

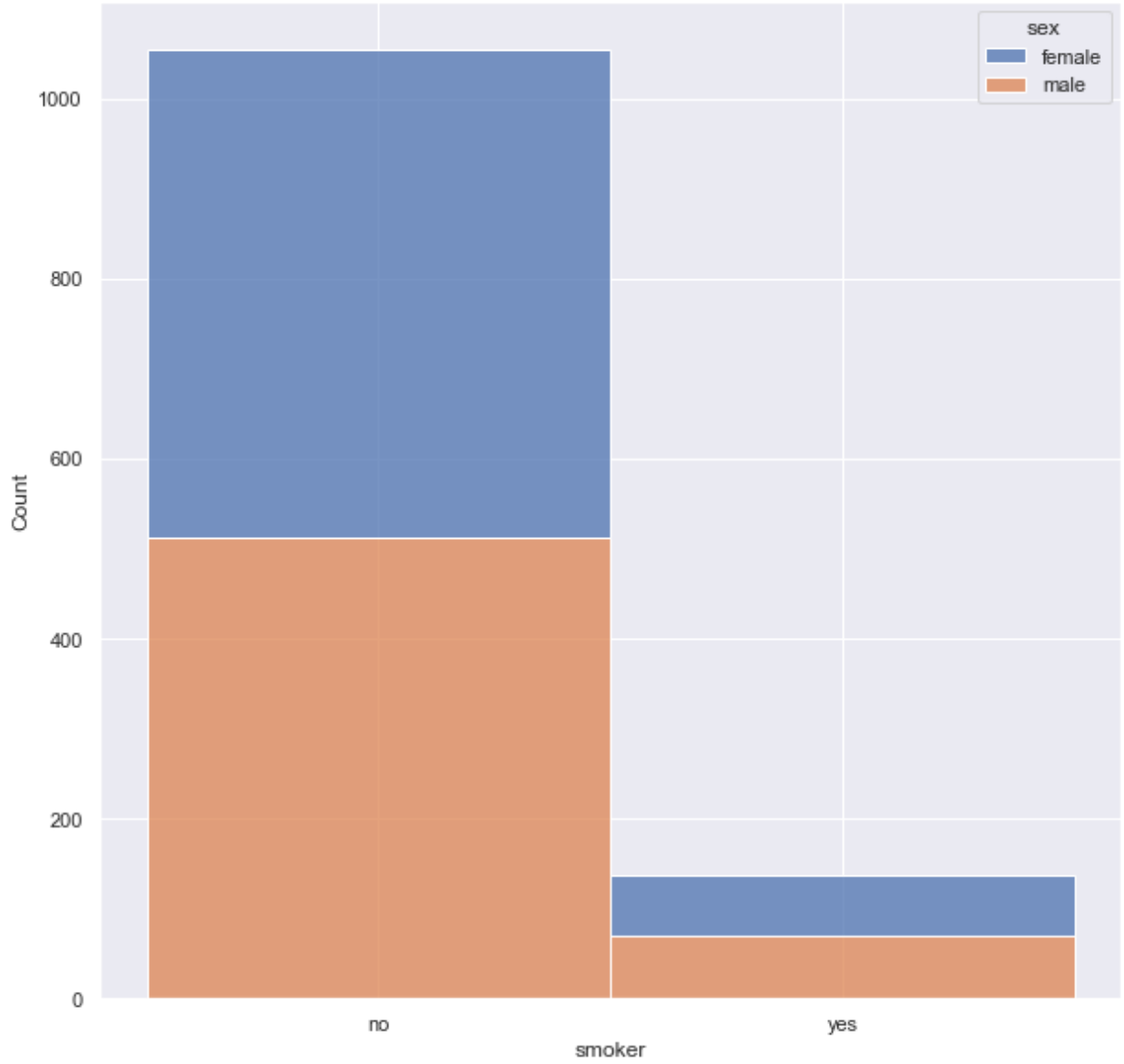


Inference: Viral load is not dependent of severity level. For all severity level, we have patients with viral load between 9 to 11.

sex v/s smoker

In [32]:
```
sns.histplot(binwidth=0.5, x="smoker", hue="sex", data=df, stat="count", multiple
```

Out[32]: <AxesSubplot:xlabel='smoker', ylabel='Count'>

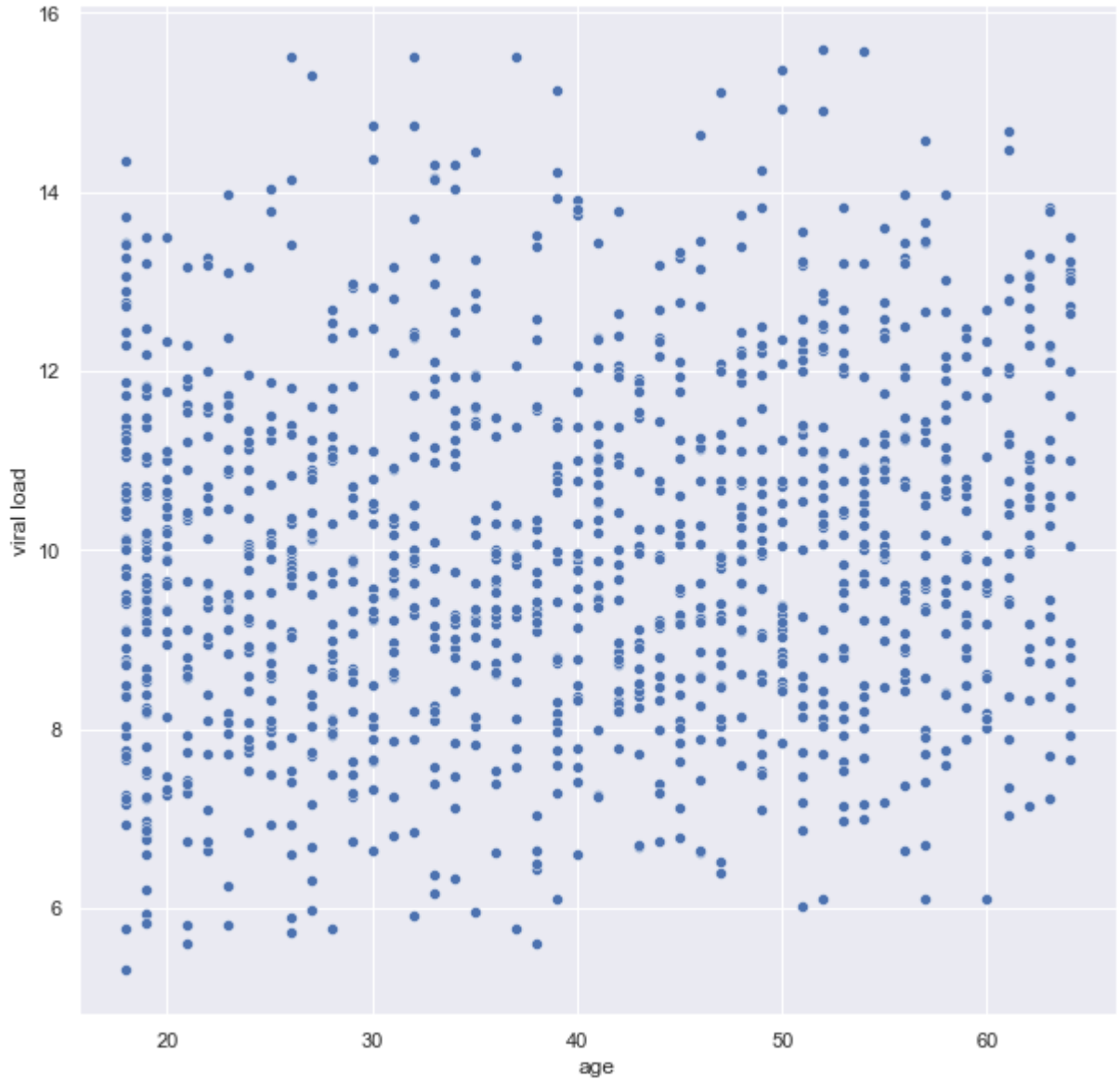


Inference: Among smokers, 50% people are men and rest 50% people are women.

