In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import scipy.stats as stats
import statsmodels.api as sm
import statsmodels.formula.api as smf


# Load the dataset
insurance_df = pd.read_csv("/Users/balakrishnamupparaju/Downloads/insurance.


# Inspect the dataset
print(insurance_df.info())
print(insurance_df.head())
```

```
<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   bmi       1338 non-null   float64
 3   children  1338 non-null   int64  
 4   smoker    1338 non-null   object 
 5   region    1338 non-null   object 
 6   charges   1338 non-null   float64
dtypes: float64(2), int64(2), object(3)
memory usage: 73.3+ KB
None
   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
```

In [3]:

```
# Check for missing values
print(insurance_df.isnull().sum())

# Data types of variables
print(insurance_df.dtypes)
```

```
age         0
sex         0
bmi         0
children    0
smoker      0
region      0
charges     0
dtype: int64
age           int64
sex          object
bmi         float64
children      int64
smoker       object
region       object
charges     float64
dtype: object
```

In [5]:
```python
print(insurance_df.columns)
```

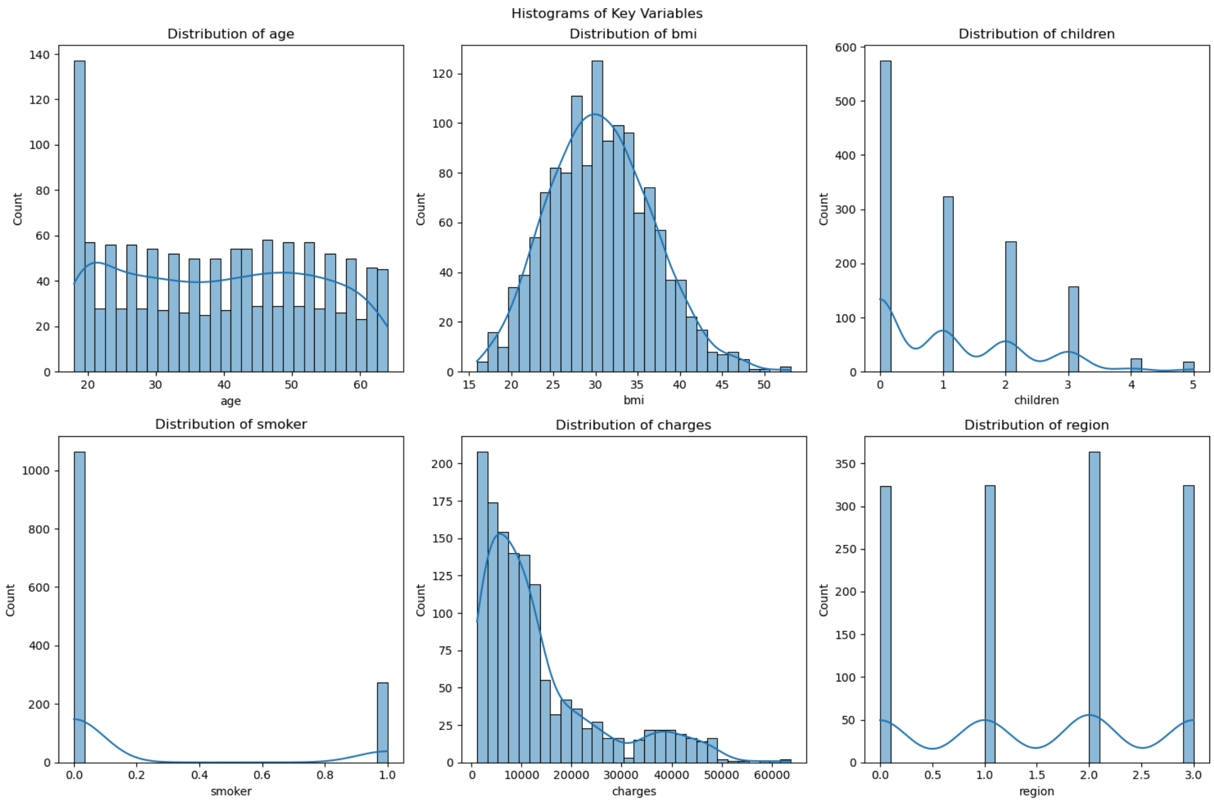
```
Index(['age', 'sex', 'bmi', 'children', 'smoker', 'region', 'charges'], dtyp
e='object')
```

In [9]:
```python
# Convert categorical variables
insurance_df['smoker'] = insurance_df['smoker'].map({'yes': 1, 'no': 0})
insurance_df['sex'] = insurance_df['sex'].map({'male': 1, 'female': 0})  # M
insurance_df['region'] = insurance_df['region'].astype('category').cat.codes
```

In [13]:
```python
# Plot histograms for key variables
fig, axes = plt.subplots(2, 3, figsize=(15, 10))
fig.suptitle('Histograms of Key Variables')

