Python初級數據分析員證書

（六）數據分析 及可視化專案

13. 數據分析專案 Demo 18 – Insurance Bill

Review

- Statistics
- Hypothesis testing
- Algebra
- Linear regression
- Propositional logic
- Python
- R
- SQL

- Pandas, NumPy, SciPy
- Data Visualization, Matplotlib, Seaborn, Plotly
- Dashboard Visualization, Business Intelligence
- Storytelling

Chapter Summary

- Scenario
- Data Import
- Data Wrangling
- EDA
- Corrwith, Heatmap

Scenario

In this chapter we go into what factors influenced the insurance bill charge of a specific patient. In order to do this we must look for patterns in our data analysis and gain extensive insight of what the data is telling us.

Acknowledgements

https://github.com/stedy/Machine-Learning-with-R-datasets

Data import

```
1 import numpy as np
2 import pandas as pd
3 import matplotlib.pyplot as pl
4 import seaborn as sns
5 from sklearn.preprocessing import LabelEncoder
6 import warnings
7 #warnings.filterwarnings('ignore')
```

```
1 data = pd.read_csv('insurance.csv')
2 data.head(5)
```

	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

Columns

- age: age of primary beneficiary
- sex: insurance contractor gender, female, male
- bmi: Body mass index, providing an understanding of body, weights that are relatively high or low relative to height,
  objective index of body weight (kg / m ^ 2) using the ratio of height to weight, ideally 18.5 to 24.9
- children: Number of children covered by health insurance / Number of dependents
- smoker: Smoking
- region: the beneficiary's residential area in the US, northeast, southeast, southwest, northwest.
- charges: Individual medical costs billed by health insurance

Overview all columns

```
1 data.describe(include='all')
```

	age	sex	bmi	children	smoker	region	charges
count	1338.000000	1338	1338.000000	1338.000000	1338	1338	1338.000000
unique	NaN	2	NaN	NaN	2	4	NaN
top	NaN	male	NaN	NaN	no	southeast	NaN
freq	NaN	676	NaN	NaN	1064	364	NaN
mean	39.207025	NaN	30.663397	1.094918	NaN	NaN	13270.422265
std	14.049960	NaN	6.098187	1.205493	NaN	NaN	12110.011237
min	18.000000	NaN	15.960000	0.000000	NaN	NaN	1121.873900
25%	27.000000	NaN	26.296250	0.000000	NaN	NaN	4740.287150
50%	39.000000	NaN	30.400000	1.000000	NaN	NaN	9382.033000
75%	51.000000	NaN	34.693750	2.000000	NaN	NaN	16639.912515
max	64.000000	NaN	53.130000	5.000000	NaN	NaN	63770.428010

Check NaN and Null

```
1 data.isnull().sum()
```

```
age         0
sex         0
bmi         0
children    0
smoker      0
region      0
charges     0
dtype: int64
```

```
1 data.isna().sum()
```

```
age         0
sex         0
bmi         0
children    0
smoker      0
region      0
charges     0
dtype: int64
```

Categorical Data Label Encoding

```
#sex
le = LabelEncoder()
le.fit(data.sex.drop_duplicates())
data.sex = le.transform(data.sex)
le_sex_mapping = dict(zip(le.classes_, le.transform(le.classes_)))
print("Gender/Sex label mapping", le_sex_mapping)

# smoker or not
le.fit(data.smoker.drop_duplicates())
data.smoker = le.transform(data.smoker)
le_smoker_mapping = dict(zip(le.classes_, le.transform(le.classes_)))
print("Smoker label mapping", le_smoker_mapping)

#region
le.fit(data.region.drop_duplicates())
data.region = le.transform(data.region)
le_region_mapping = dict(zip(le.classes_, le.transform(le.classes_)))
print("Region label mapping", le_region_mapping)
```

```
Gender/Sex label mapping {'female': 0, 'male': 1}
Smoker label mapping {'no': 0, 'yes': 1}
Region label mapping {'northeast': 0, 'northwest': 1, 'southeast': 2, 'southwest': 3}
```

Correlation

```
1 data.corrwith(data['charges']).sort_values(ascending=False)
```

```
charges     1.000000
smoker      0.787251
age         0.299008
bmi         0.198341
children    0.067998
sex         0.057292
region     -0.006208
dtype: float64
```

A strong correlation is observed only with the fact of smoking patient.

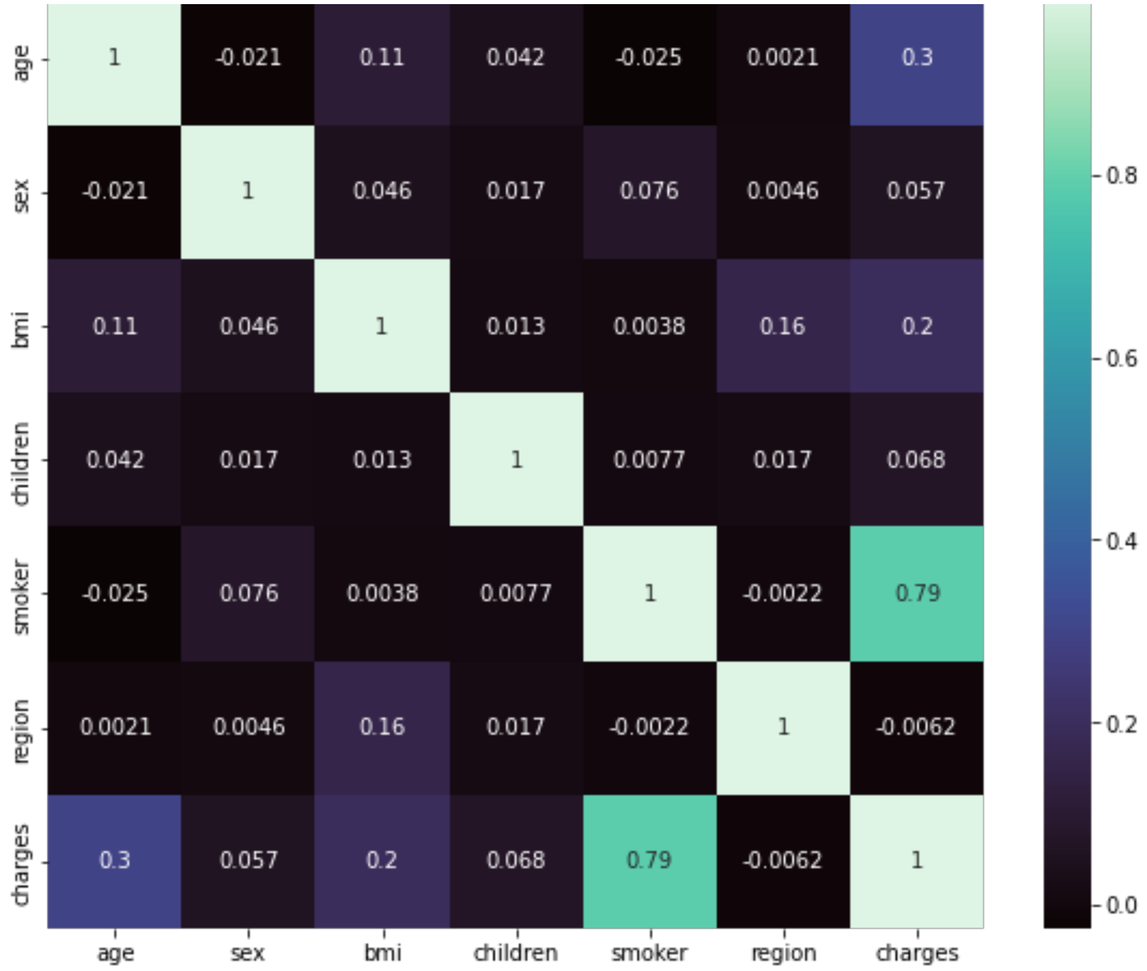
Alternatively can use

```
1 data.corr()['charges'].sort_values(ascending=False)
```

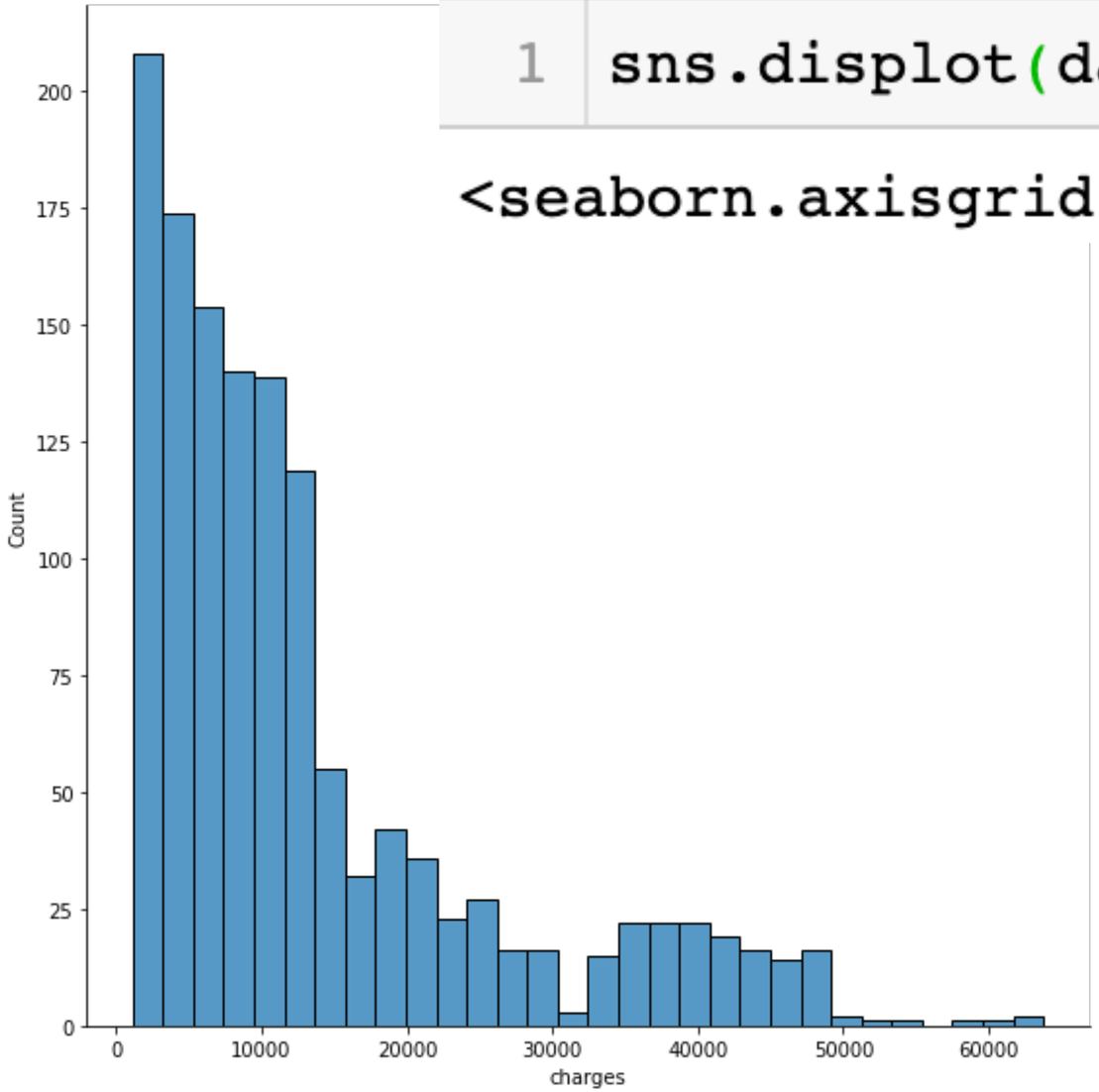
```
charges     1.000000
smoker      0.787251
age         0.299008
bmi         0.198341
children    0.067998
sex         0.057292
region     -0.006208
Name: charges, dtype: float64
```

Heatmap

```
1 f, ax = pl.subplots(figsize=(10, 8))
2 corr = data.corr()
3 sns.heatmap(corr, square=True, ax=ax, annot=True, cmap='mako')
```