age v/s viral load

In [33]:
```
sns.scatterplot(data=df, x="age", y="viral load")
```

Out[33]:
```
<AxesSubplot:xlabel='age', ylabel='viral load'>
```

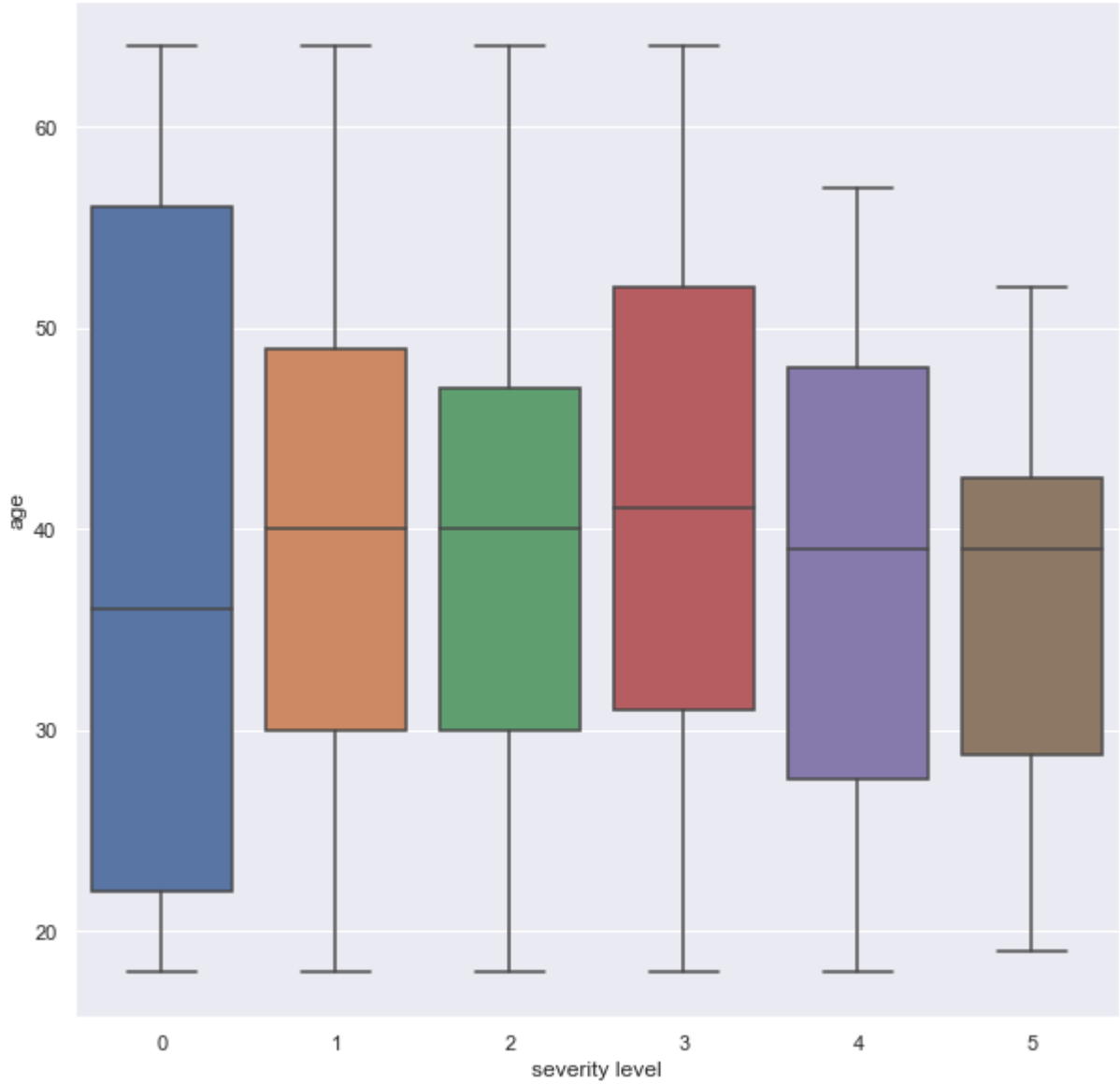


Inference: Age and viral load are independent to each other. I do not find any relationship between age and viral load.

age v/s severity level

In [34]:
```
sns.boxplot(data = df, x = "severity level", y = "age")
```

Out[34]:
```
<AxesSubplot:xlabel='severity level', ylabel='age'>
```

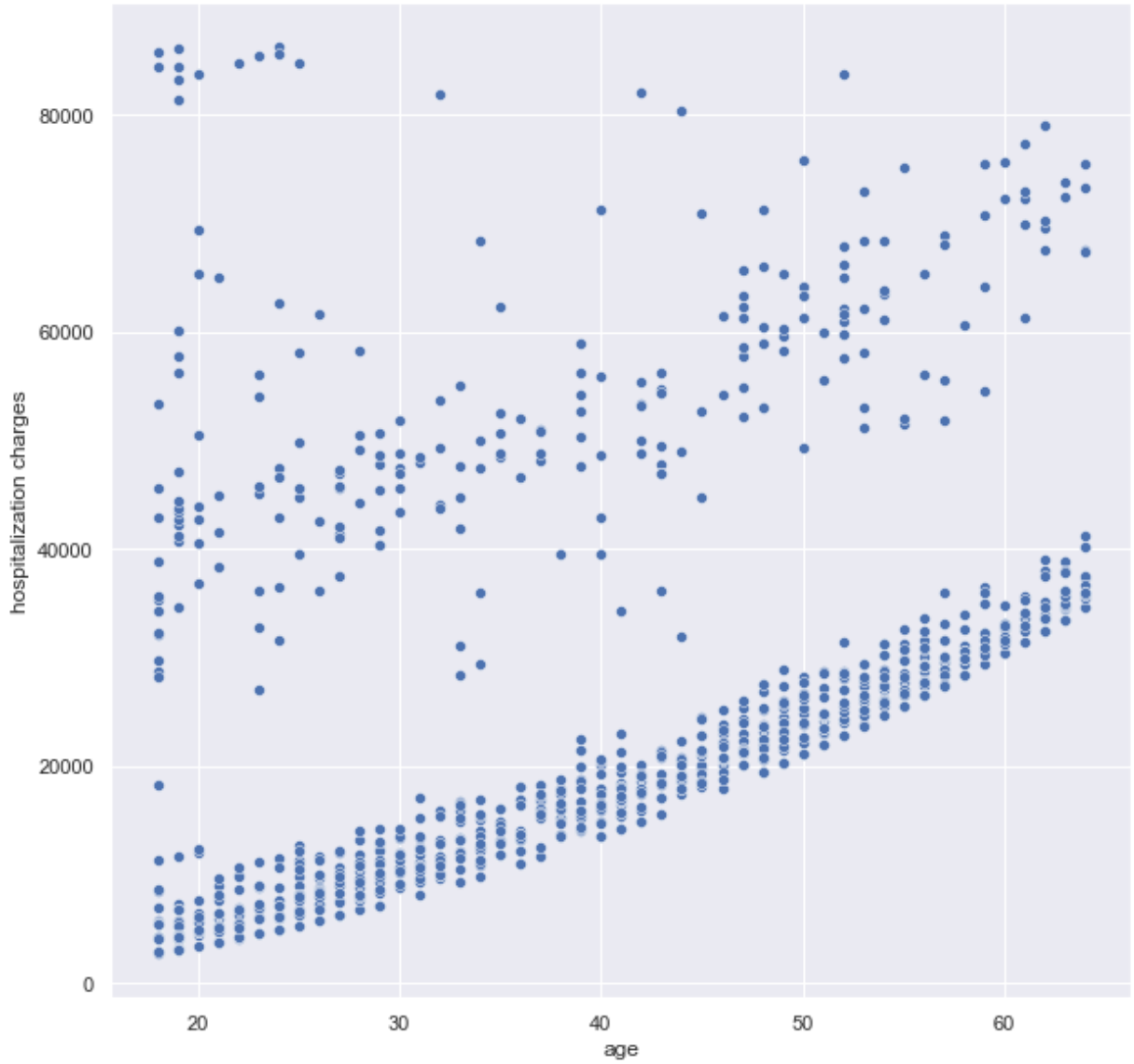


Inference: On an average people at age=40 have some severity.

