Министерство науки и высшего образования РФ
ФГБОУ ВО «Нижневартовский государственный университет»
Факультет информационных технологий и математики
Кафедра информатики и методики преподавания информатики

Курсовой проект по дисциплине
«Основы научно-исследовательской деятельности»

АНАЛИЗ СТАТИСТИКИ СТРАХОВЫХ ВЫПЛАТ СРЕДСТВАМИ ЯЗЫКА PYTHON

Исполнитель:
студент группы 3353
Горидько Иван Александрович
Руководитель:
кандидат технических наук,
доцент кафедры информатики и методики
преподавания информатики
Катермина Татьяна Сергеевна

_____ _____
(подпись)

Нижневартовск, 2025

Оглавление:

Анализ статистики страховых выплат средствами языка Python

Введение

Актуальность исследований. В современном мире страхование является неотъемлемой частью финансовой стабильности как отдельных людей, так и целых предприятий. Страховые выплаты составляют значительную долю в расходах компаний, и их точное прогнозирование напрямую влияет на прибыльность и устойчивость страхового бизнеса. Машинное обучение и интеллектуальный анализ данных стали мощными инструментами для решения этой задачи, позволяя выявлять сложные, неочевидные для человека взаимосвязи в исторических данных. Язык Python с его богатой экосистемой библиотек, таких как Scikit-learn, Pandas и NumPy, является одной из самых популярных и эффективных сред для реализации подобных аналитических систем. Прогнозирование страховых выплат — это сложная и многогранная задача. Для ее успешного решения необходимо разбить процесс на ключевые этапы: от сбора и предобработки данных до построения, валидации и интерпретации прогнозных моделей.

Объект исследования: Формирование размеров страховых выплат на основе статистических данных.

Процесс исследования: Формирование размеров страховых выплат средствами интеллектуального анализа данных.

Цель исследования: Реализовать предсказания страховых выплат средствами языка python.

Задачи исследования:

- Провести анализ и предобработку предоставленного набора данных о страховых случаях.
- Реализовать и обучить модель линейной регрессии для прогнозирования размера страховых выплат.
- Реализовать и обучить модель регрессии на основе алгоритма "Случайный лес".
- Провести сравнительный анализ эффективности построенных моделей и

выбрать оптимальную.

Глава 1. ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ПРОГНОЗИРОВАНИЯ СТРАХОВЫХ ВЫПЛАТ СРЕДСТВАМИ МАШИННОГО ОБУЧЕНИЯ

1.1. Задача предсказания страховых выплат

Задача предсказания страховых выплат относится к классу задач прогнозной аналитики и машинного обучения с учителем, а именно — к задачам регрессии.

Её основная цель заключается в построении математической модели, способной на основе исторических данных о клиентах (таких как возраст, индекс массы тела, вредные привычки, регион проживания и другие демографические признаки) прогнозировать непрерывную числовую величину — ожидаемый размер страховой выплаты.

Точное прогнозирование позволяет страховым компаниям оптимизировать тарифную политику, более эффективно управлять рисками и финансовыми резервами, а также выявлять скрытые закономерности, влияющие на стоимость страховых случаев.

1.2. Уточненная постановка задачи

В рамках данной работы решается задача множественной линейной регрессии, где целевой переменной (зависимой переменной) является столбец charges (размер страховых выплат), а признаками (независимыми переменными) — все остальные столбцы в наборе данных: age, sex, bmi, children, smoker, region.

Категориальные признаки (такие как sex, smoker, region, а также производный признак weight_category) перед обучением модели подвергаются процедуре One-Hot Encoding.

Процесс решения задачи включает в себя следующие этапы: первичный разведочный анализ данных (EDA), предобработка данных (обработка пропусков, кодирование категориальных переменных, нормализация числовых признаков), разделение данных на обучающую и тестовую выборки, обучение и валидация моделей, а также сравнительный анализ их эффективности по метрикам MSE, MAE и R^2.

1.3. Библиотеки, язык и среда разработки

Работа выполнена на языке программирования Python версии 3.x, который был выбран благодаря его широкой распространённости в задачах анализа данных и машинного обучения, а также наличию мощного стека специализированных библиотек.

В качестве основной среды разработки использовалась Jupyter Notebook, что позволяет интерактивно выполнять код, визуализировать данные и документировать ход исследования.

Для решения поставленных задач были применены следующие библиотеки: Pandas — для манипуляций с данными и их загрузки; NumPy — для выполнения численных операций; Matplotlib и Seaborn — для визуализации данных и построения графиков; Scikit-learn — для предобработки данных, реализации алгоритмов машинного обучения (Linear Regression, Random Forest), их обучения и оценки качества.

1.4. Первичный разведочный анализ данных (EDA). Часть 1.