columns = ['age', 'bmi', 'children', 'smoker', 'charges','region']
for i, col in enumerate(columns):
    sns.histplot(insurance_df[col], bins=30, kde=True, ax=axes[i//3, i%3])
    axes[i//3, i%3].set_title(f'Distribution of {col}')

plt.tight_layout()
plt.show()
```



In [15]:
```python
# Summary statistics
insurance_df.describe()
```

Out[15]:

	age	sex	bmi	children	smoker	regi
count	1338.000000	1338.000000	1338.000000	1338.000000	1338.000000	1338.0000
mean	39.207025	0.505232	30.663397	1.094918	0.204783	1.5156
std	14.049960	0.500160	6.098187	1.205493	0.403694	1.1048
min	18.000000	0.000000	15.960000	0.000000	0.000000	0.0000
25%	27.000000	0.000000	26.296250	0.000000	0.000000	1.0000
50%	39.000000	1.000000	30.400000	1.000000	0.000000	2.0000
75%	51.000000	1.000000	34.693750	2.000000	0.000000	2.0000
max	64.000000	1.000000	53.130000	5.000000	1.000000	3.0000

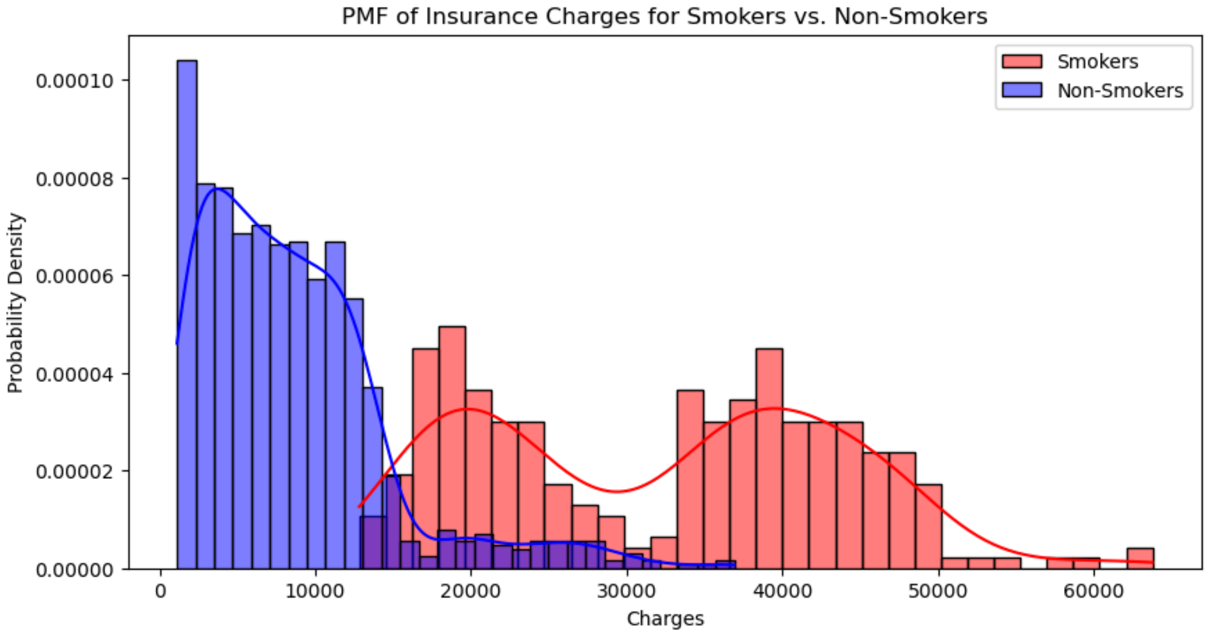
In [ ]:
```python
#Look at mean, median, and standard deviation.
#Identify potential outliers in BMI and charges.
C#heck the skewness of charges (often right-skewed).
```

In [19]:
```python
# PMF for smokers vs. non-smokers (charges)
smoker_charges = insurance_df[insurance_df['smoker'] == 1]['charges']
non_smoker_charges = insurance_df[insurance_df['smoker'] == 0]['charges']