age v/s hospitalization charges

In [35]:
```
sns.scatterplot(data=df, x="age", y="hospitalization charges")
```

Out[35]:
```
<AxesSubplot:xlabel='age', ylabel='hospitalization charges'>
```

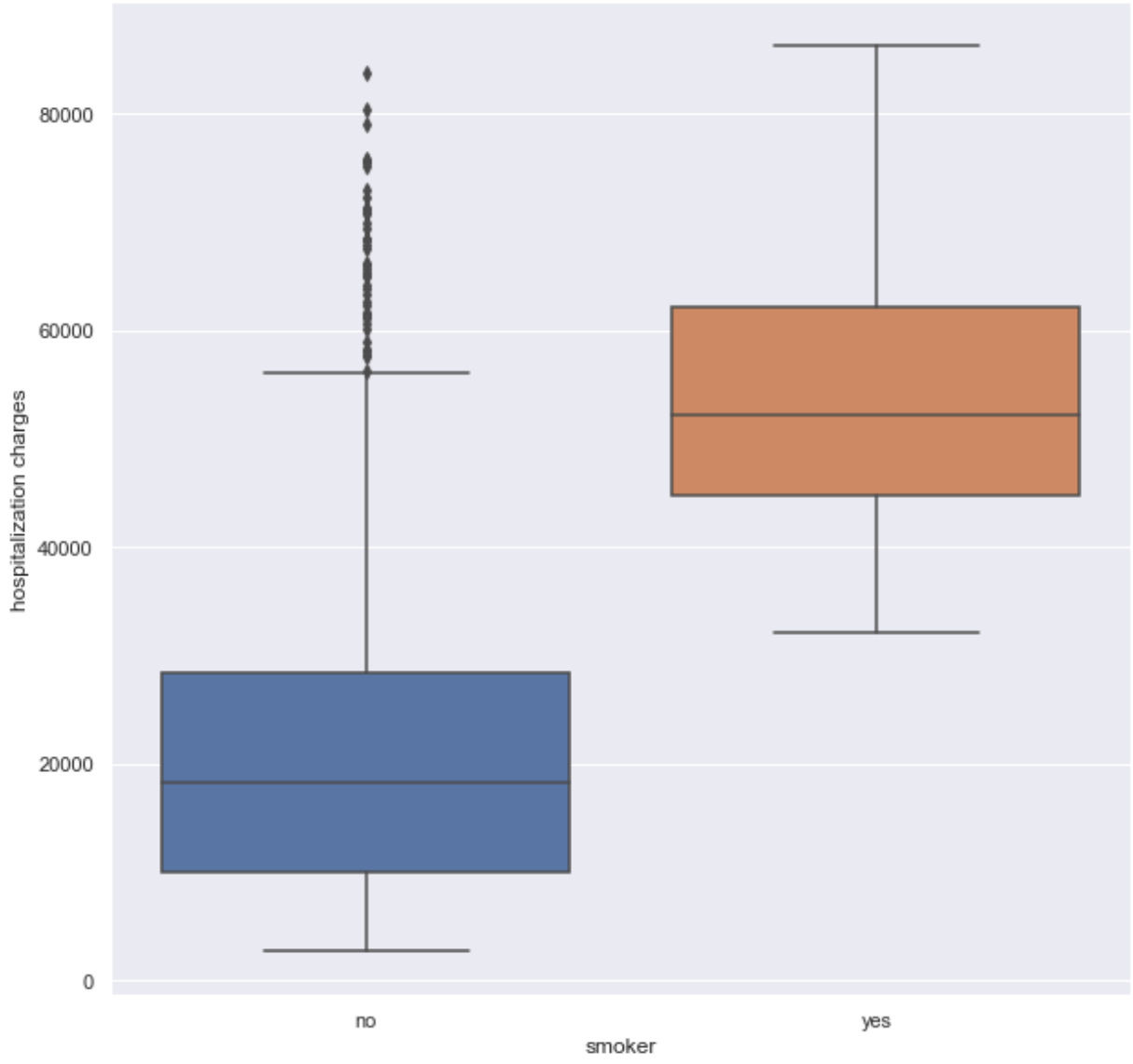


Inference: I can see some upward trend which tells that as the age increases, your hospital bills also increases.

hospitalization charges v/s smoker

In [36]:
```
sns.boxplot(data = df, x = "smoker", y = "hospitalization charges")
```