1. Импортируем базовые библиотеки и выгрузим таблицу из csv файла:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

med_insurance = pd.read_csv('insurance.csv')
print(med_insurance.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
```

2. Суммируем данные:

```
print(med_insurance.describe(include="all"))
```

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

```
          charges
count    1338.000000
unique           NaN
top              NaN
freq             NaN
mean    13270.422265
std     12110.011237
min      1121.873900
25%      4740.287150
50%      9382.033000
75%     16639.912515
max     63770.428010
```

print(med_insurance.dtypes)

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

3. Нормализуем данные, создадим производный признак weight_category:

```
partition_by_bmi = lambda bmi: 'Underweight' if bmi < 25.0 else 'Overweight'
bmi_copy = med_insurance['bmi']
med_insurance_bmi = bmi_copy.apply(partition_by_bmi)
med_insurance_bmi = med_insurance_bmi.to_frame()

med_insurance_bmi.rename(columns={'bmi': 'weight_category'}, inplace=True)
med_insurance_bmi['bmi'] = med_insurance.bmi

partition_by_bmi_2 = lambda row: 'Normal weight' if (row.bmi >= 18.5 and row.bmi <
25.0) else row.weight_category
med_insurance_bmi['weight_category'] =
med_insurance_bmi.apply(partition_by_bmi_2, axis=1)

partition_by_bmi_3 = lambda row: 'Obesity' if row.bmi > 30.0 else
row.weight_category
med_insurance_bmi['weight_category'] =
med_insurance_bmi.apply(partition_by_bmi_3, axis=1)

print(med_insurance_bmi.head(30))
```

```
   weight_category     bmi
0       Overweight  27.900
1          Obesity  33.770
2          Obesity  33.000
3    Normal weight  22.705
4       Overweight  28.880
5       Overweight  25.740
6          Obesity  33.440
7       Overweight  27.740
8       Overweight  29.830
9       Overweight  25.840
10      Overweight  26.220
11      Overweight  26.290
12         Obesity  34.400
13         Obesity  39.820
14         Obesity  42.130
15   Normal weight  24.600
16         Obesity  30.780
17   Normal weight  23.845
18         Obesity  40.300
19         Obesity  35.300
20         Obesity  36.005
21         Obesity  32.400
22         Obesity  34.100
23         Obesity  31.920
24      Overweight  28.025
25      Overweight  27.720
26   Normal weight  23.085
27         Obesity  32.775
28     Underweight  17.385
29         Obesity  36.300
```

4. Нормализуем weight_category и region при помощи One-Hot Encoding:

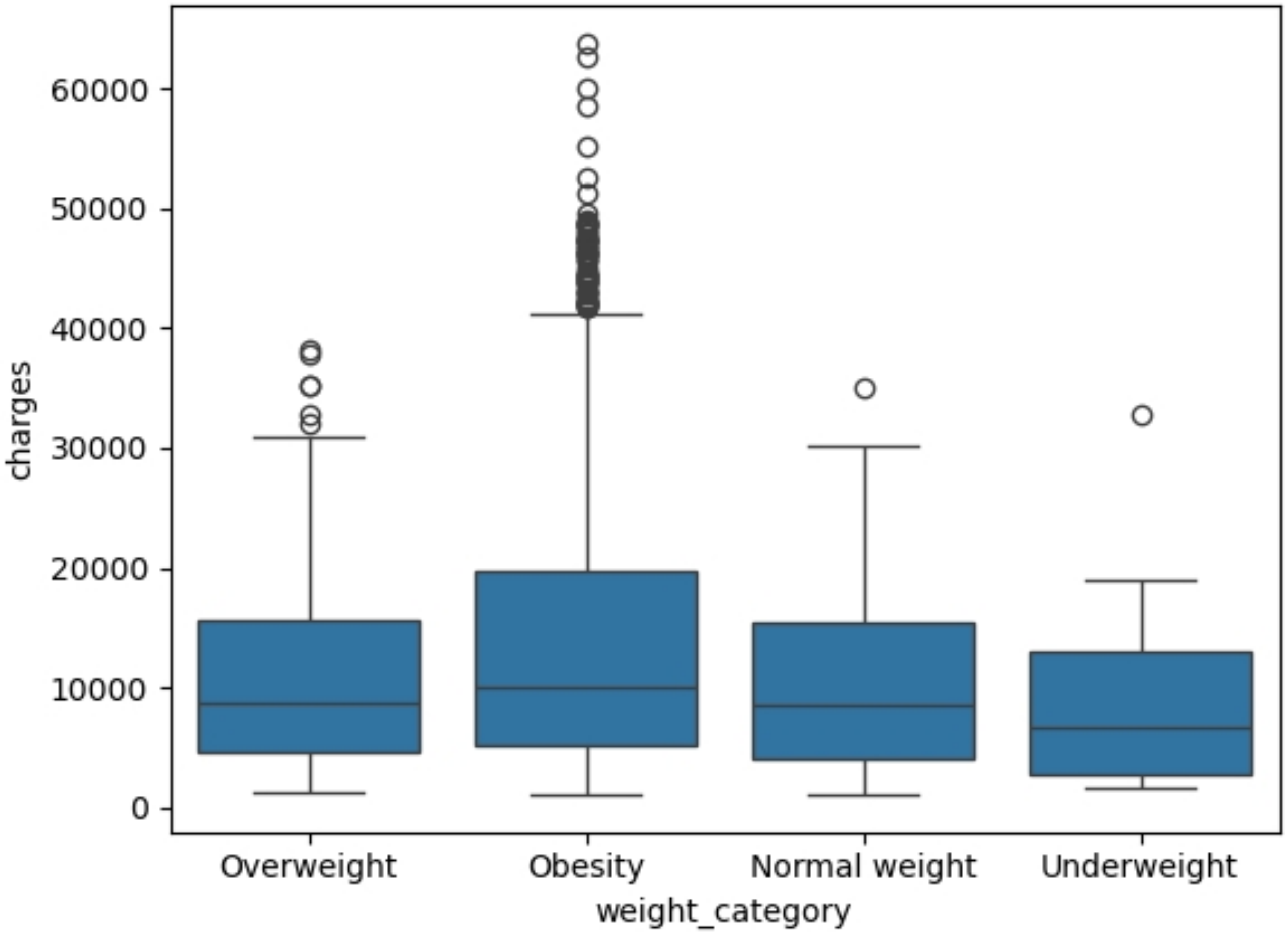
```
med_insurance_2 = med_insurance[['age', 'sex', 'bmi', 'children', 'smoker', 'region', 'charges']]
med_insurance_2['weight_category'] = med_insurance_bmi['weight_category']
print(med_insurance_2.head(5))