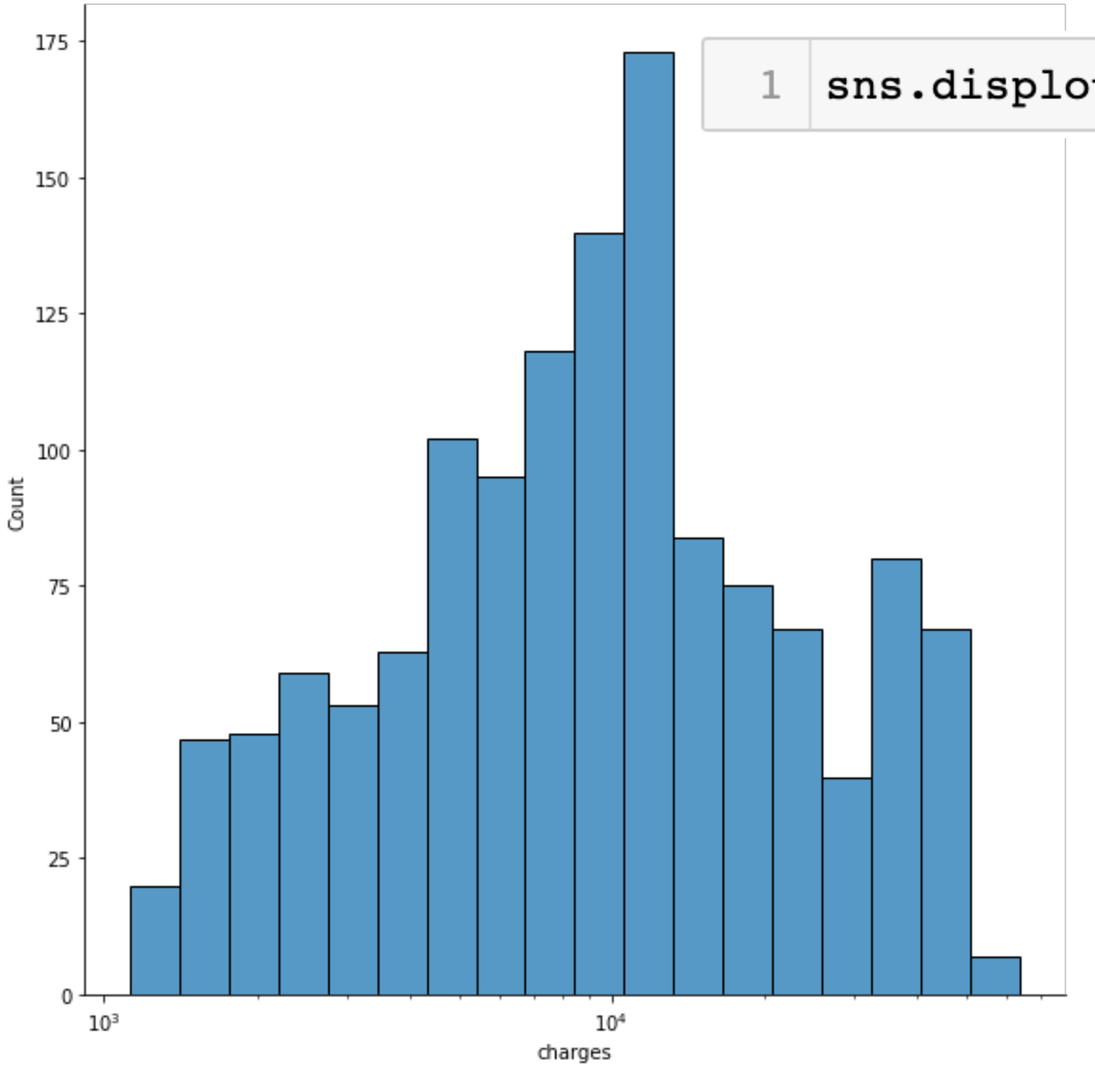
Distribution of charges




```
1  sns.displot(data, x="charges", height=8)
```

```
<seaborn.axisgrid.FacetGrid at 0x1354114c0>
```

Distribution of charges – log scale



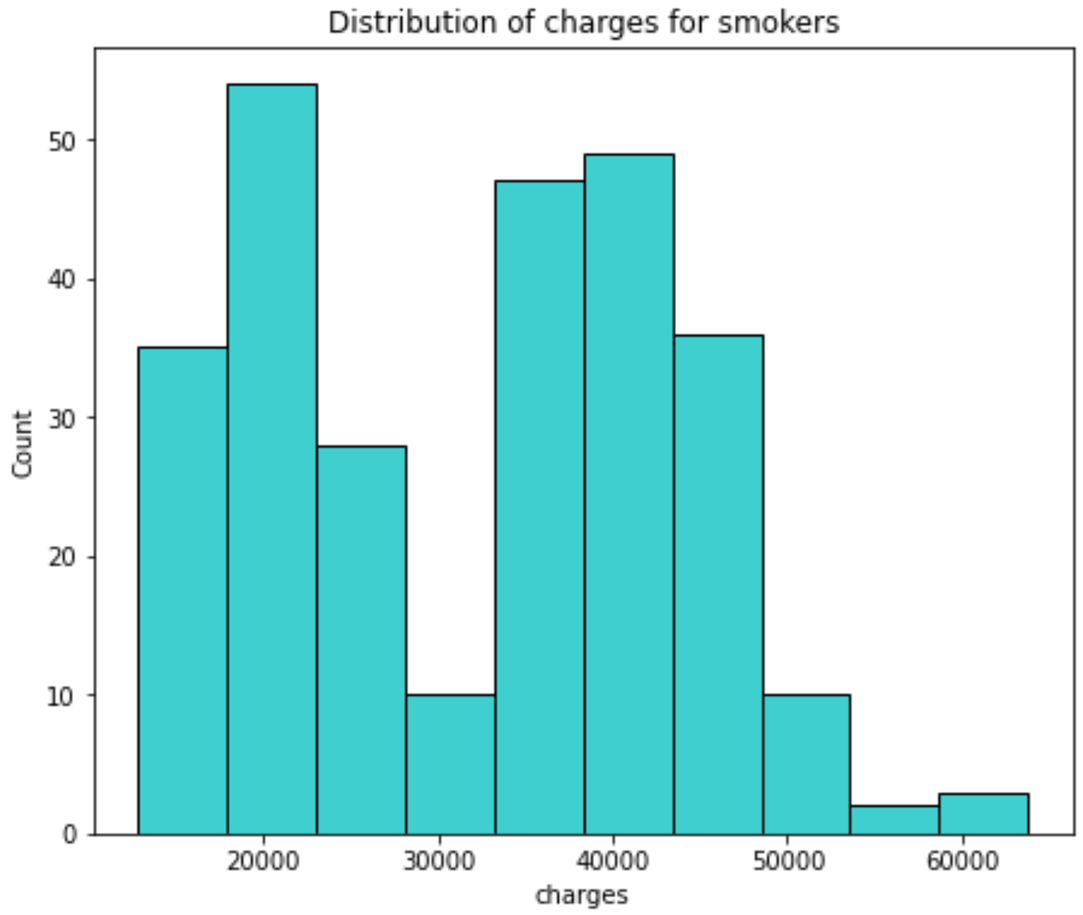

```
1  sns.displot(data, x="charges", height=8, log_scale=True)
```

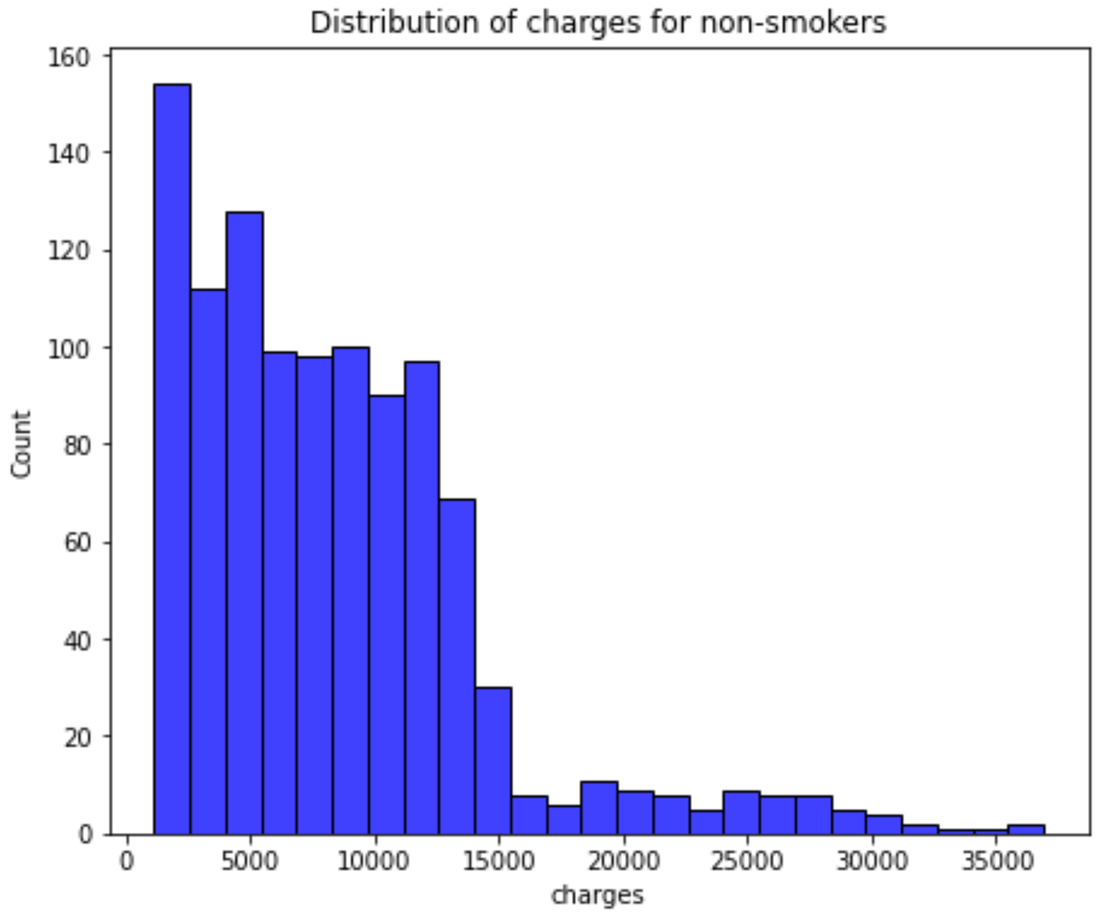
Charge on smoker and non-smoker

```
f= pl.figure(figsize=(16,6))

ax=f.add_subplot(121)
sns.histplot(data[(data.smoker == 1)]["charges"],color='c',ax=ax)
ax.set_title('Distribution of charges for smokers')

ax=f.add_subplot(122)
sns.histplot(data[(data.smoker == 0)]['charges'],color='b',ax=ax)
ax.set_title('Distribution of charges for non-smokers')
```

Charge on smoker and non-smoker

Smoking patients spend more on treatment.