Out[36]:
```
<AxesSubplot:xlabel='smoker', ylabel='hospitalization charges'>
```

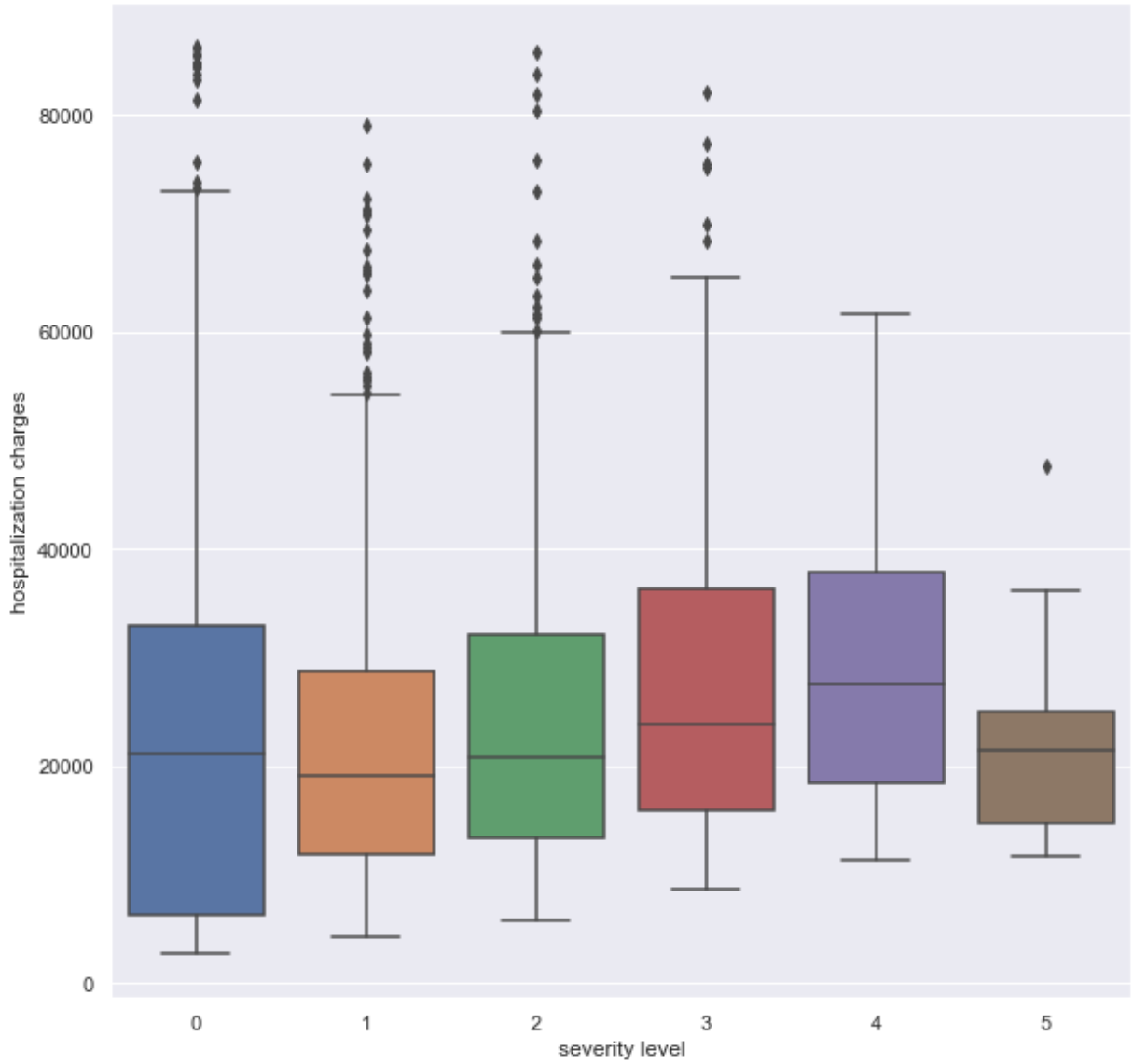


Inference: It is clearly evident that people who smoke tend to spend more on hospitalization charges.

hospitalization charges v/s severity level

In [37]:
```
sns.boxplot(data = df, x = "severity level", y = "hospitalization charges")
```

Out[37]:
```
<AxesSubplot:xlabel='severity level', ylabel='hospitalization charges'>
```

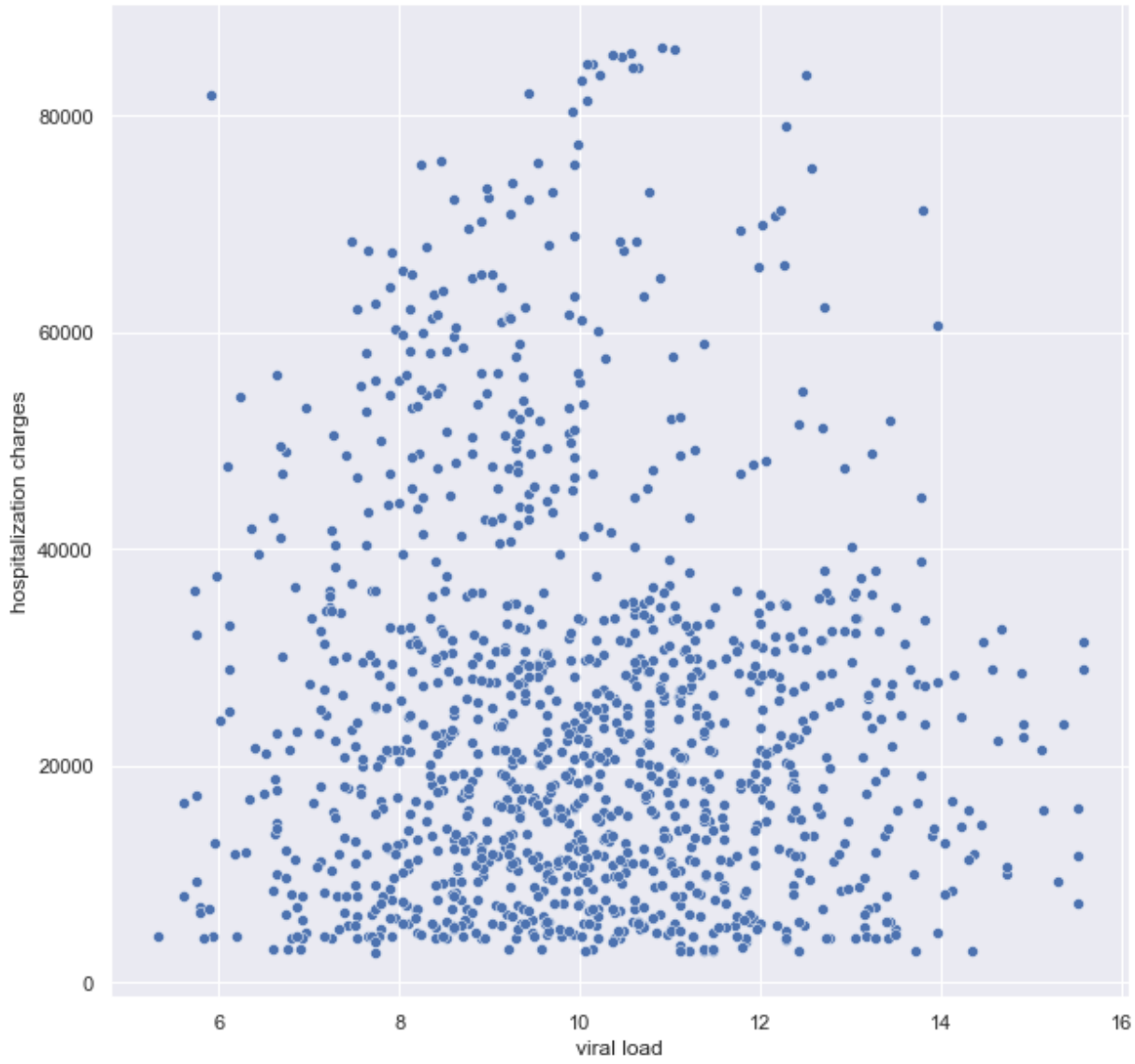


Inference: Even with 0 severity, you can have high hospitalization charges. As the severity increases, the hospitalization charges also increases. Since there are lesser number of severity-5 people, the trend has gone down from severity-4 people.

hospitalization charges v/s viral load

In [38]:
```
sns.scatterplot(data=df, x="viral load", y="hospitalization charges")
```

Out[38]:
```
<AxesSubplot:xlabel='viral load', ylabel='hospitalization charges'>
```