med_insurance_2 = pd.get_dummies(data=med_insurance_2, columns=['weight_category', 'region'], dtype='int')
print(med_insurance_2.head(5))
```

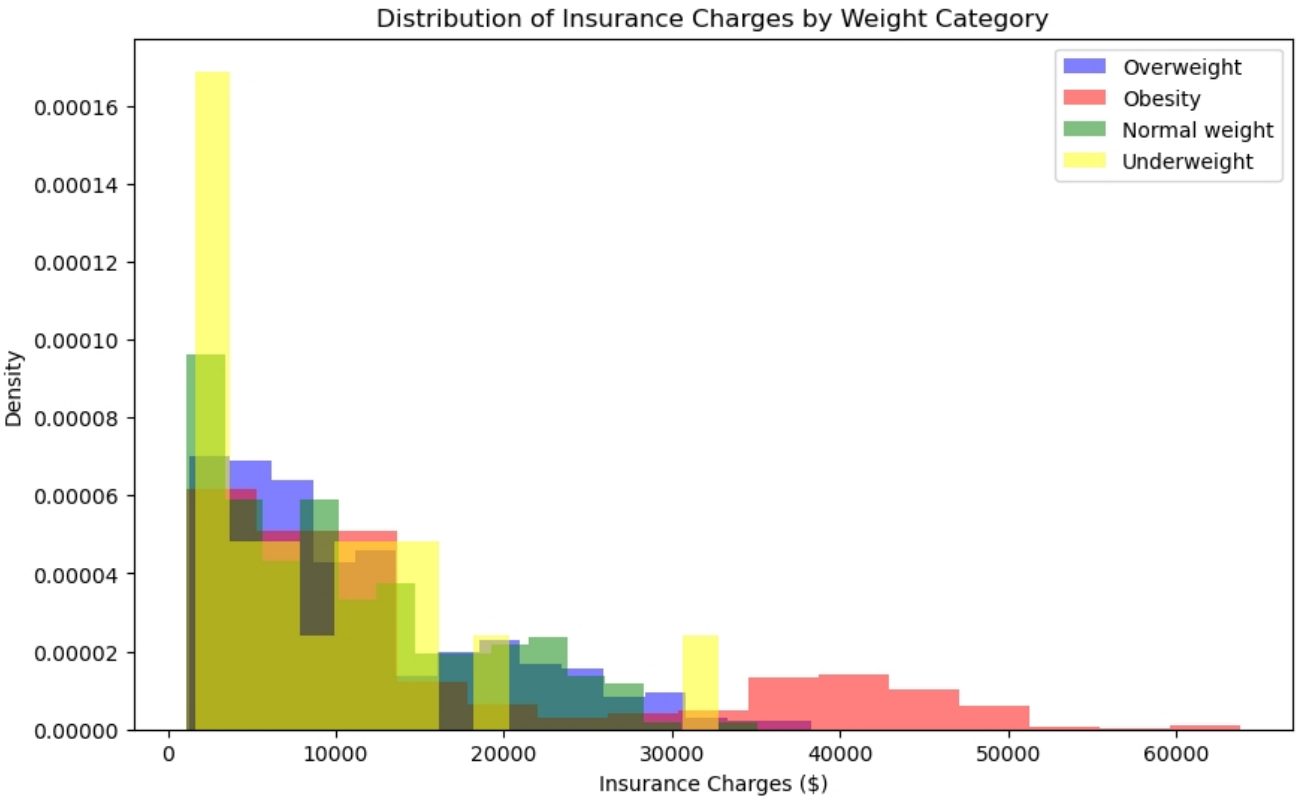
```
   age     sex     bmi  children smoker      charges  \
0   19  female  27.900         0    yes  16884.92400
1   18    male  33.770         1     no   1725.55230
2   28    male  33.000         3     no   4449.46200
3   33    male  22.705         0     no  21984.47061
4   32    male  28.880         0     no   3866.85520