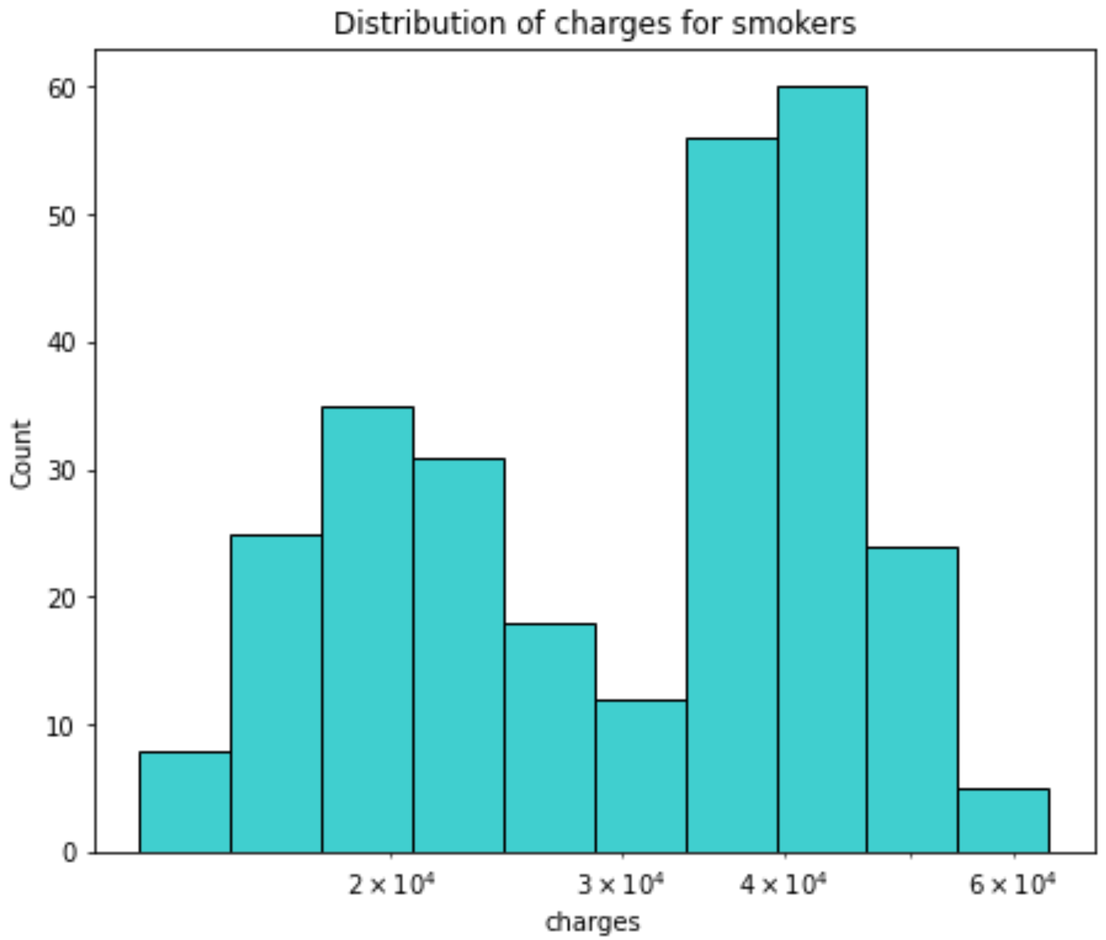


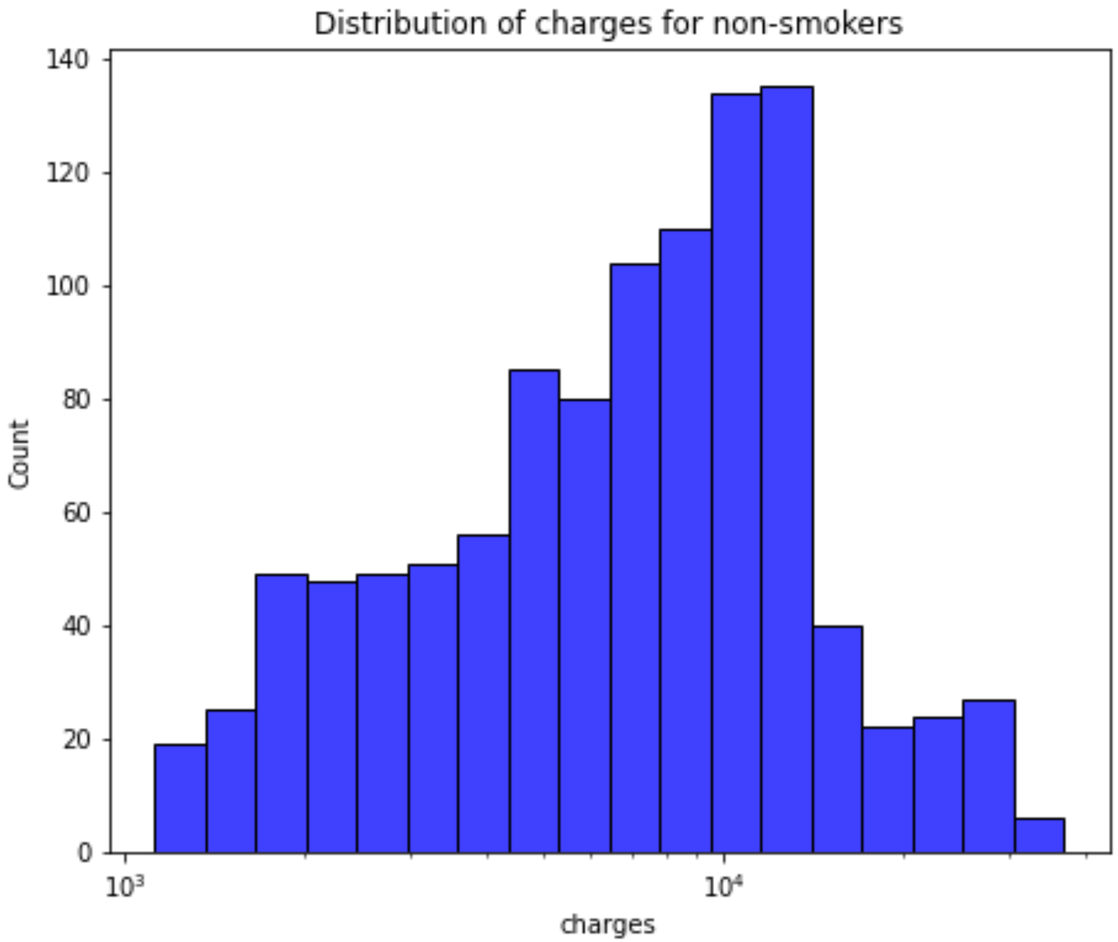


Charge on smoker and non-smoker – log scale

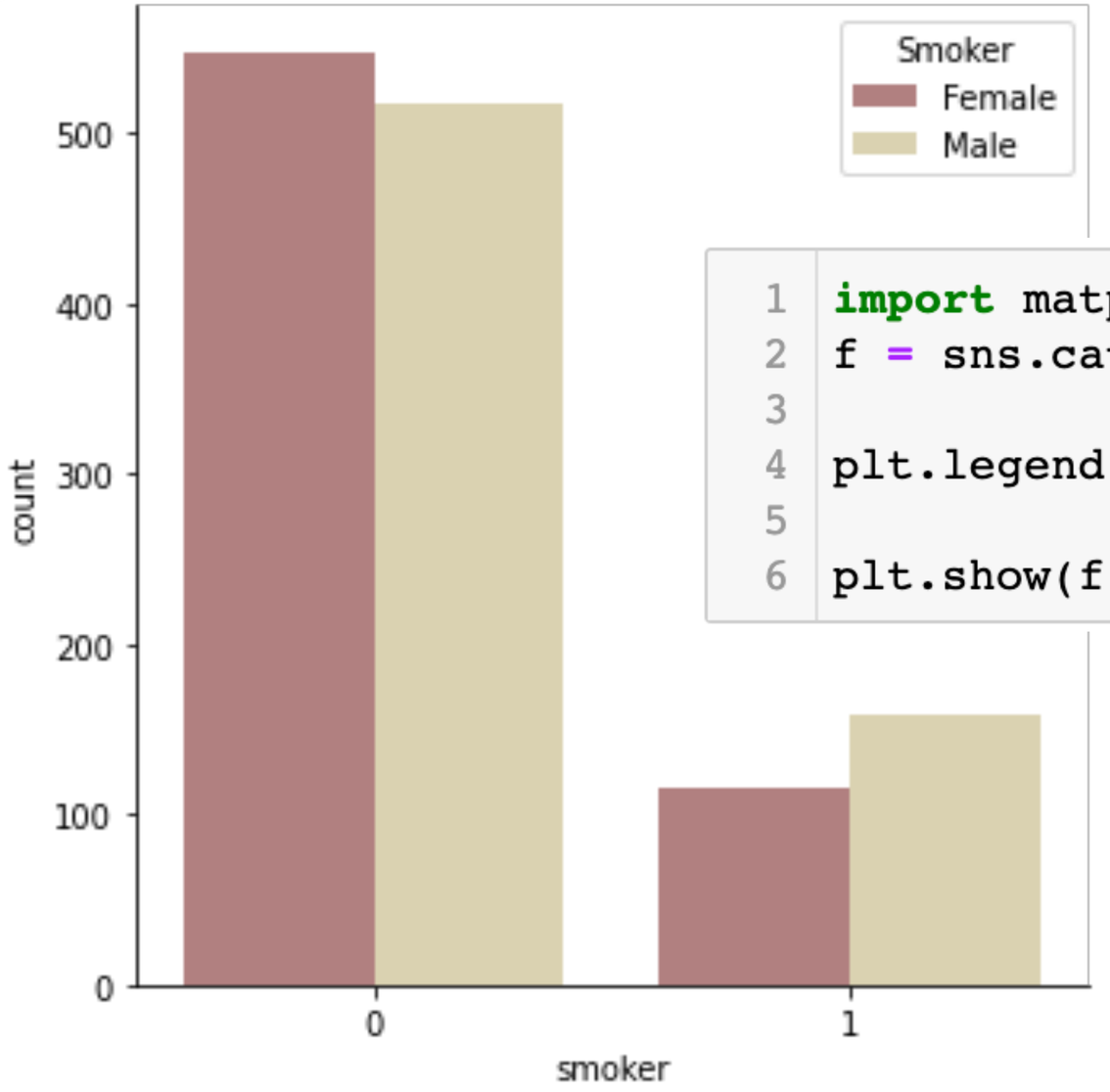
```
# Log-scale
f= pl.figure(figsize=(16,6))

ax=f.add_subplot(121)
sns.histplot(data[(data.smoker == 1)]["charges"],
             color='c', ax=ax, log_scale=True,)
ax.set_title('Distribution of charges for smokers')

ax=f.add_subplot(122)
sns.histplot(data[(data.smoker == 0)]['charges'],
             color='b', ax=ax, log_scale=True,)
ax.set_title('Distribution of charges for non-smokers')
```

Charge on smoker and non-smoker – log scale





Patient gender count



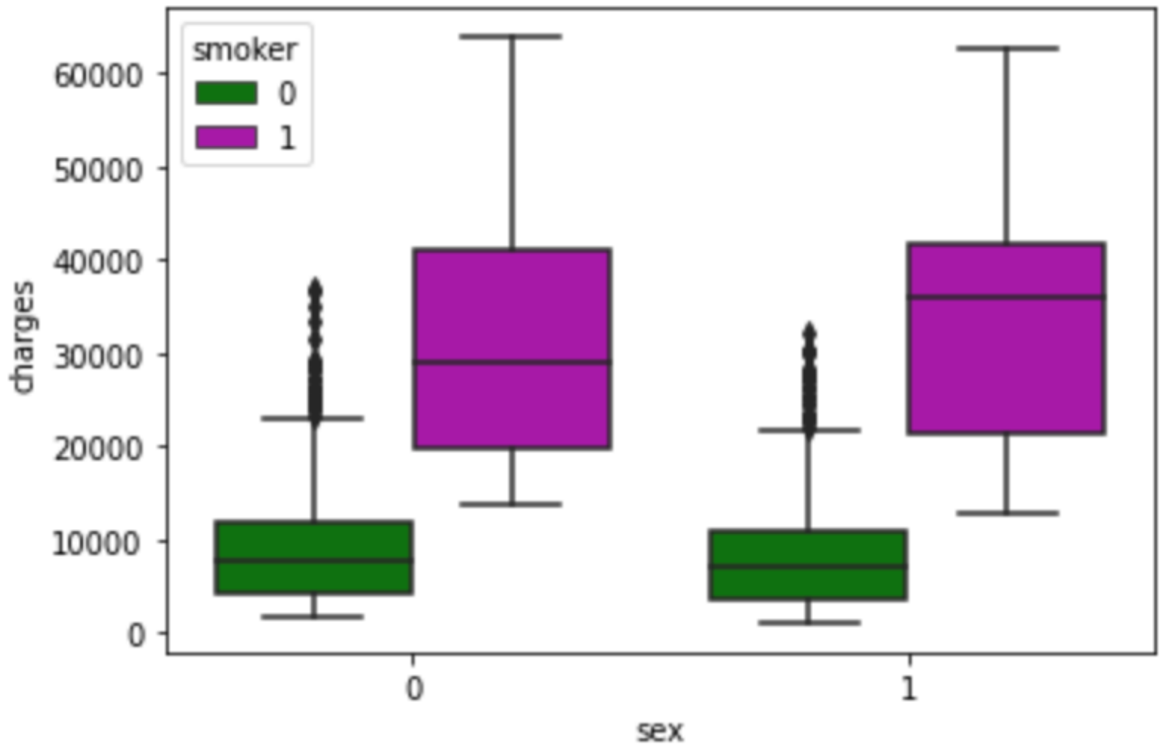

```
1  import matplotlib.pyplot as plt
2  f = sns.catplot(x="smoker", kind="count",hue = 'sex',
3                  palette="pink", data=data, legend=False)
4  plt.legend(title='Smoker', loc='upper right',
5             labels=['Female', 'Male'])
6  plt.show(f)
```