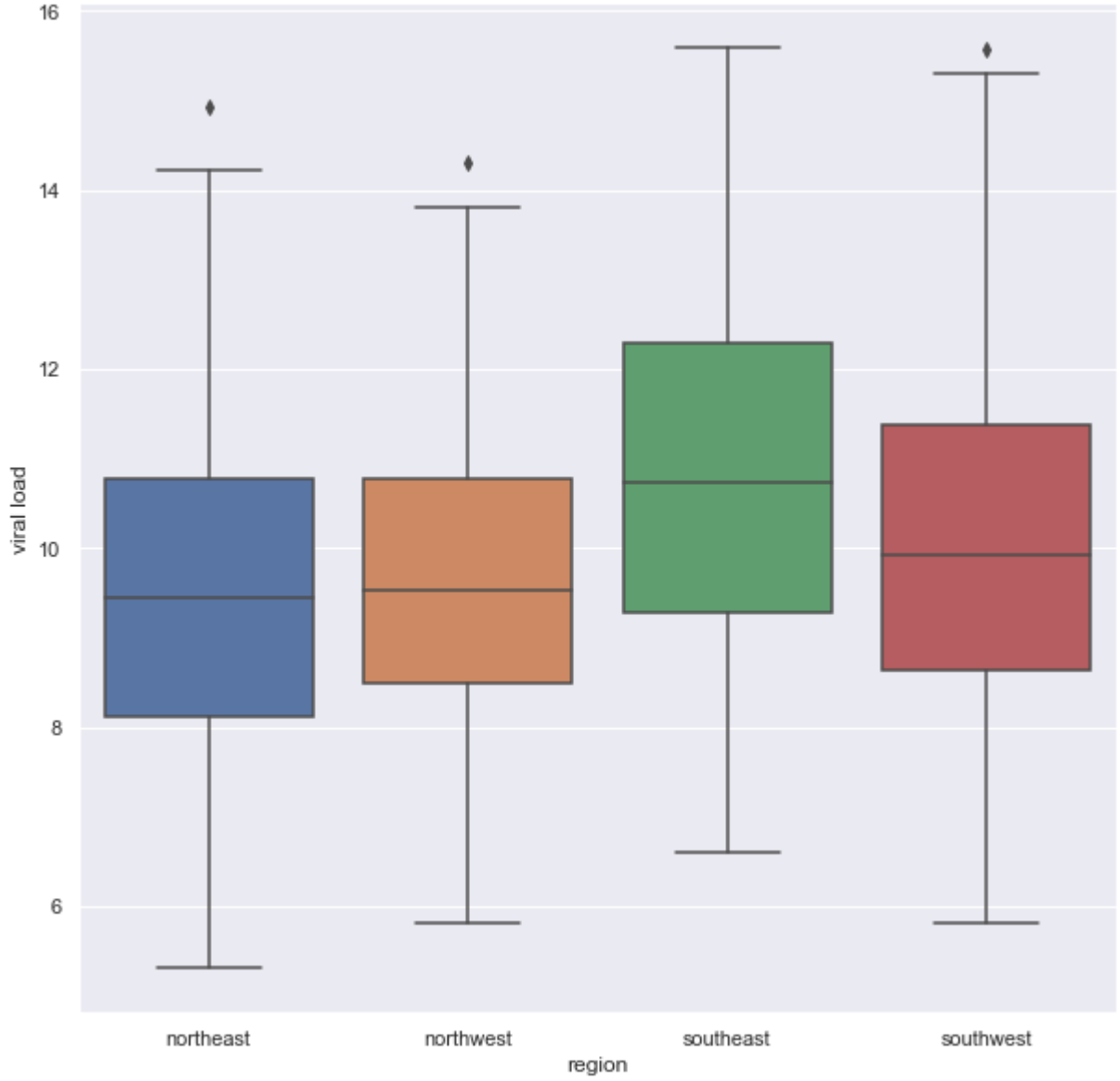


Inference: There is no relationship between viral load and hospitalization charges.

region v/s viral load

In [39]:
```
sns.boxplot(data = df, x = "region", y = "viral load")
```

Out[39]:
```
<AxesSubplot:xlabel='region', ylabel='viral load'>
```

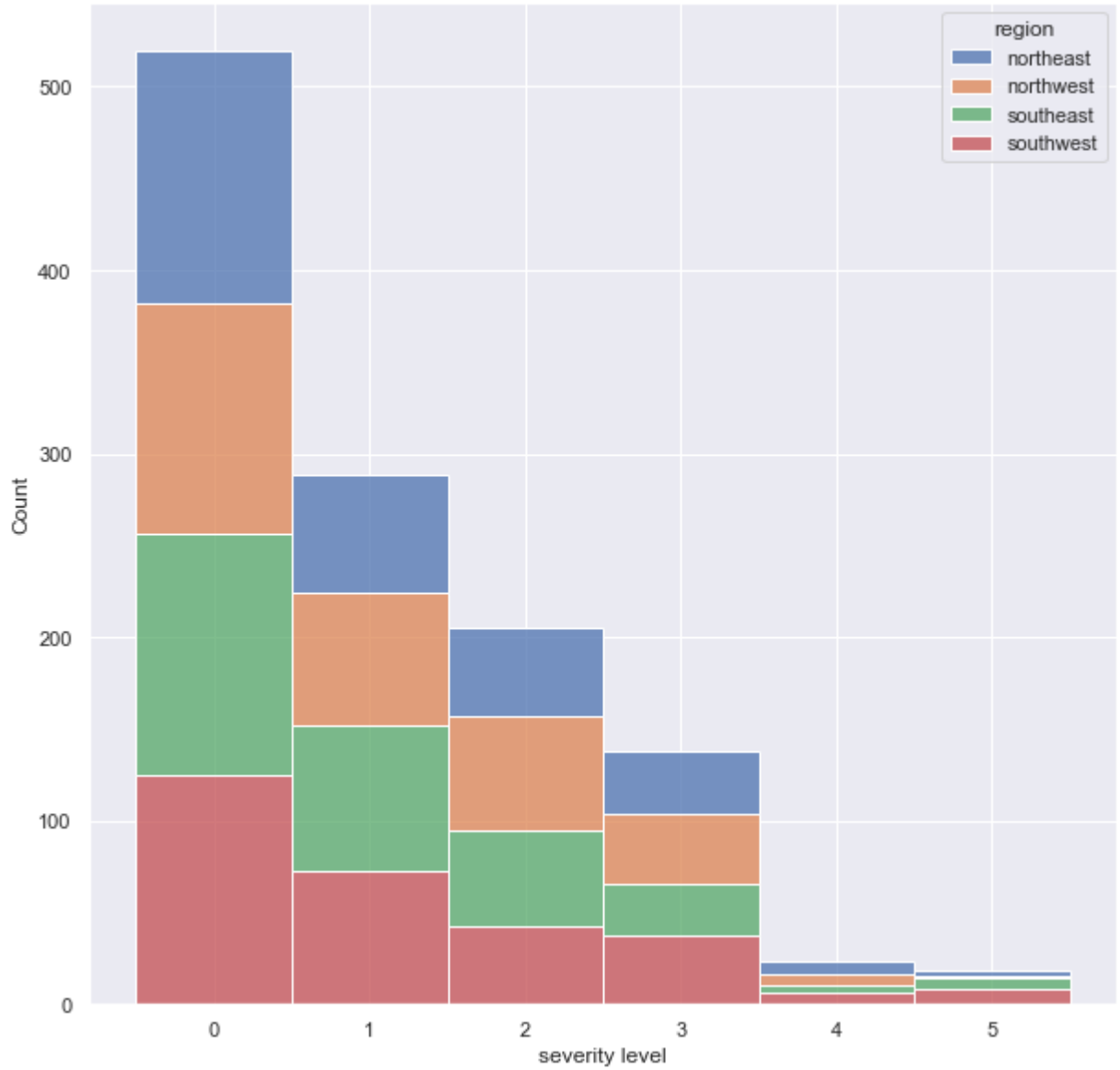


Inference: We can observe that there is higher viral loads in southeast and southwest part of Delhi

region v/s severity level

In [40]:
```
sns.histplot(binwidth=0.5, x="severity level", hue="region", data=df, stat="count
```