# Plot the PMF
plt.figure(figsize=(10, 5))
sns.histplot(smoker_charges, bins=30, kde=True, color='red', label='Smokers'
```

```python
sns.histplot(non_smoker_charges, bins=30, kde=True, color='blue', label='Nor

plt.legend()
plt.title("PMF of Insurance Charges for Smokers vs. Non-Smokers")
plt.xlabel("Charges")
plt.ylabel("Probability Density")
plt.show()
```



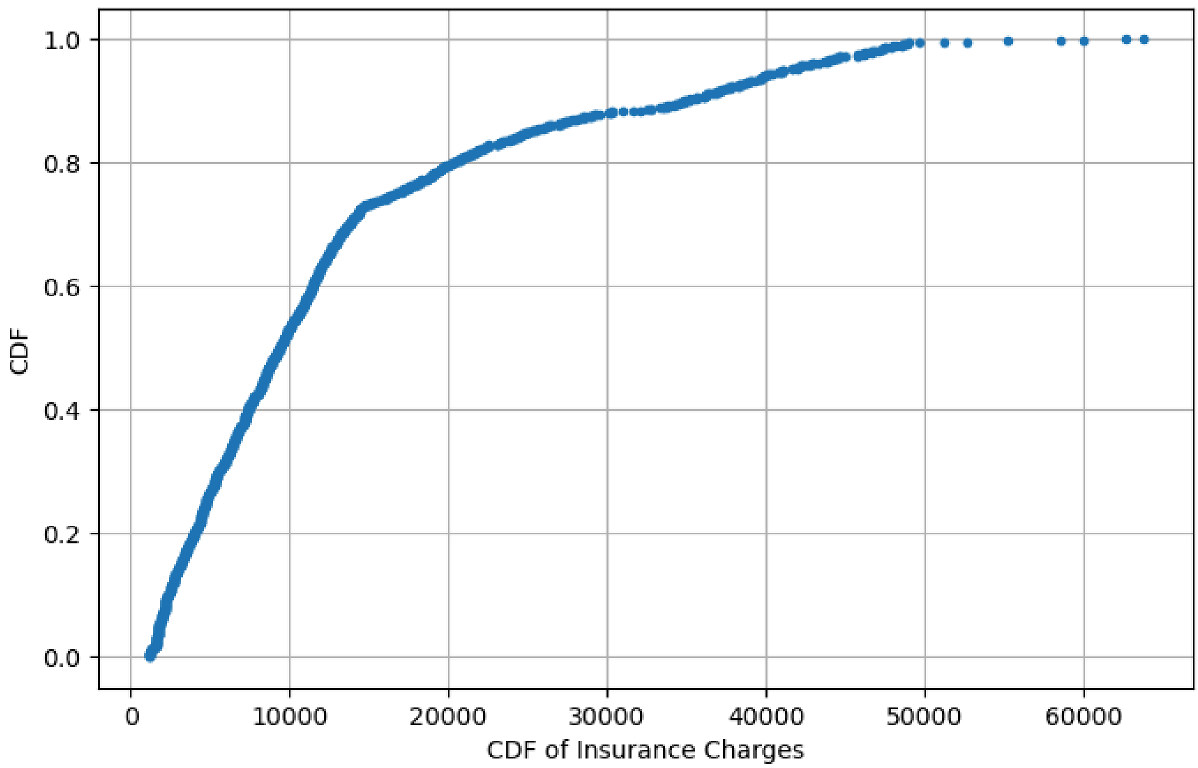
In [ ]:

```python
#Smokers have significantly higher insurance charges.
#The distribution shifts right, indicating a strong effect.
```

In [23]:

```python
# Compute CDF
def plot_cdf(data, title):
    sorted_data = np.sort(data)
    y = np.arange(1, len(sorted_data) + 1) / len(sorted_data)
    plt.plot(sorted_data, y, marker='.', linestyle='none')
    plt.xlabel(title)
    plt.ylabel('CDF')
    plt.grid()

plt.figure(figsize=(8, 5))
plot_cdf(insurance_df['charges'], "CDF of Insurance Charges")
plt.show()
```

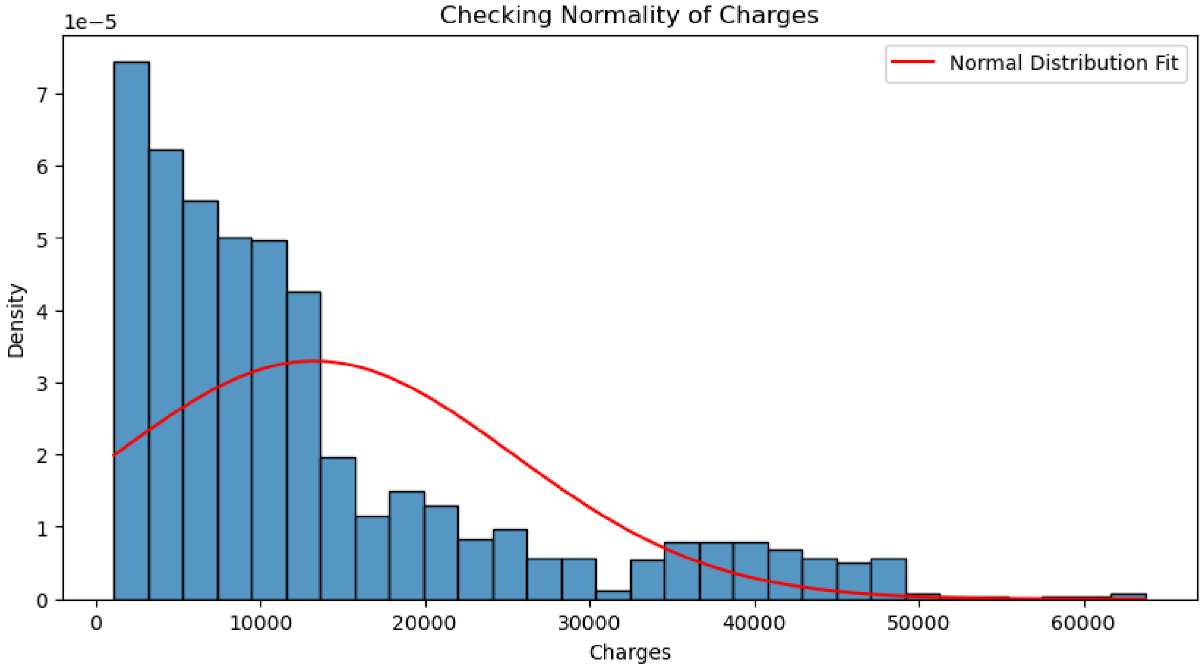


In [ ]:
```
#A steep rise indicates where most people’s charges are concentrated.
#This helps understand the distribution and percentiles.
```

In [29]:
```
# Fit a normal distribution to charges
mu, sigma = stats.norm.fit(insurance_df['charges'])

# Plot the histogram with the normal distribution curve
plt.figure(figsize=(10, 5))
sns.histplot(insurance_df['charges'], bins=30, kde=False, stat="density")
x = np.linspace(min(insurance_df['charges']), max(insurance_df['charges']),
plt.plot(x, stats.norm.pdf(x, mu, sigma), label='Normal Distribution Fit', c