   weight_category_Normal weight  weight_category_Obesity  \
0                              0                        0
1                              0                        1
2                              0                        1
3                              1                        0
4                              0                        0

   weight_category_Overweight  weight_category_Underweight  region_northeast  \
0                           1                            0                 0
1                           0                            0                 0
2                           0                            0                 0
3                           0                            0                 0
4                           1                            0                 0

   region_northwest  region_southeast  region_southwest
0                 0                 0                1
1                 0                 1                0
2                 0                 1                0
3                 1                 0                0
4                 1                 0                0
```

1.5. Первичный разведочный анализ данных (EDA). Часть 2.

Ответим на следующие вопросы:

1. Как между собой соотносятся bmi

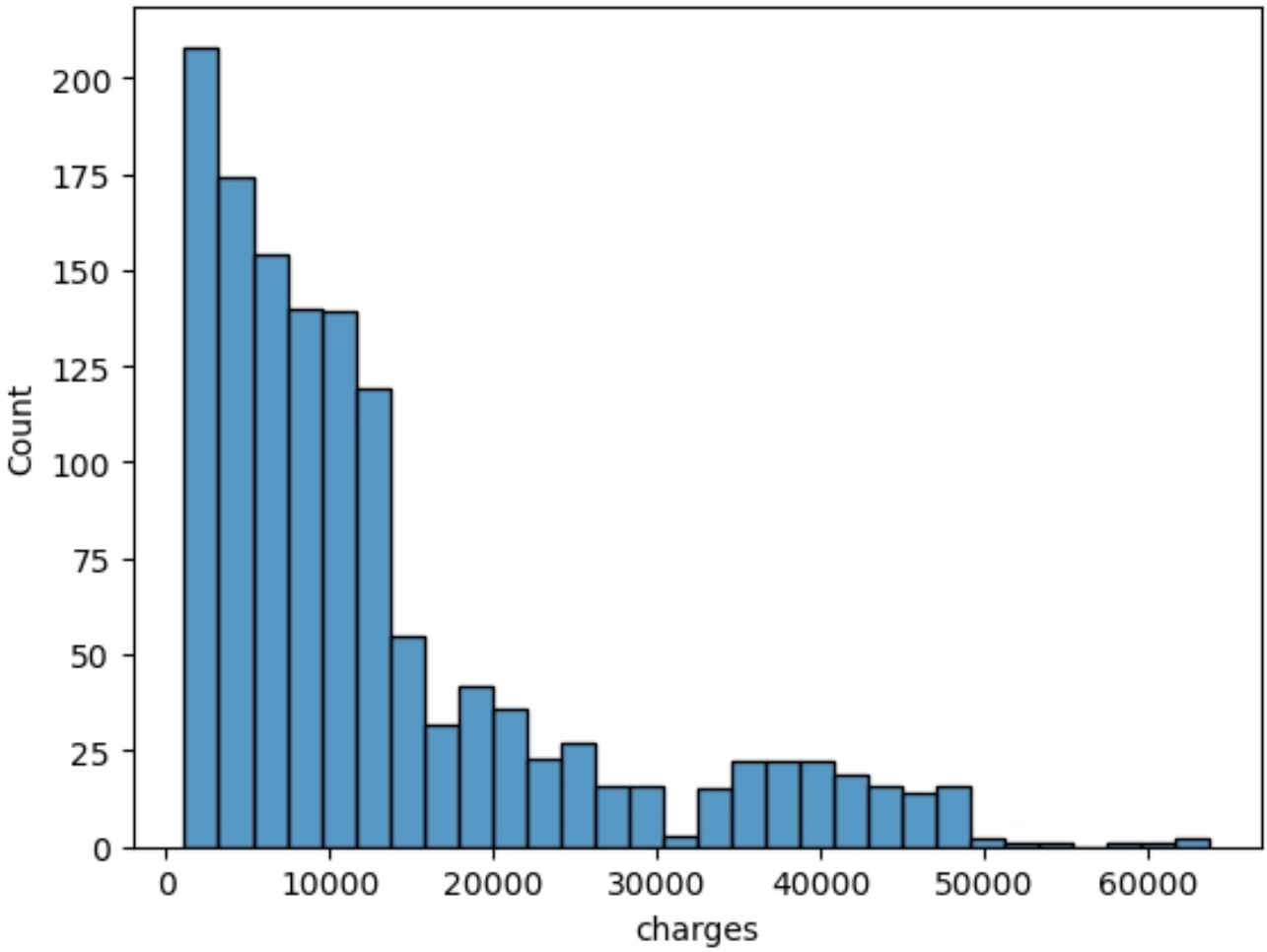

60000
50000
40000
30000
20000
10000
0
charges
Overweight
Obesity
Normal weight
Underweight
weight_category


Distribution of Insurance Charges by Weight Category
Overweight
Obesity
Normal weight
Underweight
0.00016
0.00014
0.00012
0.00010
0.00008
0.00006
0.00004
0.00002
0.00000
Density
0
10000
20000
30000
40000
50000
60000
Insurance Charges ($)

Наблюдение 1. Благодаря визуализации, мы можем рассмотреть что люди с BMI классифицированным как 'Obesity' соблюдают тенденцию иметь более высокую стоимость страховых выплат, а люди с BMI 'Underweight' имеют более низкую стоимость страховых выплат. 'Normal weight' и 'Overweight' в общем и целом одинаковы.

2. Посчитать центральную тенденцию charges.

Mean: 13270.422265141257
Median: 9382.033
Mode: 0 1639.5631
Name: charges, dtype: float64
Trimmed mean: 9877.310386652985