0: non-smoker
1: smoker

Distribution on charge – smoker vs non smoker

```
1  # 0:female 1:male
2  # 0:non-smoker 1:smoker
3  sns.boxplot(x="sex", y="charges", hue="smoker",
4              palette=["g", "m"], data=data,)
```

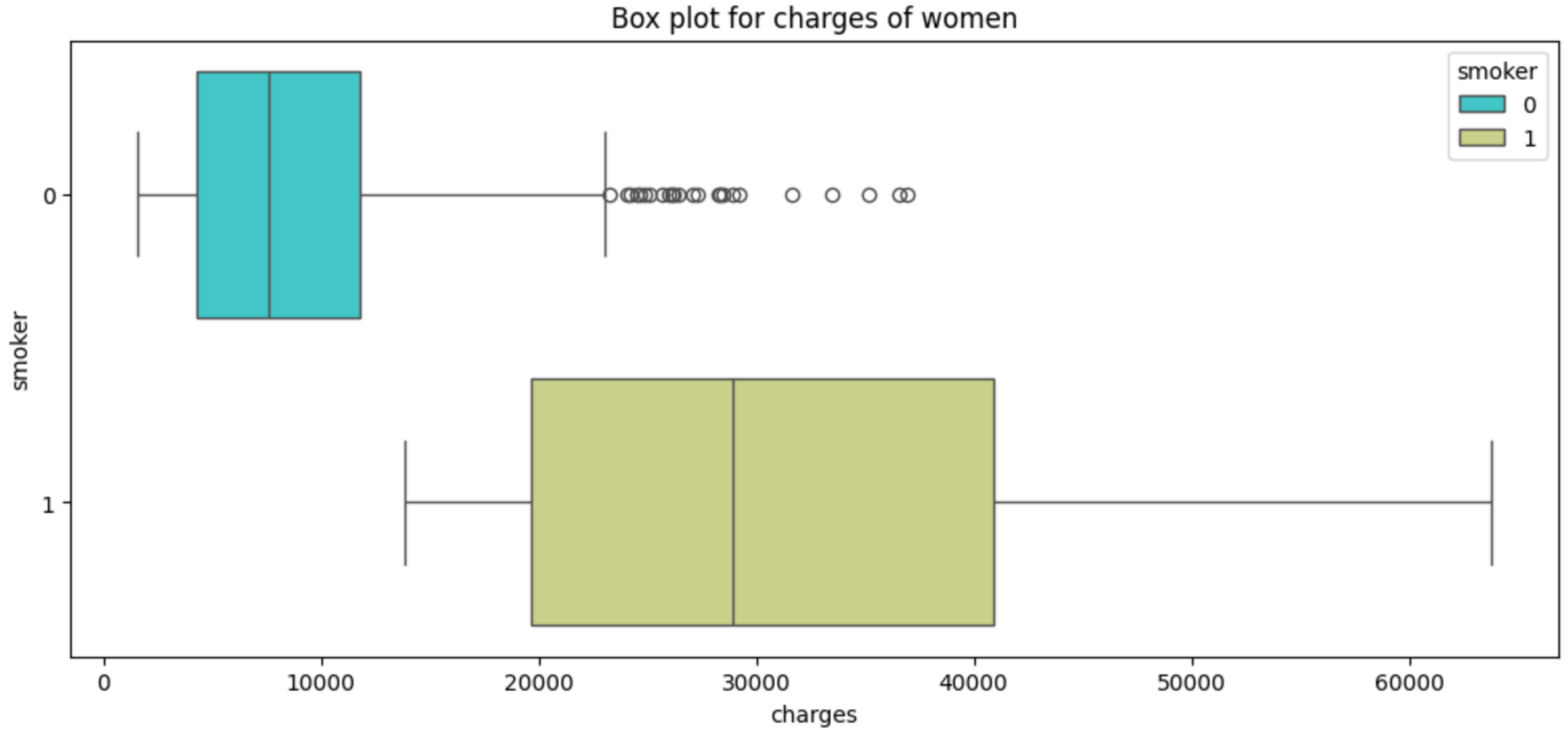
```
<AxesSubplot:xlabel='sex', ylabel='charges'>
```



Box plot for charges of women

```
1 pl.figure(figsize=(12,5))
2 pl.title("Box plot for charges of women")
3 sns.boxplot(y="smoker", x="charges", data =  data[(data.sex == 0)] , orient="h",
4             palette = 'rainbow', hue='smoker' )
```

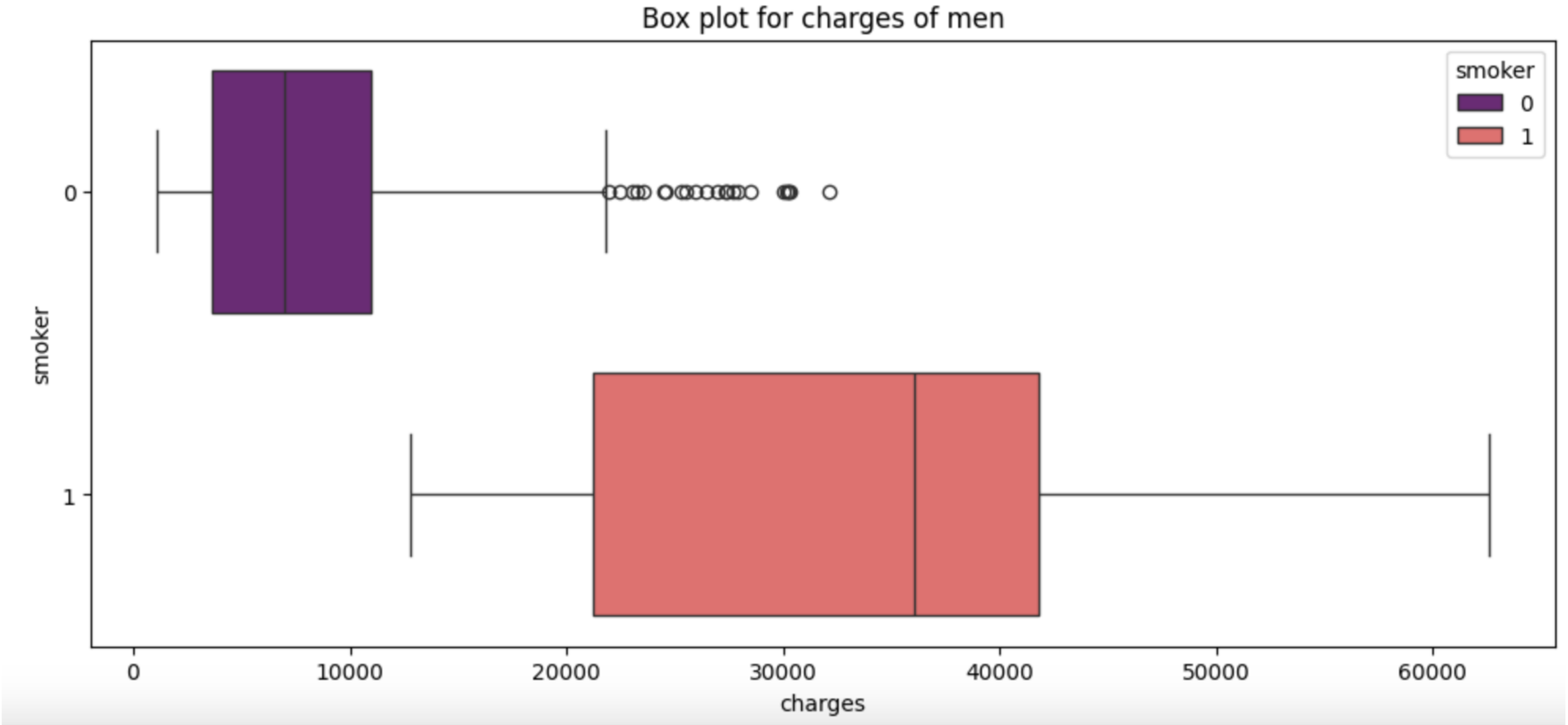
```
<Axes: title={'center': 'Box plot for charges of women'}, xlabel='charges', ylabel='smoker'>
```



Box plot for charges of men

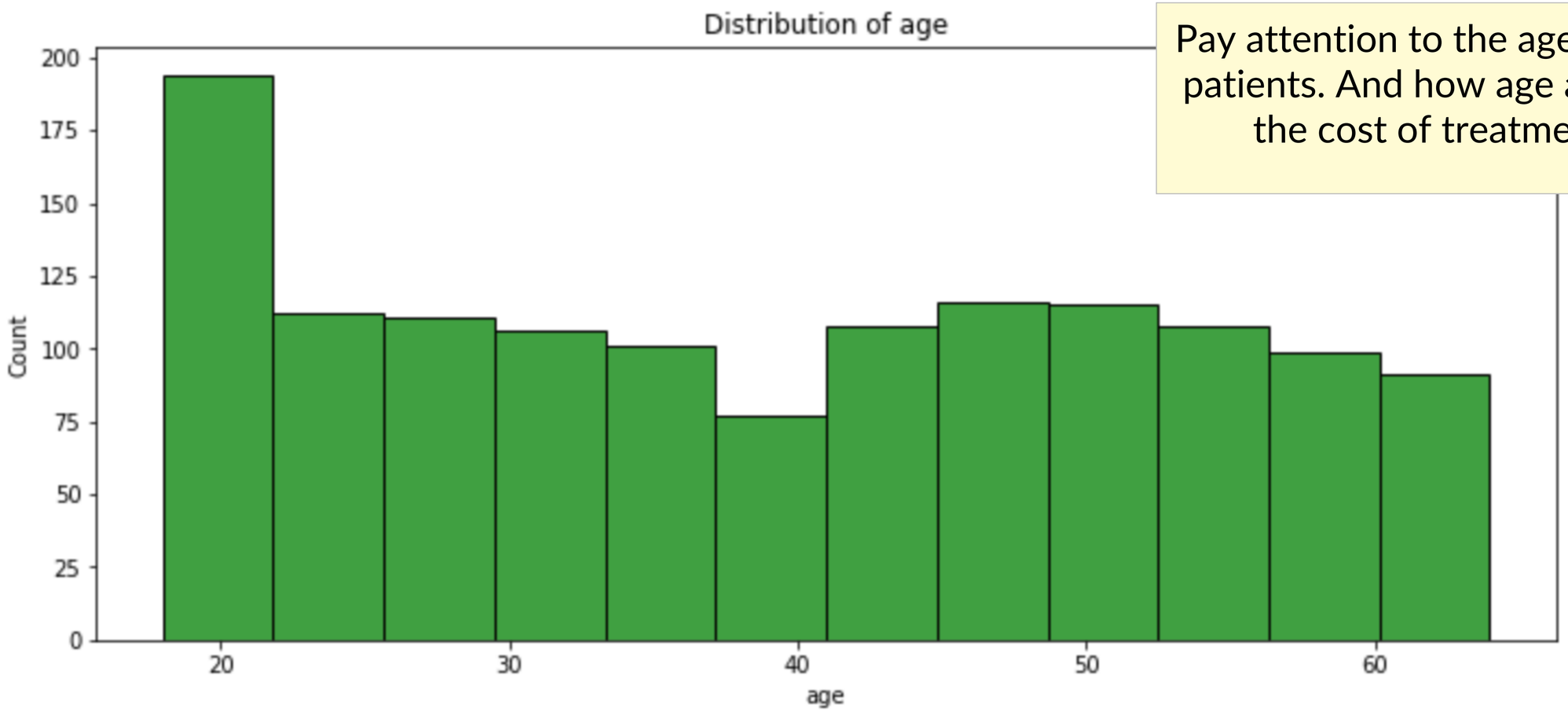
```
1 pl.figure(figsize=(12,5))
2 pl.title("Box plot for charges of men")
3 sns.boxplot(y="smoker", x="charges", data =  data[(data.sex == 1)] ,
4             orient="h", palette = 'magma', hue="smoker")
```

```
<Axes: title={'center': 'Box plot for charges of men'}, xlabel='charges', ylabel='smoker'>
```



Distribution on age

```
1 pl.figure(figsize=(12,5))
2 pl.title("Distribution of age")
3 ax = sns.histplot(data["age"], color = 'g')
```