plt.title('Checking Normality of Charges')
plt.xlabel('Charges')
plt.ylabel('Density')
plt.legend()
plt.show()
```

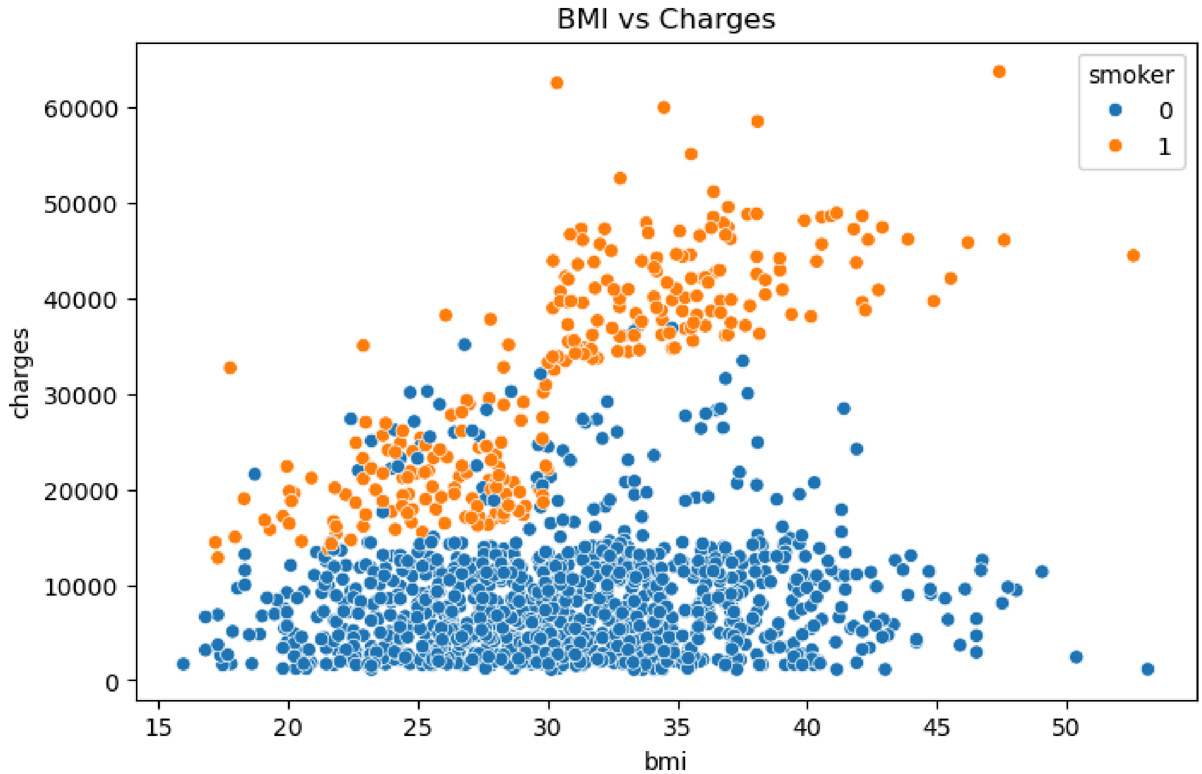


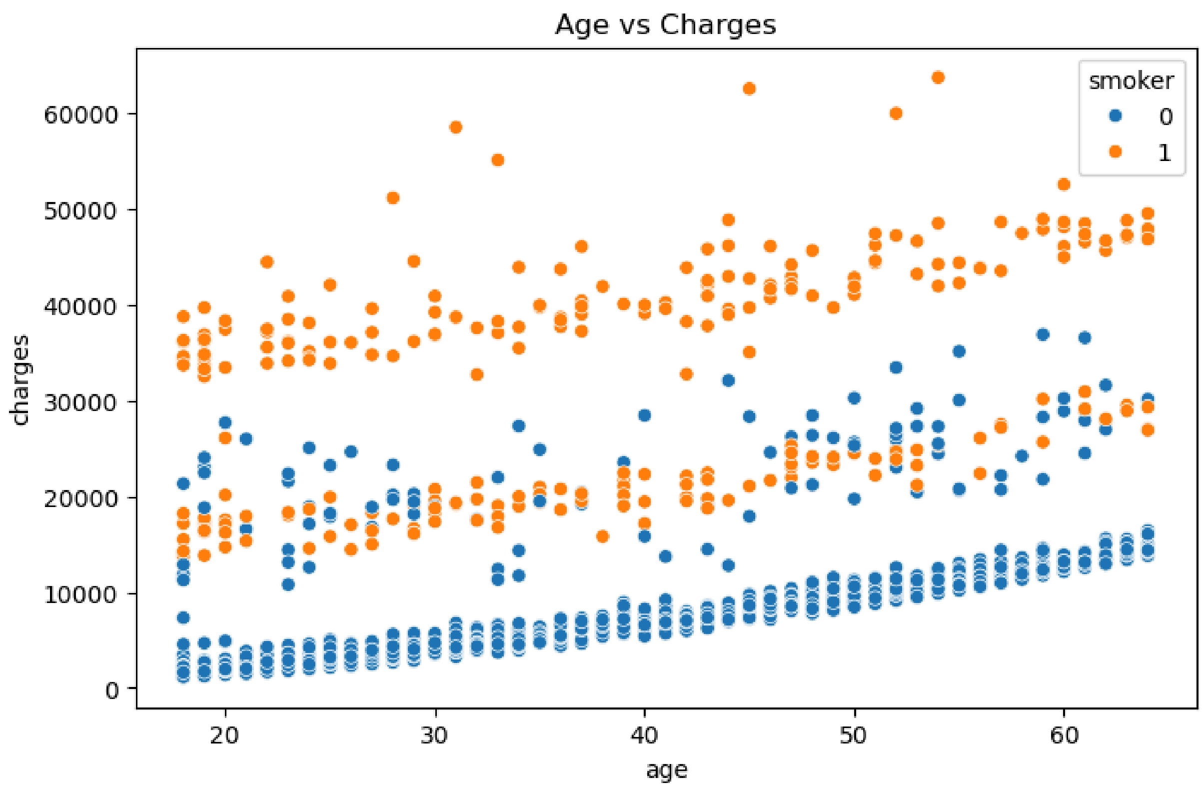
In [ ]:
```
#The actual distribution is right-skewed, not normally distributed.
#This confirms outliers affect charges.
```

In [33]:
```
# Scatter plot: BMI vs. Charges
plt.figure(figsize=(8, 5))
sns.scatterplot(x=insurance_df['bmi'], y=insurance_df['charges'], hue=insura
plt.title("BMI vs Charges")
plt.show()