Out[40]:
```
<AxesSubplot:xlabel='severity level', ylabel='Count'>
```

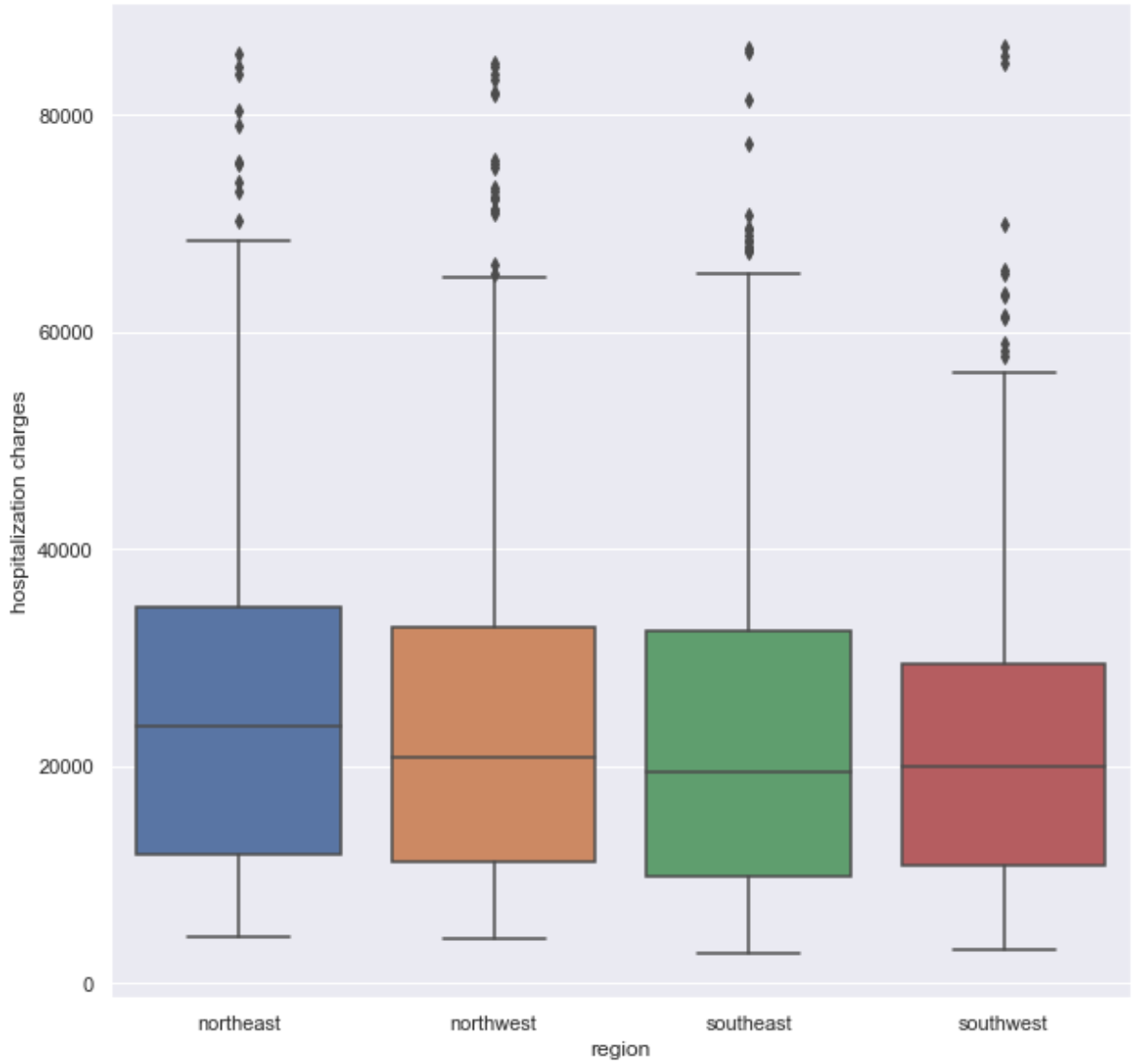


Inference: There is almost same number of people from each region with particular severity.

region v/s hospitalization charges

In [41]:
```
sns.boxplot(data = df, x = "region", y = "hospitalization charges")
```

Out[41]:
```
<AxesSubplot:xlabel='region', ylabel='hospitalization charges'>
```



Inference: For all the regions, hospital bills coming are almost same.

In [ ]:
```

```

2. Hypothesis Testing

2.a. Prove (or disprove) that the hospitalization charges of people who do smoking are greater than those who don't?

Let,
ho: hospitalization charges of people who do smoking <= hospitalization charges of people who donot do smoking.
ha: hospitalization charges of people who do smoking >= hospitalization charges of people who donot do smoking.\

T-test Right tailed

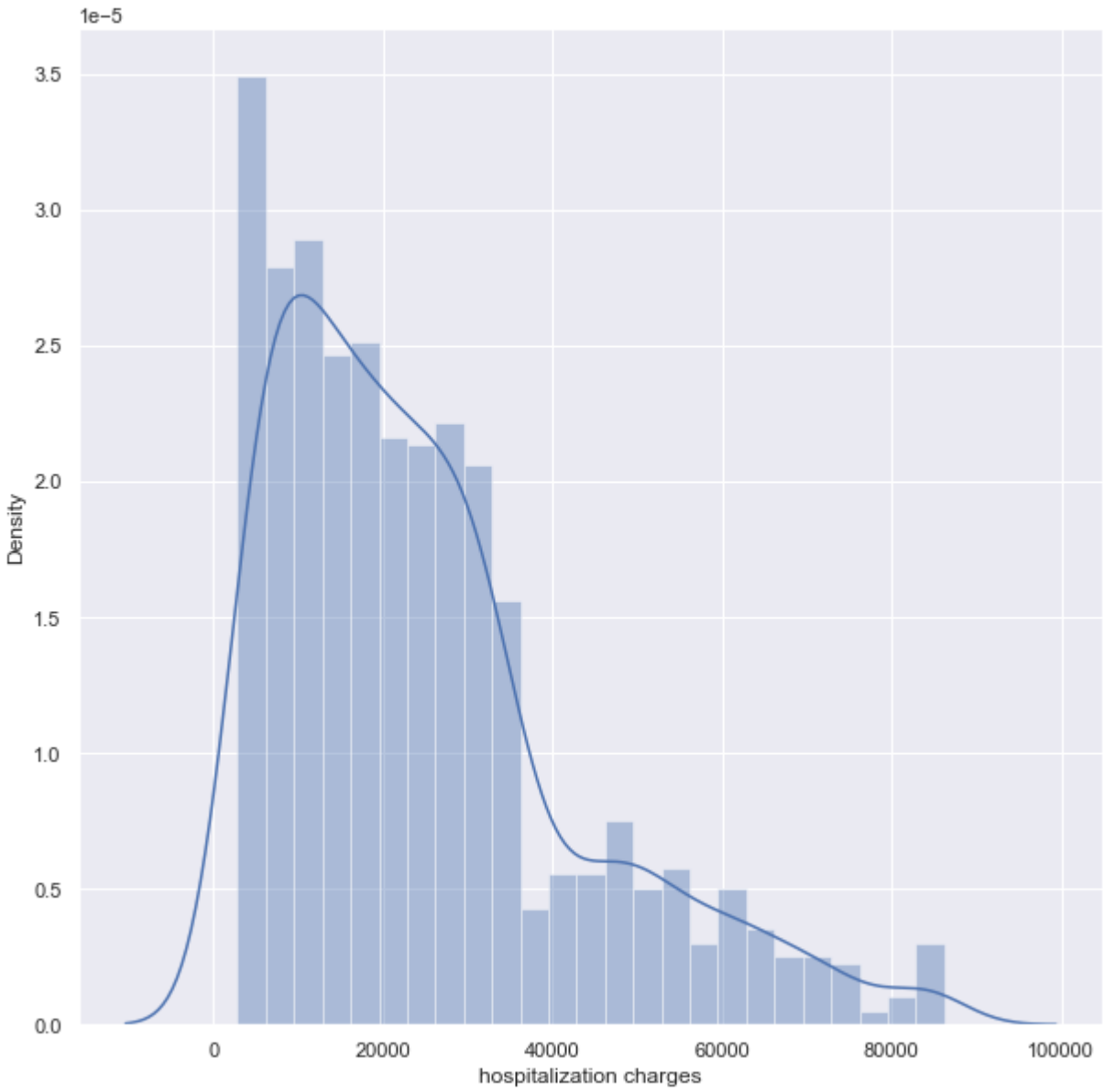
In [42]:
```
# I am not sampling from the dataset because we just have around 1000 datapints.
```