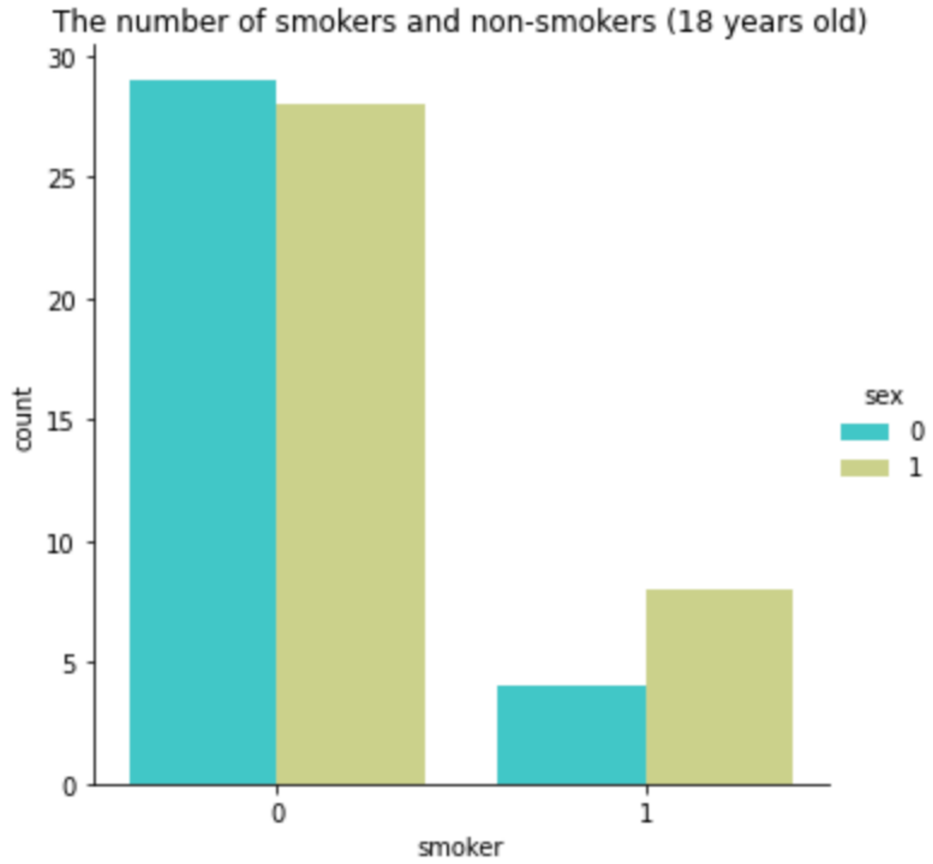


Pay attention to the age of the patients. And how age affects the cost of treatment

The number of smokers and non-smokers (18 years old)

```
1 sns.catplot(x="smoker", kind="count",hue = 'sex', palette="rainbow",
2              data=data[(data.age == 18)])
3 pl.title("The number of smokers and non-smokers (18 years old)")
```

```
Text(0.5, 1.0, 'The number of smokers and non-smokers (18 years old)')
```

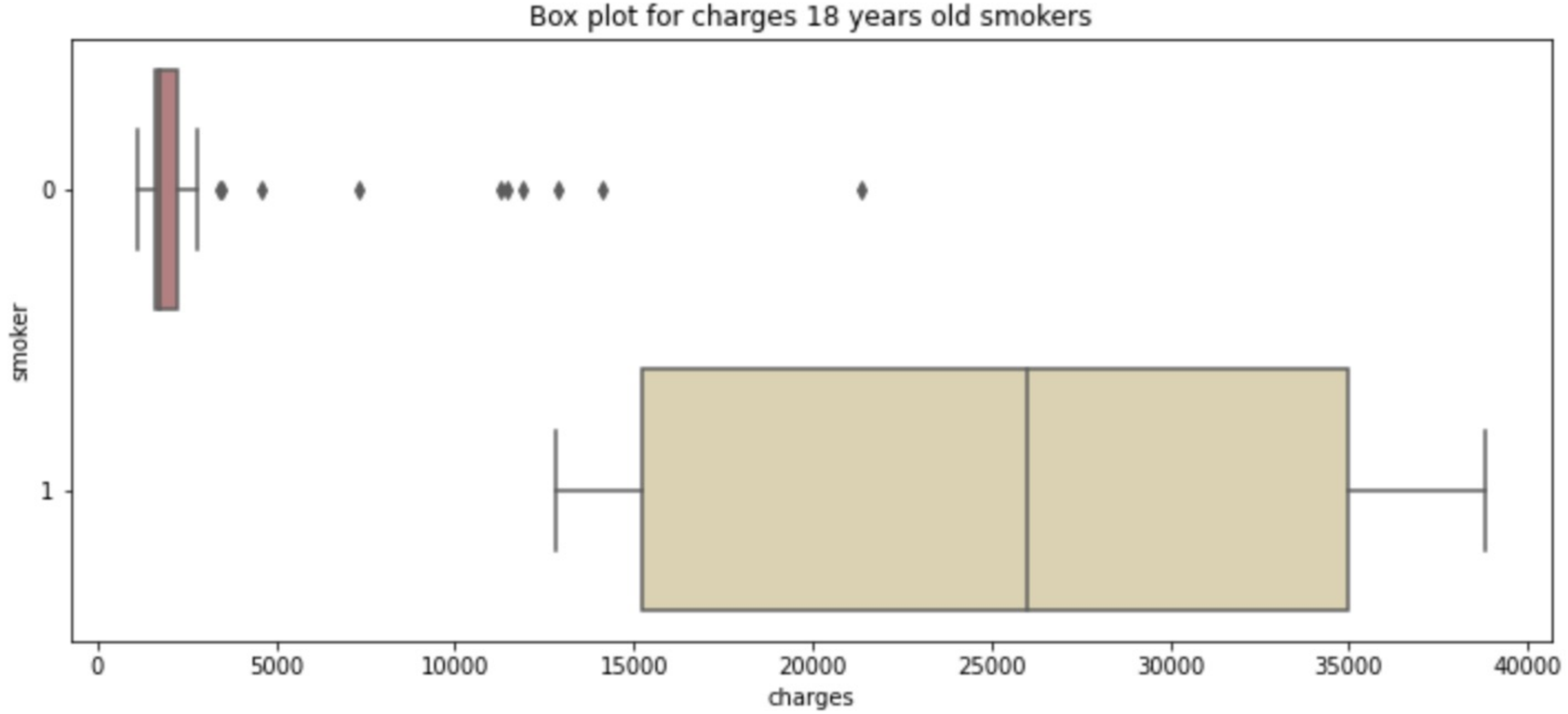


0: female, 1: male
0: non-smoker, 1: smoker

Smoking affect the cost of treatment at Age 18

```
1 pl.figure(figsize=(12,5))
2 pl.title("Box plot for charges 18 years old smokers")
3 sns.boxplot(y="smoker", x="charges", data = data[(data.age == 18)] , orient="h", palette = 'pink')
```

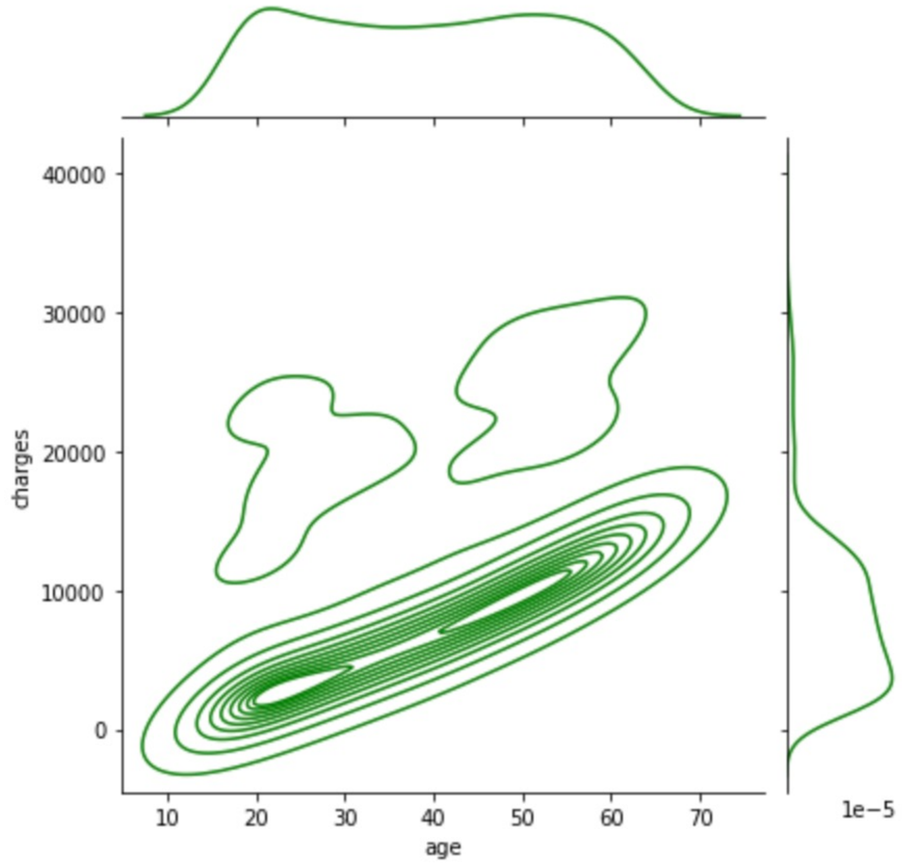
```
<AxesSubplot:title={'center':'Box plot for charges 18 years old smokers'}, xlabel='charges', ylabel='smoker'>
```



Distribution of charges and age for non-smokers

```
1  g = sns.jointplot(
2      data=data[(data.smoker == 0)],  x="age", y="charges",
3      kind="kde", color="g")
4  ax.set_title('Distribution of charges and age for non-smokers')
```

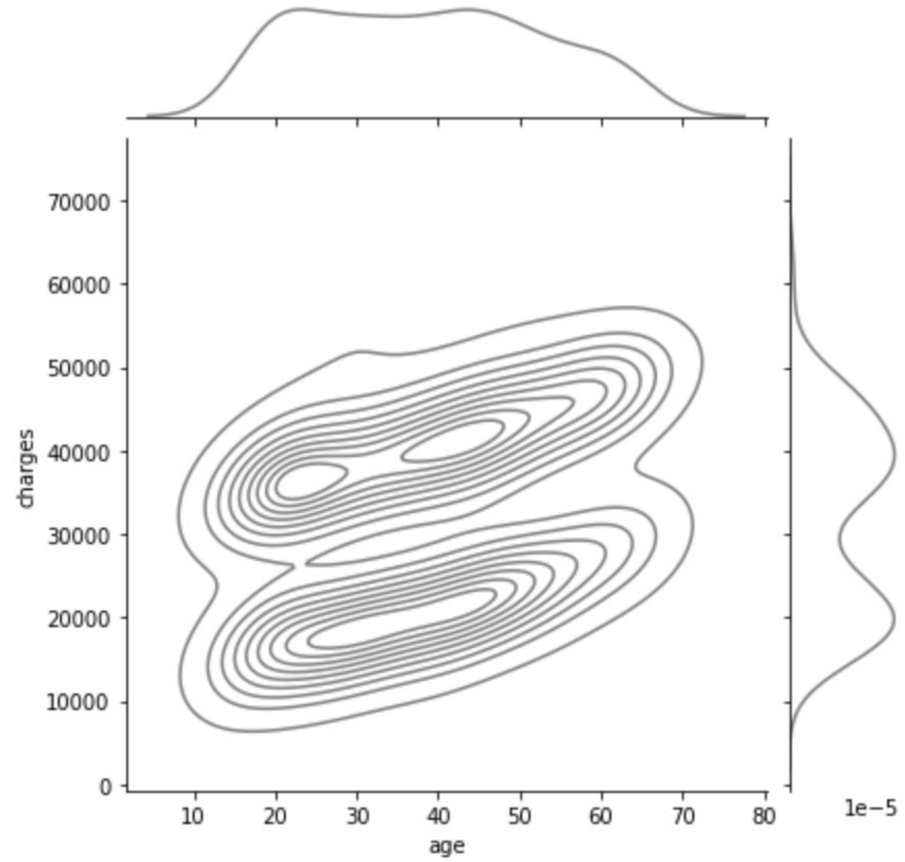
```
Text(0.5, 1.0, 'Distribution of charges and age for non-smokers')
```