# Scatter plot: Age vs. Charges
plt.figure(figsize=(8, 5))
sns.scatterplot(x=insurance_df['age'], y=insurance_df['charges'], hue=insura
plt.title("Age vs Charges")
plt.show()

# Pearson correlation
print(insurance_df[['age', 'bmi', 'charges']].corr())
```





	age	bmi	charges
age	1.000000	0.109272	0.299008
bmi	0.109272	1.000000	0.198341
charges	0.299008	0.198341	1.000000

In [ ]:
```
#BMI vs. Charges: A weak trend, but smokers have higher charges.
#Age vs. Charges: Slight positive correlation.
```

In [37]:
```python
# Perform a t-test
#Testing if smokers have significantly higher charges.



t_stat, p_val = stats.ttest_ind(smoker_charges, non_smoker_charges)

print(f"T-Statistic: {t_stat}, P-Value: {p_val}")

# Interpretation
if p_val < 0.05:
    print("Reject Null Hypothesis: Smoking significantly affects charges.")
else:
    print("Fail to Reject Null Hypothesis: No significant effect.")
```

```
T-Statistic: 46.66492117272371, P-Value: 8.271435842179101e-283
Reject Null Hypothesis: Smoking significantly affects charges.
```

In [41]:
```python
# Simple Linear Regression
model = smf.ols('charges ~ bmi', data=insurance_df).fit()
print(model.summary())
```

```
                            OLS Regression Results
==============================================================================
==
Dep. Variable:                charges   R-squared:                       0.0
39
Model:                            OLS   Adj. R-squared:                  0.0
39
Method:                 Least Squares   F-statistic:                     54.
71
Date:                Sat, 01 Mar 2025   Prob (F-statistic):           2.46e-
13
Time:                        16:00:19   Log-Likelihood:                -1445
1.
No. Observations:                1338   AIC:                         2.891e+
04
Df Residuals:                    1336   BIC:                         2.892e+
04
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
==
                 coef    std err          t      P>|t|      [0.025      0.97
5]
------------------------------------------------------------------------------
--
Intercept   1192.9372   1664.802      0.717      0.474   -2072.974    4458.8
49
bmi          393.8730     53.251      7.397      0.000     289.409     498.3
37
==============================================================================
==
Omnibus:                      261.030   Durbin-Watson:                   1.9
83
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              431.0
91
Skew:                           1.297   Prob(JB):                     2.45e-
94
Kurtosis:                       4.004   Cond. No.                         16
0.
==============================================================================
==