Наблюдение 2. Из графика видно, что данные имеют выраженное правостороннее смещение (положительную асимметрию), что приводит к тому, что среднее значение оказывается выше медианы. Это указывает на наличие экстремально высоких значений в правой части распределения.

Наблюдение 3. Мы видим, что усеченное среднее значение достаточно близко к

медиане и моде. Особенно это заметно при отсечении 20% данных с каждой стороны распределения.

Наблюдение 4. На основе анализа можно сделать вывод, что типичная стоимость страховки составляет около 9400 долларов.

3. Посчитать дисперсию charges.

Range: 62648.554110000005
Interquartile range: 11899.625365
Variance: 146652372.15285483
Standard deviation: 12110.011236693996
Mean Absolute Deviation: 9091.12658113703

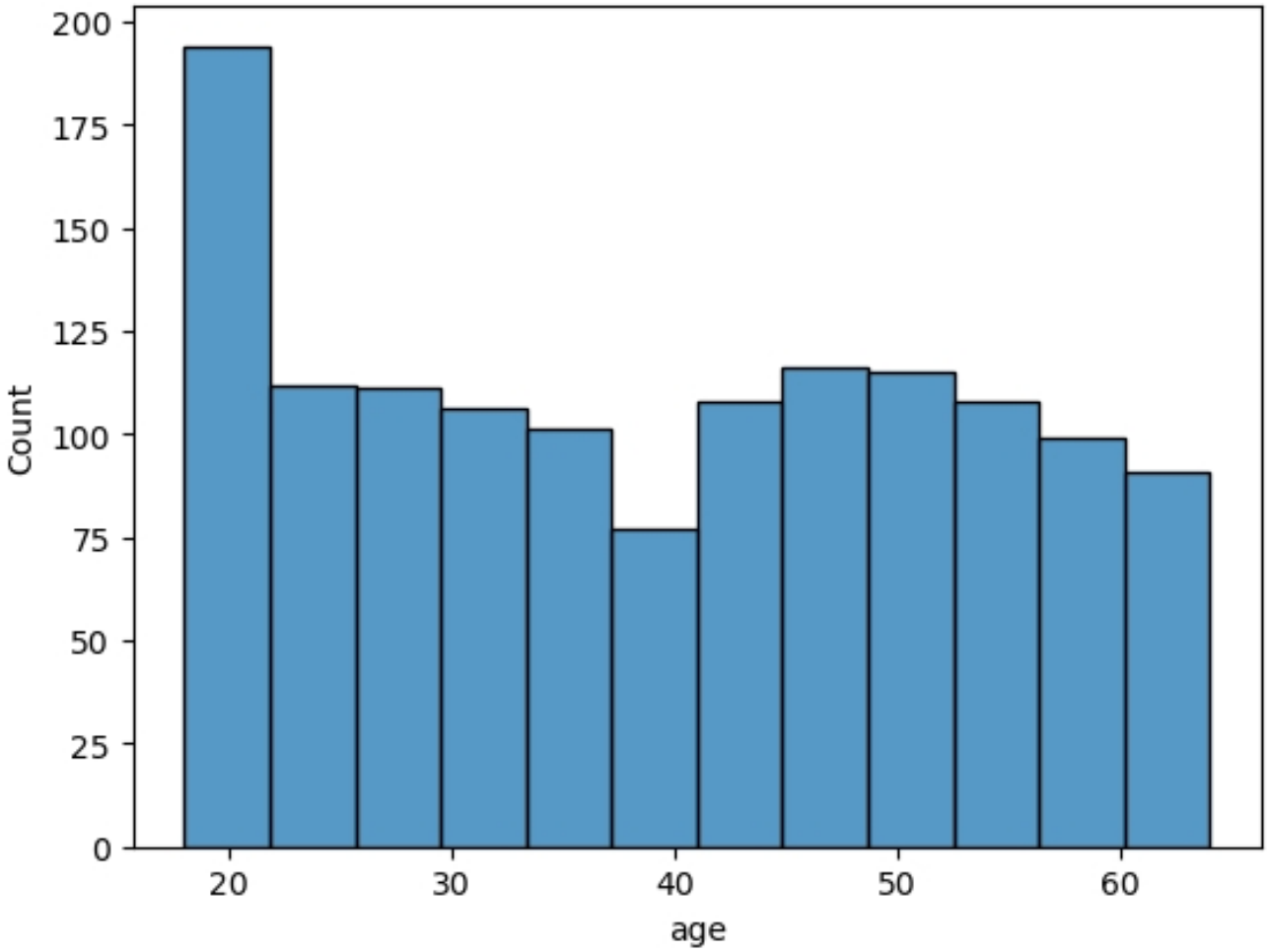
Наблюдение 5. Мы видим, что размах (range) не подходит для описания данных в данном случае, поскольку у одного или нескольких человек стоимость страховых выплат превышает 60 000$. Следовательно, нам следует рассмотреть межквартильный размах (IQR).

Наблюдение 6. Стандартное отклонение не является достаточно информативной мерой в данном случае, поскольку наши данные имеют выраженное правостороннее смещение. Поэтому целесообразно рассмотреть среднее абсолютное отклонение (MAD).