Distribution of charges and age for smokers

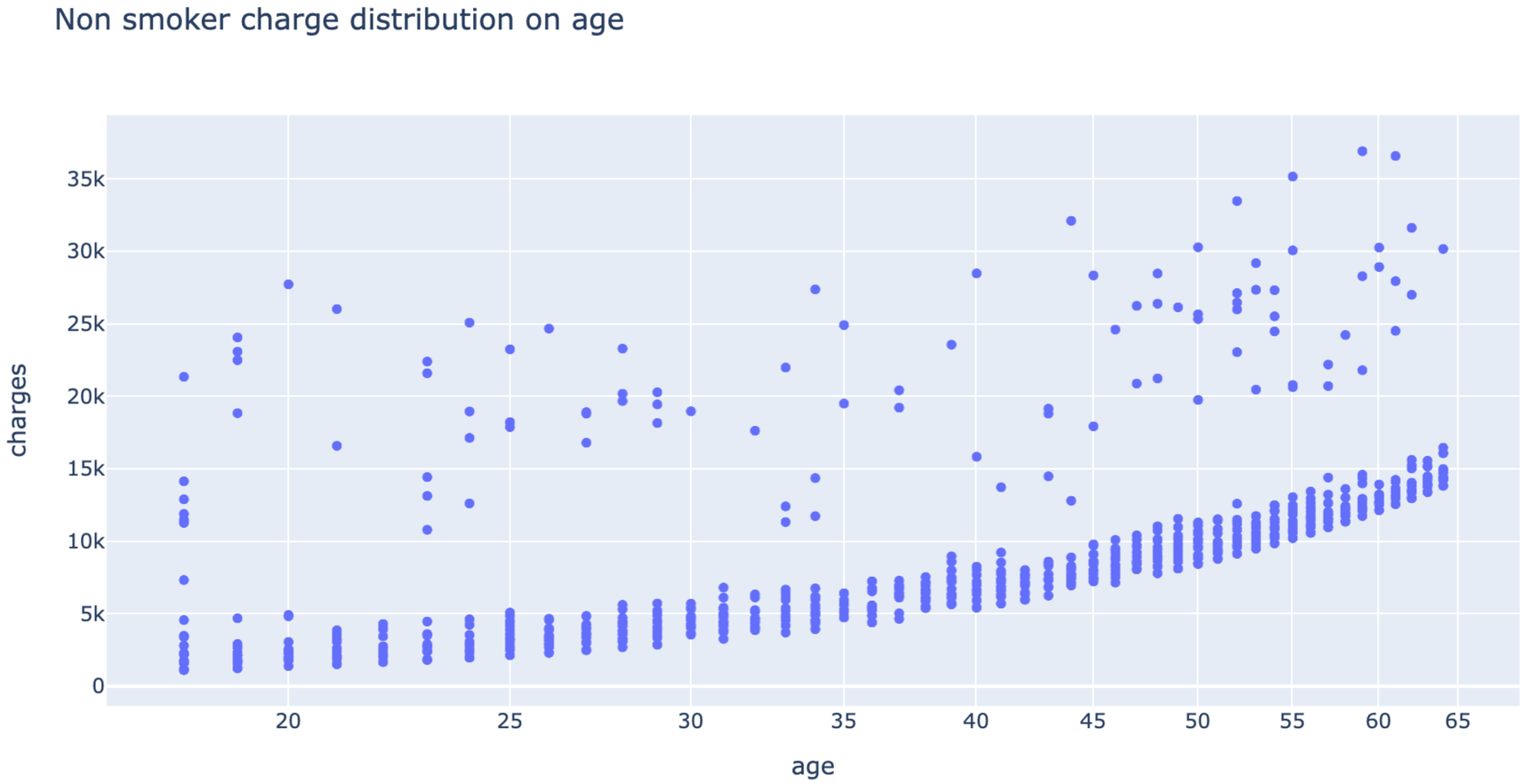
```
g = sns.jointplot(
    data=data[(data.smoker == 1)],  x="age", y="charges",
    kind="kde", color="grey")
ax.set_title('Distribution of charges and age for smokers')
```

```
Text(0.5, 1.0, 'Distribution of charges and age for smokers')
```



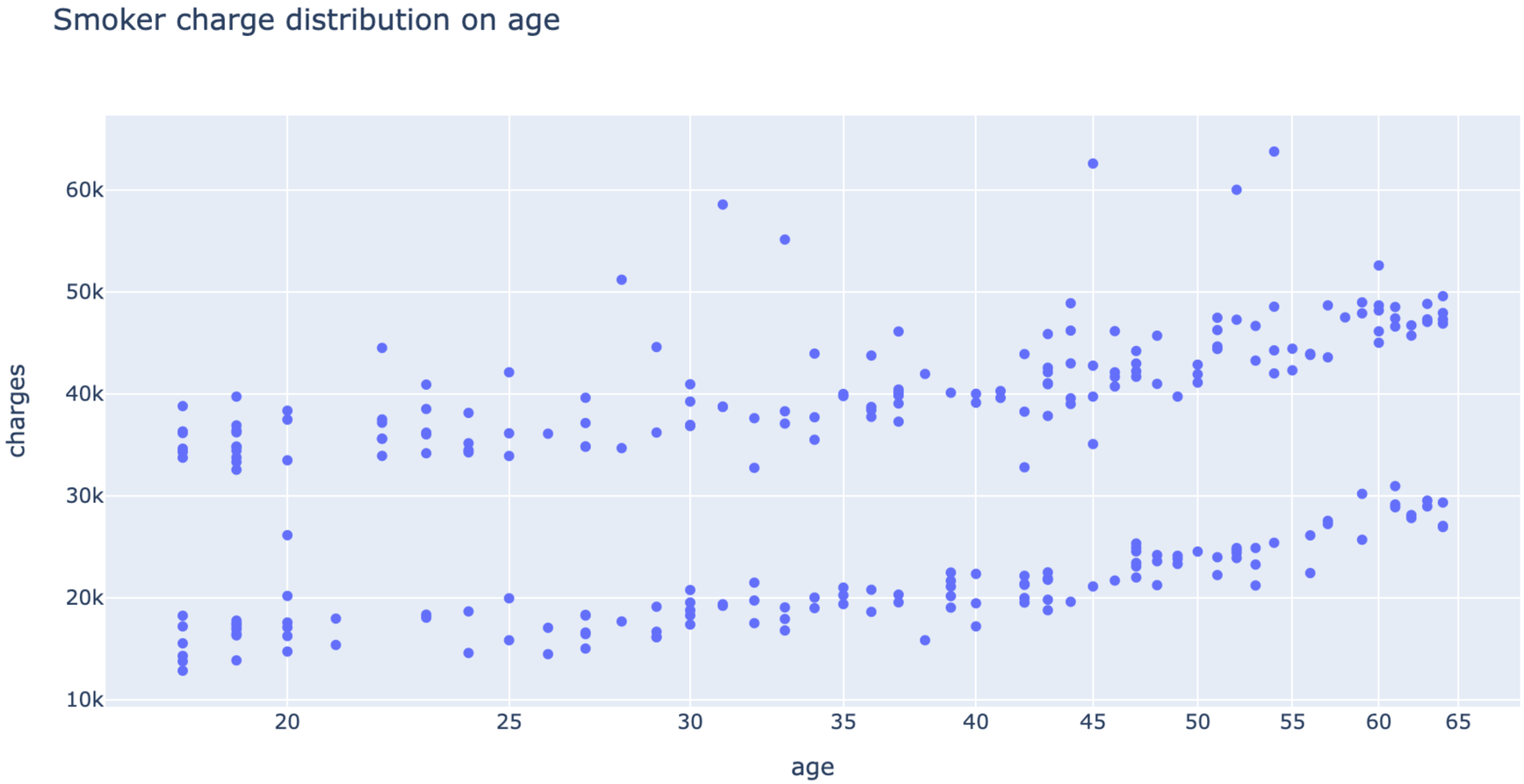
Non smoker charge distribution on age

```
1 import plotly.express as px
2
3 fig = px.scatter(data[(data.smoker == 0)], x="age", y="charges",
4                  log_x=True, size_max=60, title='Non smoker charge distribution on age')
5 fig.show()
```



Smoker charge distribution on age

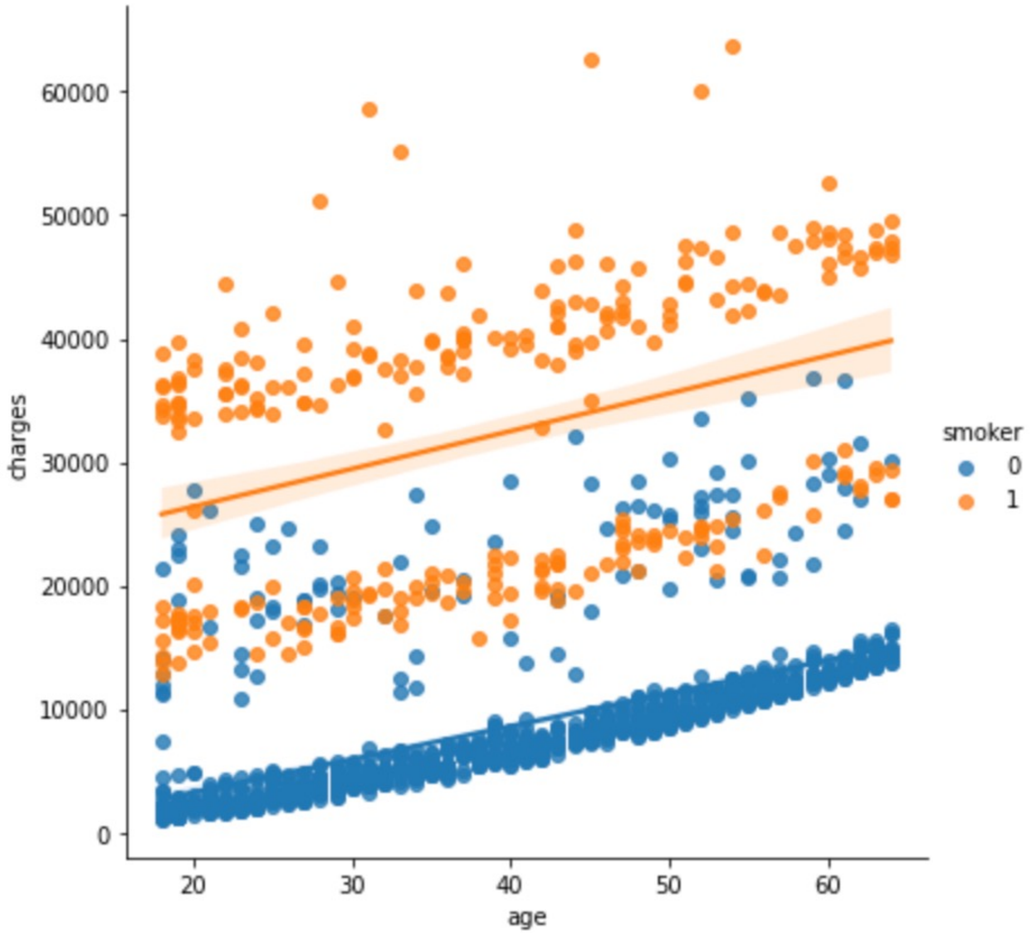
```
fig = px.scatter(data[(data.smoker == 1)], x="age", y="charges",
                 log_x=True, size_max=60,
                 title='Smoker charge distribution on age')
fig.show()
```