In [43]:
```
df["smoker"].value_counts()
```

Out[43]:
```
no     1055
yes     138
Name: smoker, dtype: int64
```

In [44]:
```
sns.distplot(df["hospitalization charges"], bins = 25)
```

Out[44]:
```
<AxesSubplot:xlabel='hospitalization charges', ylabel='Density'>
```

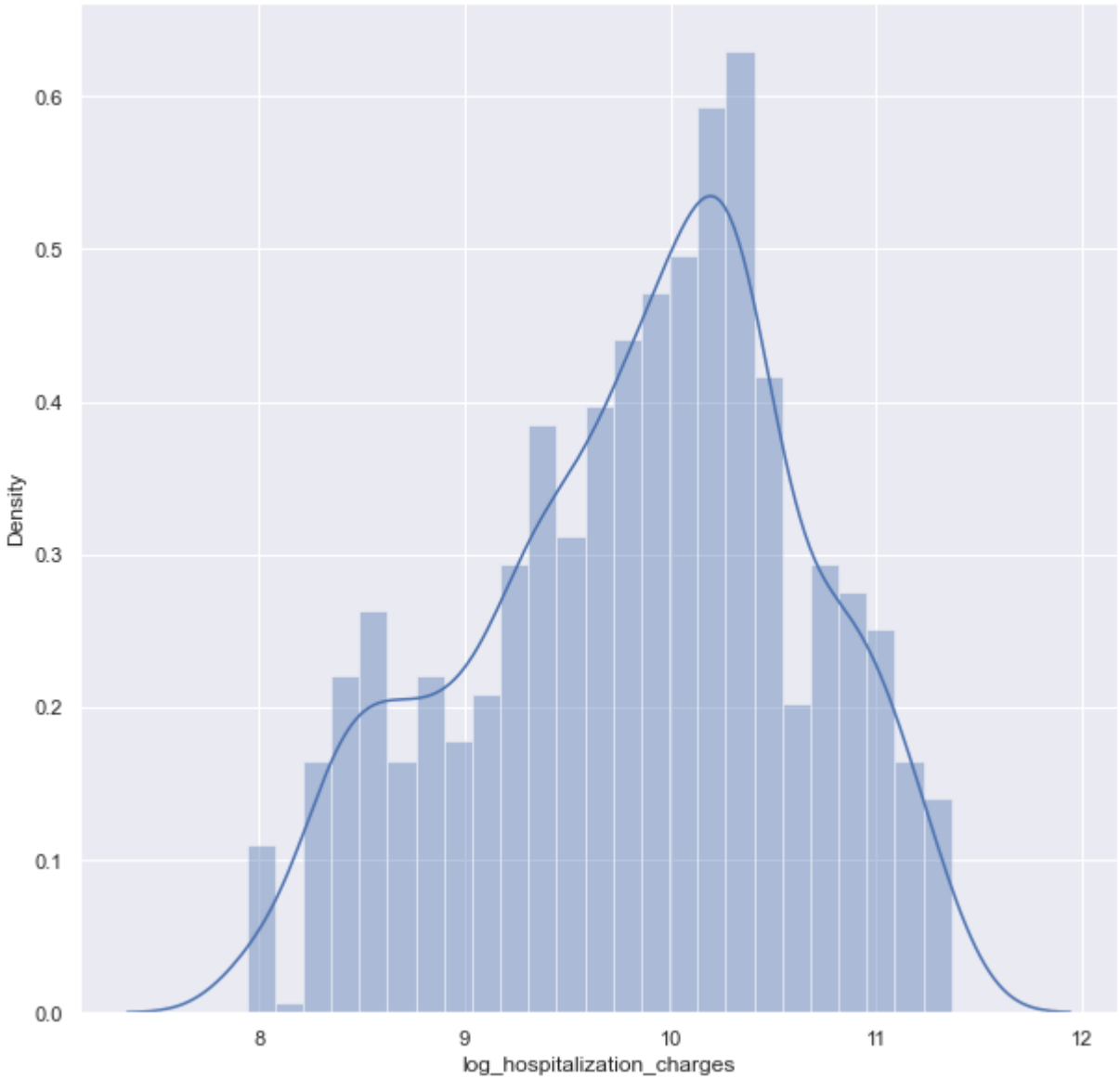


Inference: It looks like Log-normal distribution.

In [45]:
```
# Checking whether the above distribution is a Log normal or not.
df["log_hospitalization_charges"] = np.log(df["hospitalization charges"])
```

In [46]:
```
sns.distplot(df["log_hospitalization_charges"], bins = 25)
```

Out[46]:
```
<AxesSubplot:xlabel='log_hospitalization_charges', ylabel='Density'>
```



In [47]:
```
# Finding out the means for both the groups

df.groupby("smoker")["hospitalization charges"].mean()
```

Out[47]:
```
smoker
no     20907.971564
yes    55035.586957
Name: hospitalization charges, dtype: float64
```

Inference: Avg hospitalization charges of smokers is ~60% more than that of non-smokers.

In [48]:
```
# separating the data for treatment group and control group.

df_non_smokers = df[df["smoker"] == "no"]
df_smokers = df[df["smoker"] == "yes"]
```

In [49]:
```
df_smokers["hospitalization charges"].mean() - df_non_smokers["hospitalization ch
```

Out[49]:
```
34127.6153925407
```

In [50]:
```
stats.ttest_ind( df_smokers["hospitalization charges"], df_non_smokers["hospitali
```

Out[50]:
```
Ttest_indResult(statistic=26.042742964009594, pvalue=1.0)
```

Here pvalue<0.05, that difference of ~Rs.29,000 is significant enough to say that there is a huge difference between hospitalization charges of non-smokers group and smokers group. So we will be rejecting our Null Hypothesis(ho) and accept Alternate Hypothesis(ha).

Lets pick the t-critical values by the t-distribution table.
We are going with Two sample one-sided t-test.
From this table, since sample size is more than 1,000, z-stats for 0.05 Confidence is ~1.64.
Therefore tcritical = 1.64.
And tstats = 26.04, which is greater than tcritical.
Therefore we reject the Null Hypothesis and go with Alternate Hypothesis.

Therefore, hospitalization charges of people who do smoking >= hospitalization charges of people who donot do smoking.

In [ ]:

2.b. Prove (or disprove) with statistical evidence that the viral load of females is different from that of males

Let,
ho: viral load of females == viral load of males
ha: viral load of females != viral load of males\