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is corre
ctly specified.
```

In [45]:

```
# Multiple Regression
model = smf.ols('charges ~ age + bmi + smoker', data=insurance_df).fit()
print(model.summary())
```

```
                            OLS Regression Results
============================================================================
==
Dep. Variable:                charges   R-squared:                       0.7
47
Model:                            OLS   Adj. R-squared:                  0.7
47
Method:                 Least Squares   F-statistic:                     131
6.
Date:                Sat, 01 Mar 2025   Prob (F-statistic):               0.
00
Time:                        16:00:52   Log-Likelihood:                -1355
7.
No. Observations:                1338   AIC:                         2.712e+
04
Df Residuals:                    1334   BIC:                         2.714e+
04
Df Model:                           3
Covariance Type:            nonrobust
============================================================================
==
                 coef    std err          t      P>|t|      [0.025      0.97
5]
----------------------------------------------------------------------------
--
Intercept  -1.168e+04    937.569    -12.454      0.000   -1.35e+04   -9837.5
61
age          259.5475     11.934     21.748      0.000     236.136     282.9
59
bmi          322.6151     27.487     11.737      0.000     268.692     376.5
38
smoker      2.382e+04    412.867     57.703      0.000     2.3e+04    2.46e+
04
============================================================================
==
Omnibus:                      299.709   Durbin-Watson:                   2.0
77
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              710.1
37
Skew:                           1.213   Prob(JB):                    6.25e-1
55
Kurtosis:                       5.618   Cond. No.                         28
9.
============================================================================
==

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is corre
ctly specified.
```

In [37]:

```
import statsmodels.api as sm

# Define independent variables (X) and dependent variable (y)
X = insurance_df[['age', 'bmi', 'smoker', 'region_northwest', 'region_southe
y = insurance_df['charges']
```

```
# Add a constant for the regression intercept
X = sm.add_constant(X)
```

In [39]:

```
# Fit the regression model
model = sm.OLS(y, X).fit()

# Display regression results
print(model.summary())
```

```
                            OLS Regression Results
==============================================================================
==
Dep. Variable:                charges   R-squared:                       0.7
49
Model:                            OLS   Adj. R-squared:                  0.7
48
Method:                 Least Squares   F-statistic:                     66
0.8
Date:                Sat, 01 Mar 2025   Prob (F-statistic):               0.
00
Time:                        15:34:21   Log-Likelihood:                -1355
4.
No. Observations:                1338   AIC:                         2.712e+
04
Df Residuals:                    1331   BIC:                         2.716e+
04
Df Model:                           6
Covariance Type:            nonrobust
==============================================================================
========
                       coef    std err          t      P>|t|      [0.025
0.975]
------------------------------------------------------------------------------
--------
const             -1.16e+04    976.200    -11.884      0.000   -1.35e+04   -
9686.503
age                258.6365     11.930     21.680      0.000     235.233
282.040
bmi                340.0076     28.673     11.858      0.000     283.759
396.256
smoker            2.385e+04    413.508     57.683      0.000     2.3e+04
2.47e+04
region_northwest  -303.5207    477.850     -0.635      0.525   -1240.943
633.901
region_southeast -1038.6326    480.486     -2.162      0.031   -1981.225
-96.040
region_southwest  -915.9394    479.558     -1.910      0.056   -1856.712
24.833
==============================================================================
==
Omnibus:                      298.282   Durbin-Watson:                   2.0
79
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              705.0
89
Skew:                           1.208   Prob(JB):                    7.80e-1
54
Kurtosis:                       5.609   Cond. No.                         30
7.
==============================================================================
==

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is corre
ctly specified.
```

In [ ]:
```
#Smoking is the strongest predictor of charges.
#BMI has a mild effect but interacts with smoking.
#Age slightly influences charges.
```