Smokers and non-smokers on age

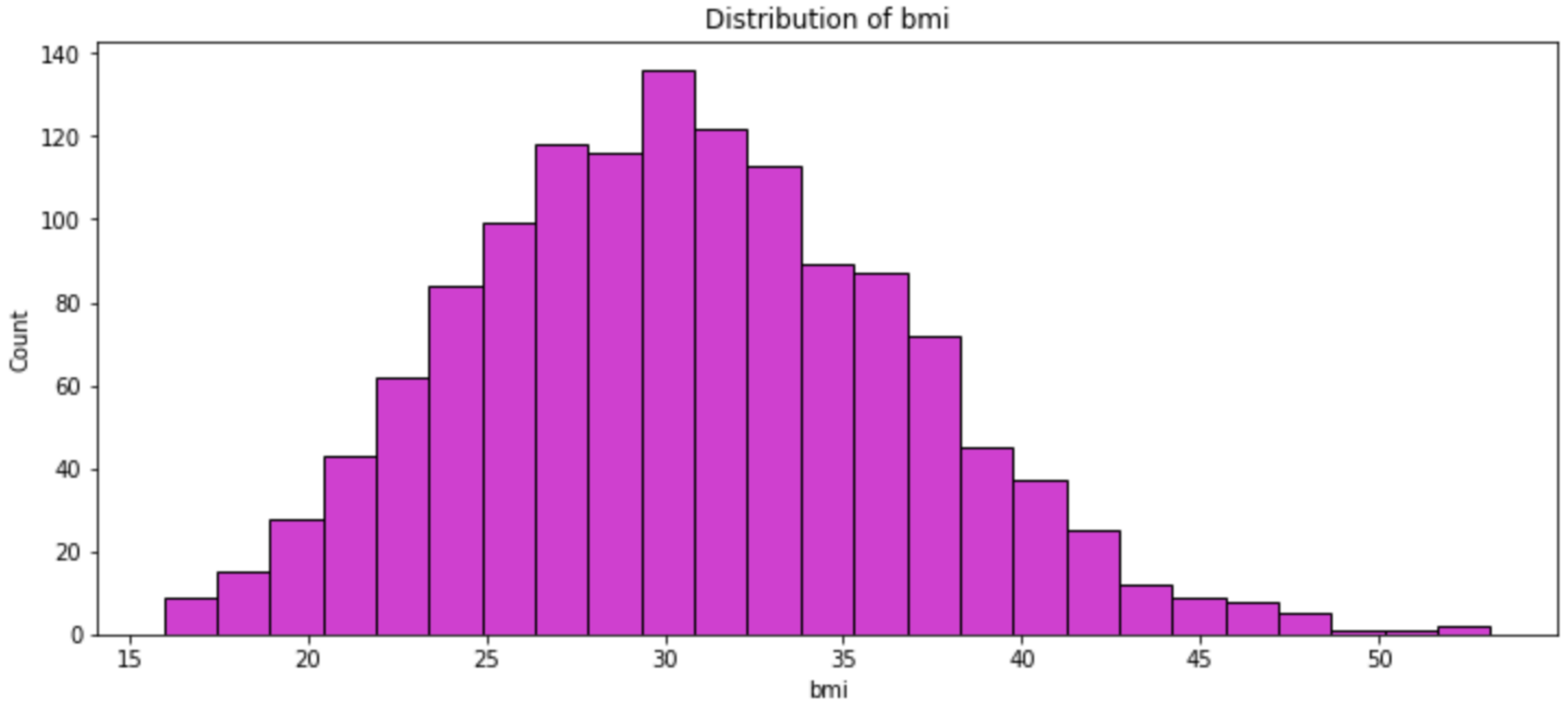
```
1  sns.lmplot(x="age", y="charges", hue="smoker", data=data, height=6 )
2  ax.set_title('Smokers and non-smokers')
```

```
Text(0.5, 1.0, 'Smokers and non-smokers')
```



Distribution of bmi

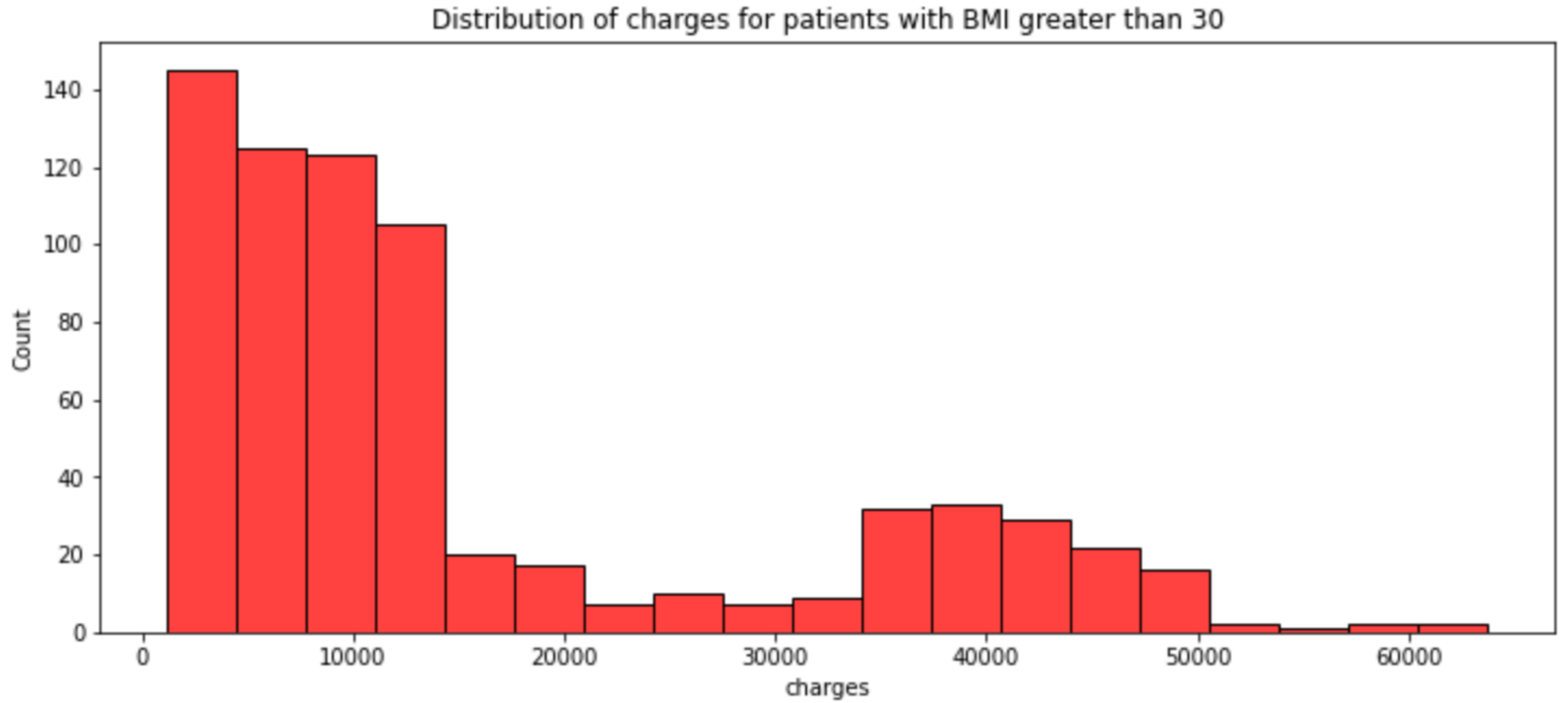
```
1 pl.figure(figsize=(12,5))
2 pl.title("Distribution of bmi")
3 ax = sns.histplot(data["bmi"], color = 'm')
```



Distribution of charges for patients with BMI greater than 30

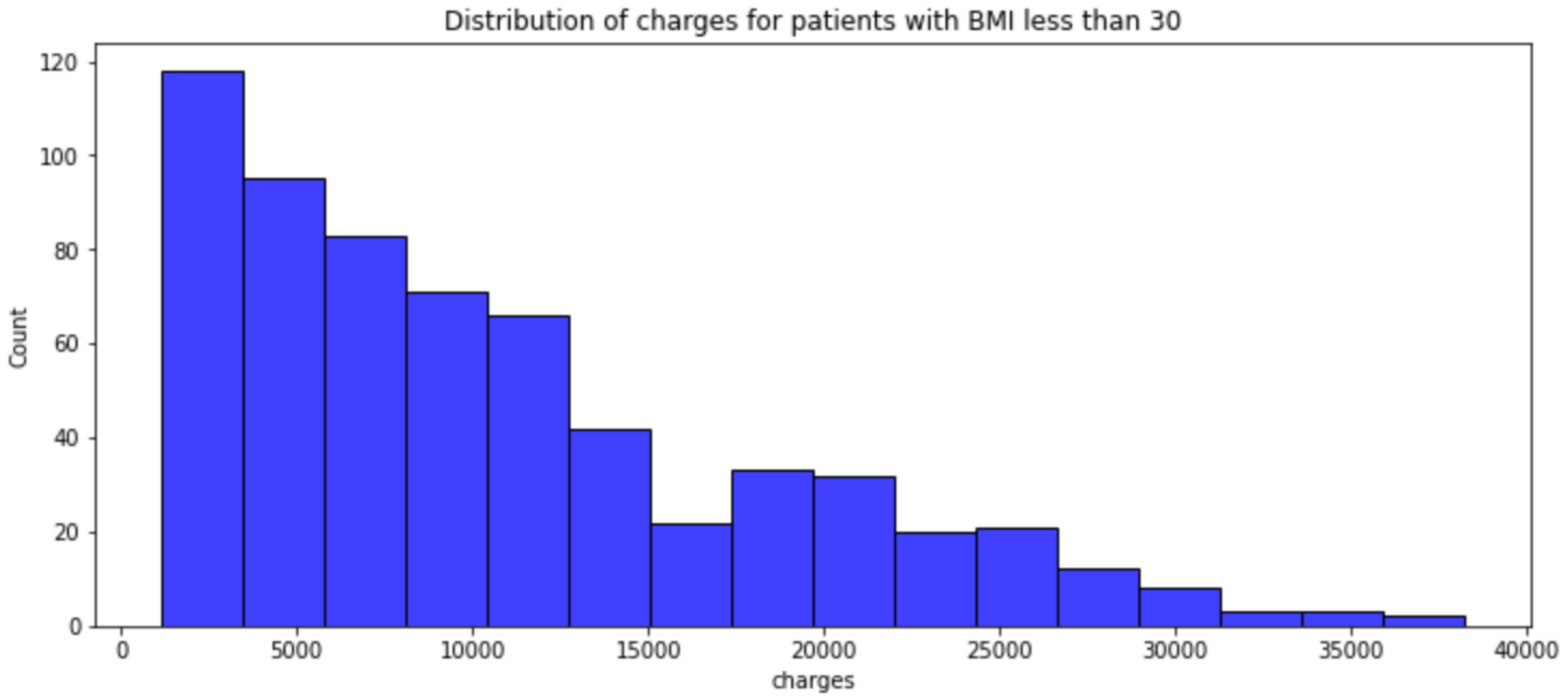
With a value equal to 30 starts obesity. We also calculated my BMI and now we can safely eat a sandwich. Let's start to explore! First, let's look at the distribution of costs in patients with BMI greater than 30 and less than 30.

```
1  pl.figure(figsize=(12,5))
2  pl.title("Distribution of charges for patients with BMI greater than 30")
3  ax = sns.histplot(data[(data.bmi >= 30)]['charges'], color = 'r')
```