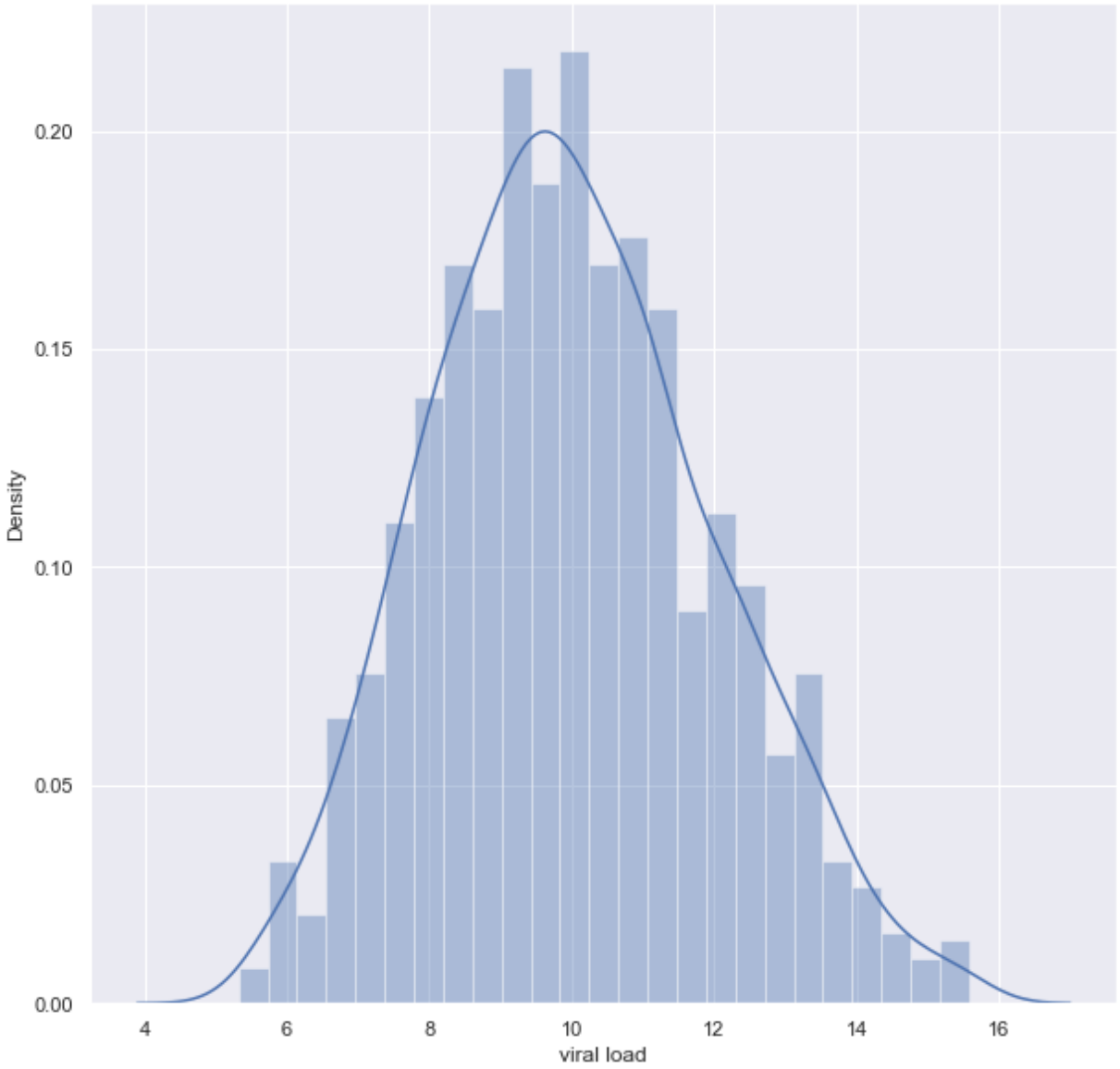
T-test Two tailed

In [51]:
```
df["sex"].value_counts()
```

Out[51]:
```
female    611
male      582
Name: sex, dtype: int64
```

In [52]:
```
sns.distplot(df["viral load"], bins = 25)
```

Out[52]:
```
<AxesSubplot:xlabel='viral load', ylabel='Density'>
```



Inference: viral load follows Normal distribution.

In [53]:
```
# Finding out the means for both the groups
df.groupby("sex")["viral load"].mean()
```

Out[53]:
```
sex
female     9.968298
male      10.032440
Name: viral load, dtype: float64
```

In [54]:
```
# separating the data for treatment group and control group.

df_male = df[df["sex"] == "male"]
df_female = df[df["sex"] == "female"]
```

In [55]:
```
df_male["viral load"].mean() - df_female["viral load"].mean()
```

Out[55]:
```
0.06414199020253086
```

In [56]:
```
stats.ttest_ind(df_male["viral load"], df_female["viral load"], alternative="less
```

Out[56]:
```
Ttest_indResult(statistic=0.5660752690218366, pvalue=0.7142753730121758)
```

Here pvalue>0.05, that difference of ~viral load=0.06 is not that significant enough to say that there is a huge difference between viral loads in male and female. So we are keeping our Null Hypothesis(ho) as it is.

Lets pick the t-critical values by the t-distribution table.
We are going with Two sample two-sided t-test.
From this table, since sample size is more than 1,000, z-stats for 0.05 Confidence is ~1.64.
Therefore tcritical = 1.64.
And tstats = 0.566, which is lesser than tcritical.
Therefore we accept the Null Hypothesis as it is.

Therefore, viral load of females == viral load of males.

In [ ]:

2.c. Is the proportion of smoking significantly different across different regions?

Let,
ho: Proportion of smoking people is same across different regions
ha: Proportion of smoking people is different across different regions

Chi-square Test

In [57]:
```
ctab = pd.crosstab(df['smoker'], df['region'])
ctab
```

Out[57]:

smoker \ region	northeast	northwest	southeast	southwest
no	256	267	267	265
yes	39	38	35	26

In [58]:
```
from scipy.stats import chi2_contingency
```

In [59]:
```
stat, p, dof, expected = chi2_contingency(ctab)
```

In [60]:
```
print("Chi-square stat:",stat)
print("pvalue :",p)
```

```
Chi-square stat: 2.99680663546149
pvalue : 0.39211779235957156
```

dof = (r-1)*(c-1) = 3
alpha = 0.05(for 95% Confidence)
chi-stats = 2.9968
See the Chi-square distribution table to know the chi-crit.
chi-crit = 7.815
chi-crit > chi-stats and pvalue>0.05, therefore we keep our Null hypothesis.

Therefore, there is relationship btw weather and season ie weather and season are dependent on each other.

In [ ]:

2.d. Is the mean viral load of women with 0 Severity level , 1 Severity level, and 2 Severity level the same? Explain your answer with statistical evidence

Let,
ho: mean viral load of women with 0 Severity level , 1 Severity level, and 2 Severity level the same
ha: mean viral load of women with 0 Severity level , 1 Severity level, and 2 Severity level the different

One way Anova

In [61]:
```
df.sex.value_counts()
```

Out[61]:
```
female    611
male      582
Name: sex, dtype: int64
```

In [62]:
```
df.groupby('sex').agg({'viral load':'mean'}).reset_index(drop=True)
df_female = df[df["sex"]=='female']
```

In [63]:
```
female_severity_0 = df_female[df_female["severity level"]==0]["viral load"].sampl
female_severity_1 = df_female[df_female["severity level"]==1]["viral load"].sampl
female_severity_2 = df_female[df_female["severity level"]==2]["viral load"].sampl

female_severity_0 = female_severity_0.reset_index(drop=True)
female_severity_1 = female_severity_1.reset_index(drop=True)
female_severity_2 = female_severity_2.reset_index(drop=True)

dataset = pd.DataFrame({"0":female_severity_0, "1":female_severity_1, "2":female_
dataset
```

Out[63]:

	0	1	2
0	10.61	6.74	8.23
1	9.07	6.60	8.84
2	12.16	8.80	7.85
3	11.04	9.20	10.77
4	10.89	8.91	14.45
...	...	...	...
95	9.57	12.17	5.73
96	13.40	12.43	11.60
97	9.53	8.95	9.90
98	12.19	8.11	7.99
99	10.29	9.47	10.23

100 rows × 3 columns

In [64]:
```
dataset.isna().sum()
```

Out[64]:
```
0    0
1    0
2    0
dtype: int64
```

In [65]:
```
dataset['0'].mean()
```

Out[65]:
```
10.105800000000006
```

In [66]:
```
dataset['1'].mean()
```

Out[66]:
```
9.725000000000001
```

In [67]:
```
dataset['2'].mean()
```

Out[67]:
```
9.886600000000001
```

In [68]:
```
f, pval = stats.f_oneway(dataset["0"], dataset["1"], dataset["2"])
print(f, pval)
```

```
0.8789733441537709 0.4162862157828492
```

Inference: Since pvalue>0.05, we keep our Null hypothesis as it is.

Therefore, mean viral load of women with 0 Severity level , 1 Severity level, and 2 Severity level the same

In [ ]:
```

```