Наблюдение 7. Из *наблюдения 1* мы видим, что центральные 50% данных сконцентрированы вокруг медианы. *Наблюдение 2* показывает, что страховые затраты большинства людей тесно сгруппированы вокруг типичной стоимости.

4. **Посчитать центральную тенденцию ages в датасете.**

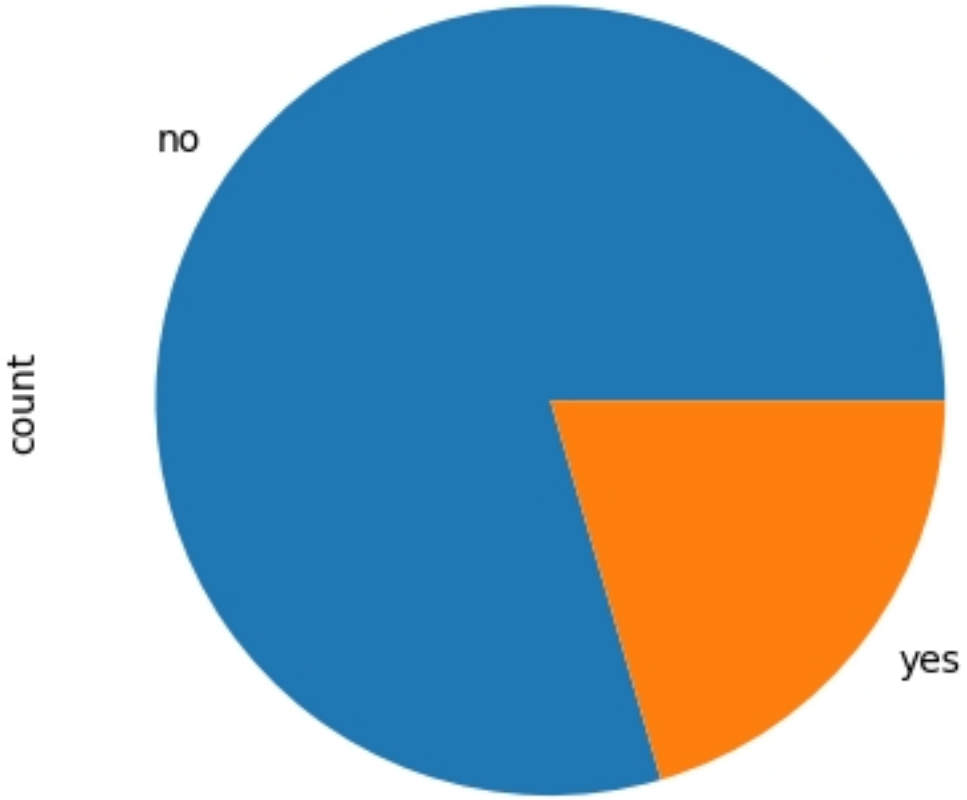
Mean: 39.20702541106129
Median: 39.0
Mode: 0 18
Name: age, dtype: int64
Trimmed mean: 39.02985074626866



Наблюдение 8. На основе данной гистограммы можно наблюдать сильно скошенное распределение с большим количеством молодых людей. Низкая модальная возрастная группа 0-18 лет смещена вниз из-за выбросов, в то время как медиана и среднее значение смещены вверх из-за значительно меньшего количества людей старшего возраста. Возраста наиболее многочисленной группы сосредоточены в самой младшей категории, однако большинство людей находятся вблизи среднего возраста 39 лет.

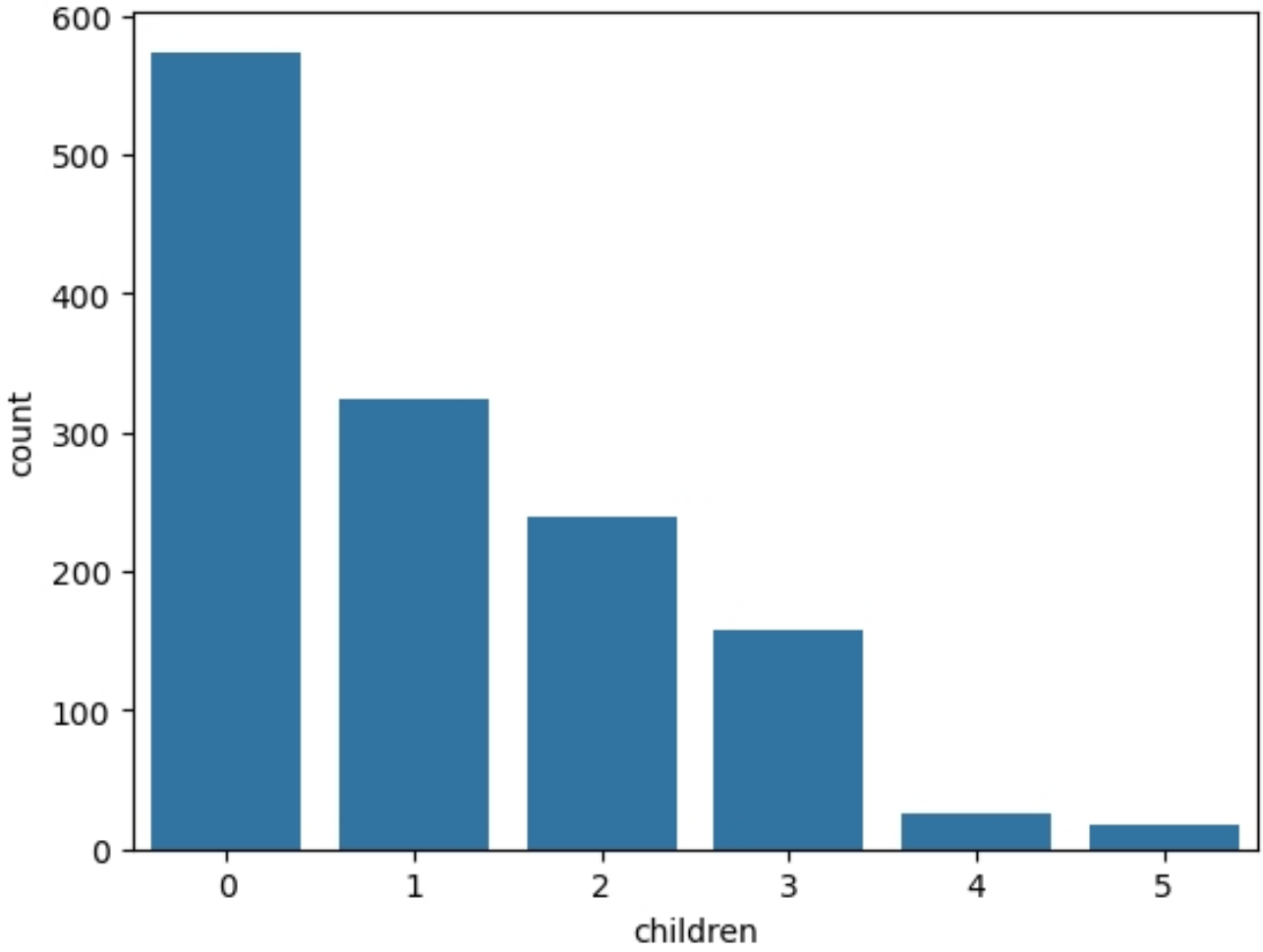
5. **Визуализировать пропорцию smokers и non-smokers.**