Distribution of charges for patients with BMI less than 30

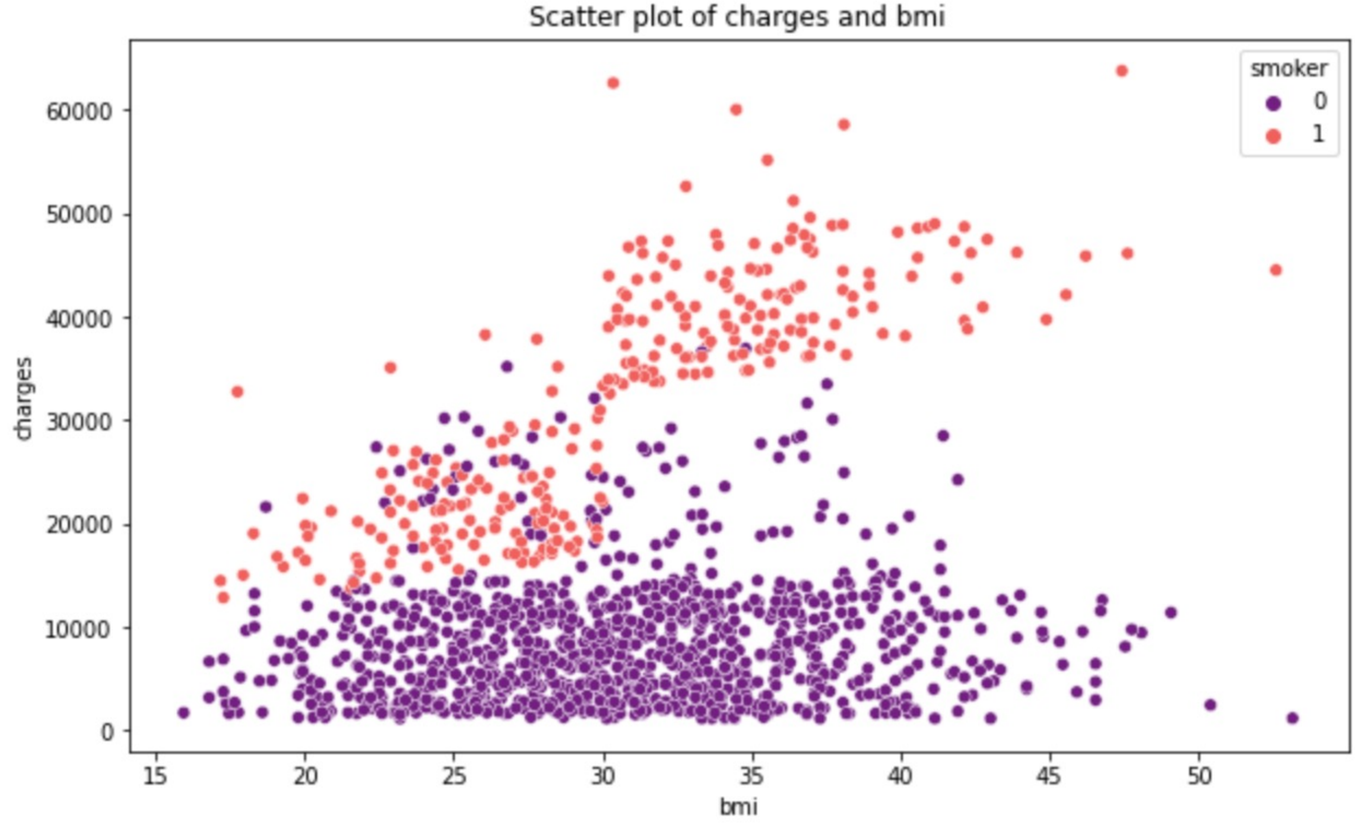
```
1 pl.figure(figsize=(12,5))
2 pl.title("Distribution of charges for patients with BMI less than 30")
3 ax = sns.histplot(data[(data.bmi < 30)]['charges'], color = 'b')
```

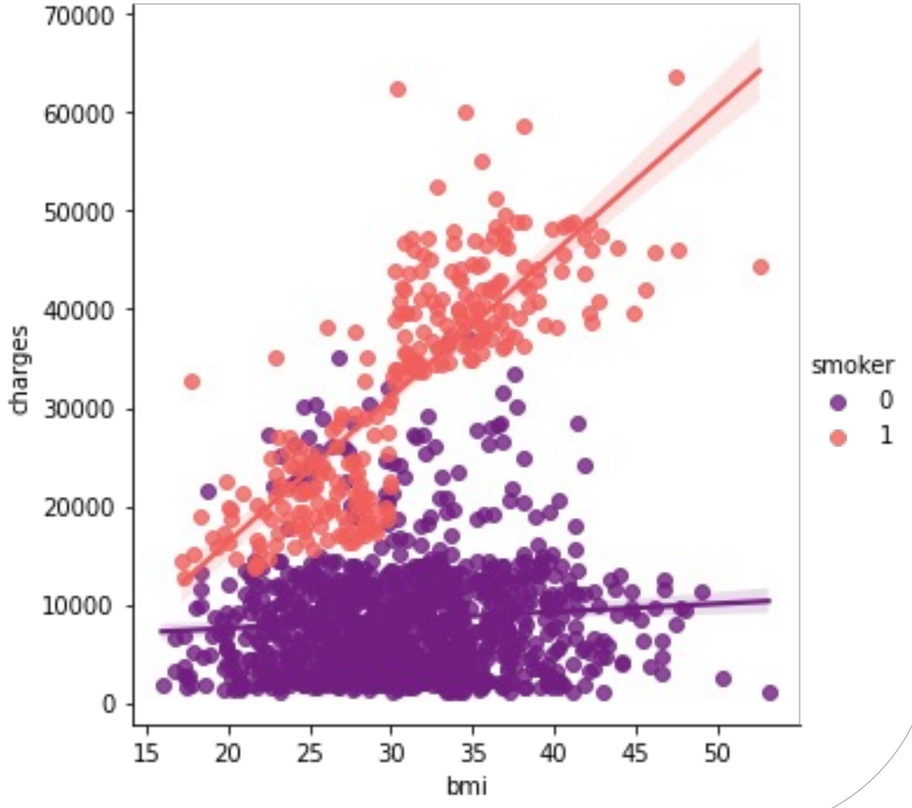


Scatter plot of charges and bmi

```
1  pl.figure(figsize=(10,6))
2  ax = sns.scatterplot(x='bmi',y='charges',data=data,palette='magma',hue='smoker')
3  ax.set_title('Scatter plot of charges and bmi')
4
5  sns.lmplot(x="bmi", y="charges", hue="smoker", data=data, palette = 'magma')
```

```
<seaborn.axisgrid.FacetGrid at 0x17e8b9ca0>
```

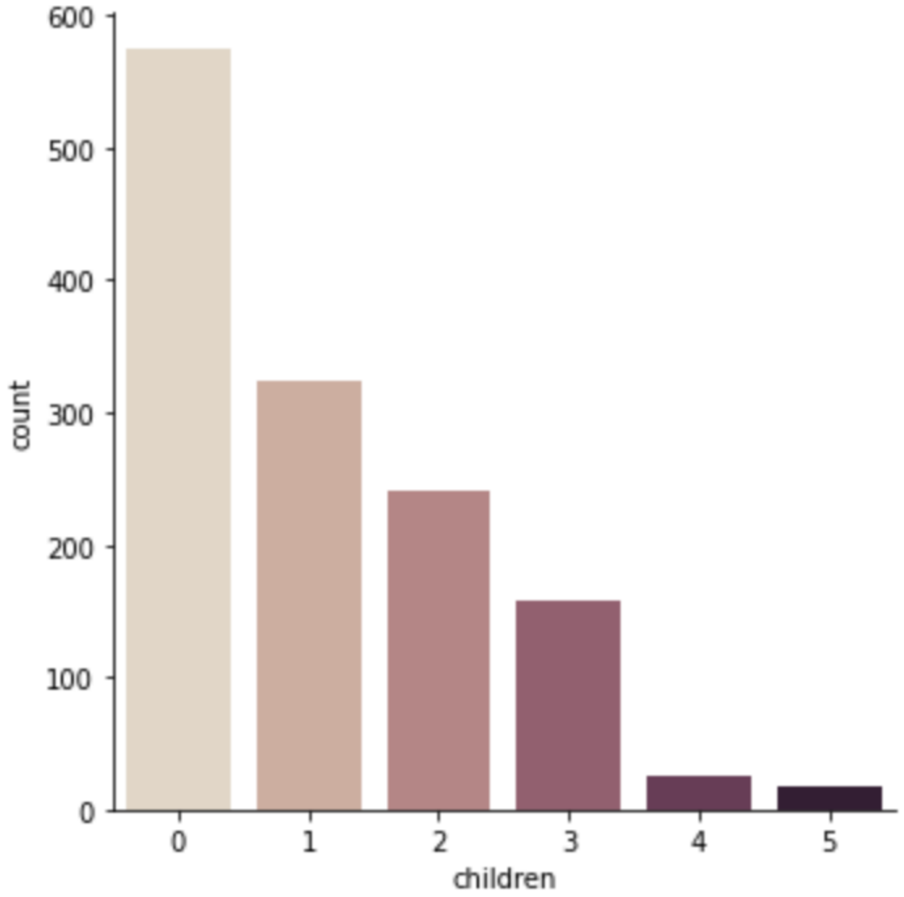




Child patient count with how many sibling

```
1  sns.catplot(x="children", kind="count", palette="ch:.25", data=data, )
```

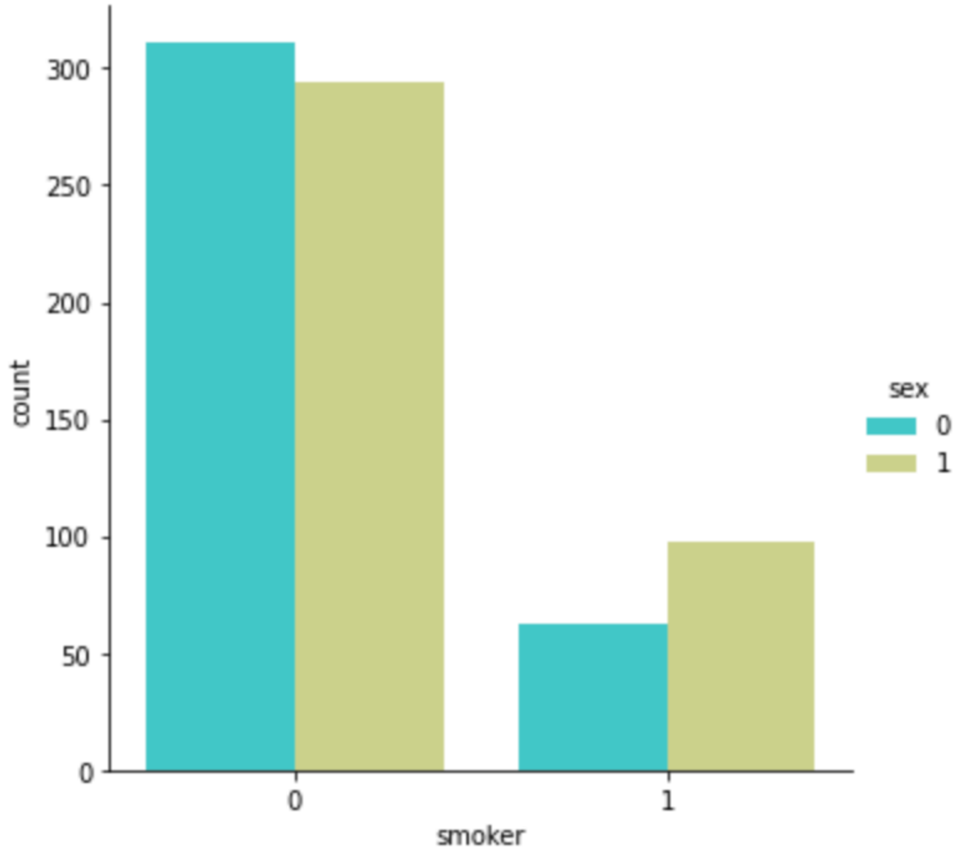
```
<seaborn.axisgrid.FacetGrid at 0x17e904400>
```



Smokers and non-smokers who have childrens

```
1  sns.catplot(x="smoker", kind="count", palette="rainbow",hue = "sex",
2              data=data[(data.children > 0)])
3  ax.set_title('Smokers and non-smokers who have childrens')
```

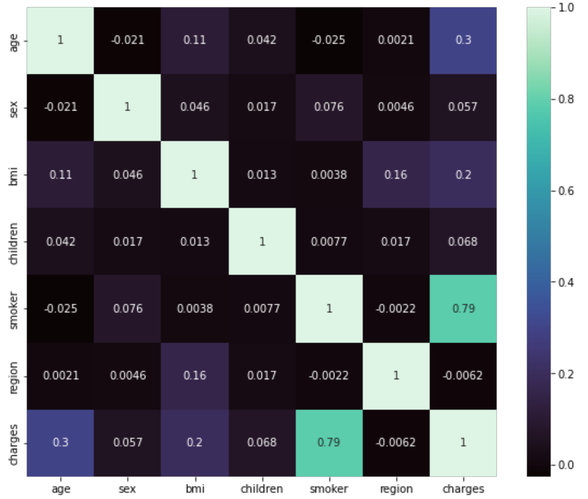
```
Text(0.5, 1.0, 'Smokers and non-smokers who have childrens')
```



Chapter Wrap UP

From all the above figure and chart, we could interpret a lot of ideas.

The cost of treatment depends on many factors: diagnosis, type of clinic, city of residence, age and so on. We have no data on the diagnosis of patients. But we have other information that can help us to make a conclusion about the health of patients and practice regression analysis.



Reference & Resources

Sklearn Website:

https://scikit-learn.org/

Plotly Graph Objects:

https://plotly.com/python/graph-objects/

Seaborn:

https://seaborn.pydata.org/examples/index.html

Matplotlib:

https://matplotlib.org/