```
smoker
no     1064
yes     274
Name: count, dtype: int64
```



Наблюдение 10. Здесь мы можем увидеть что большинство людей в датасете – не курящие.

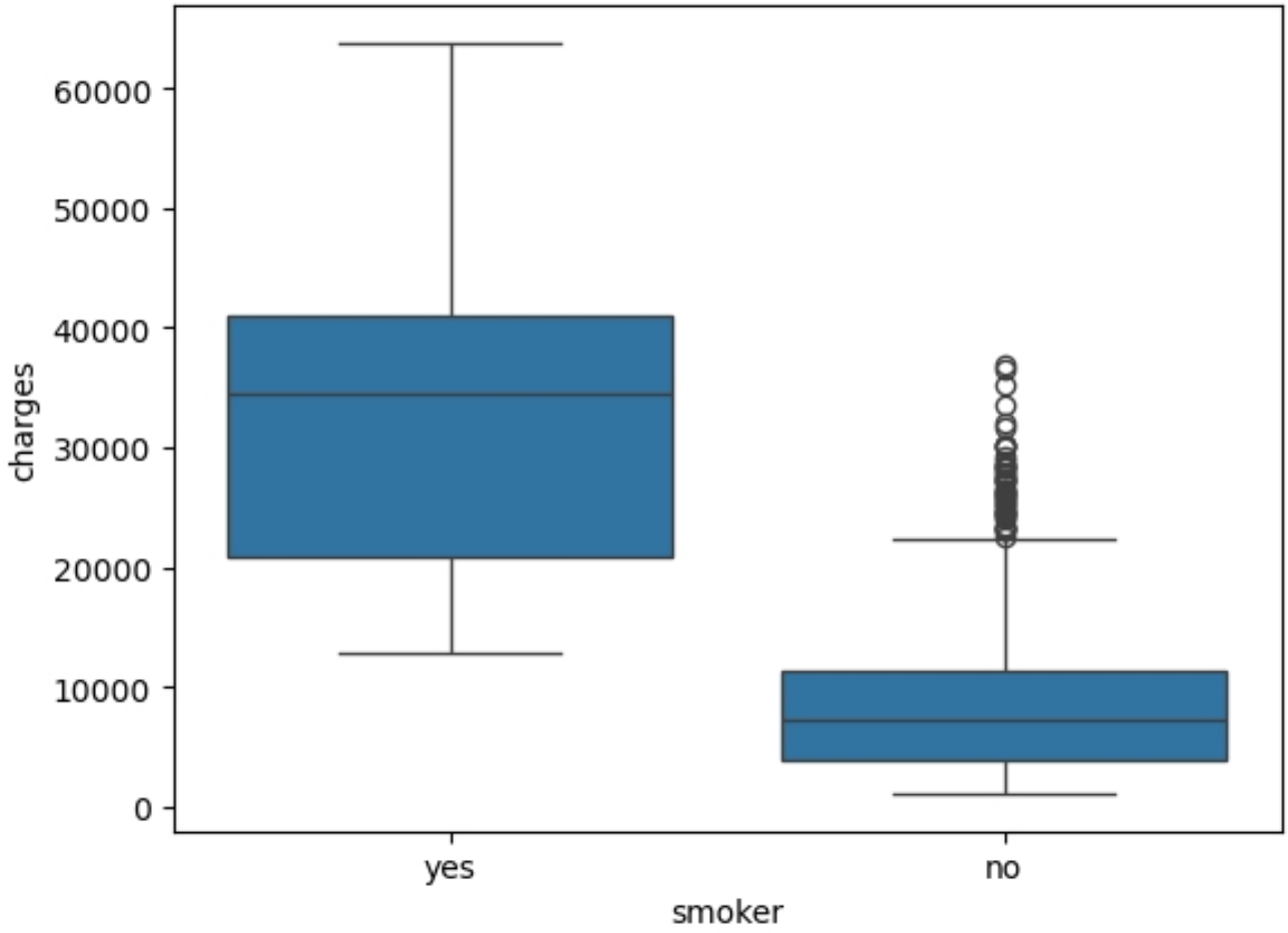
6. **Визуализировать пропорцию children count в датасете.**

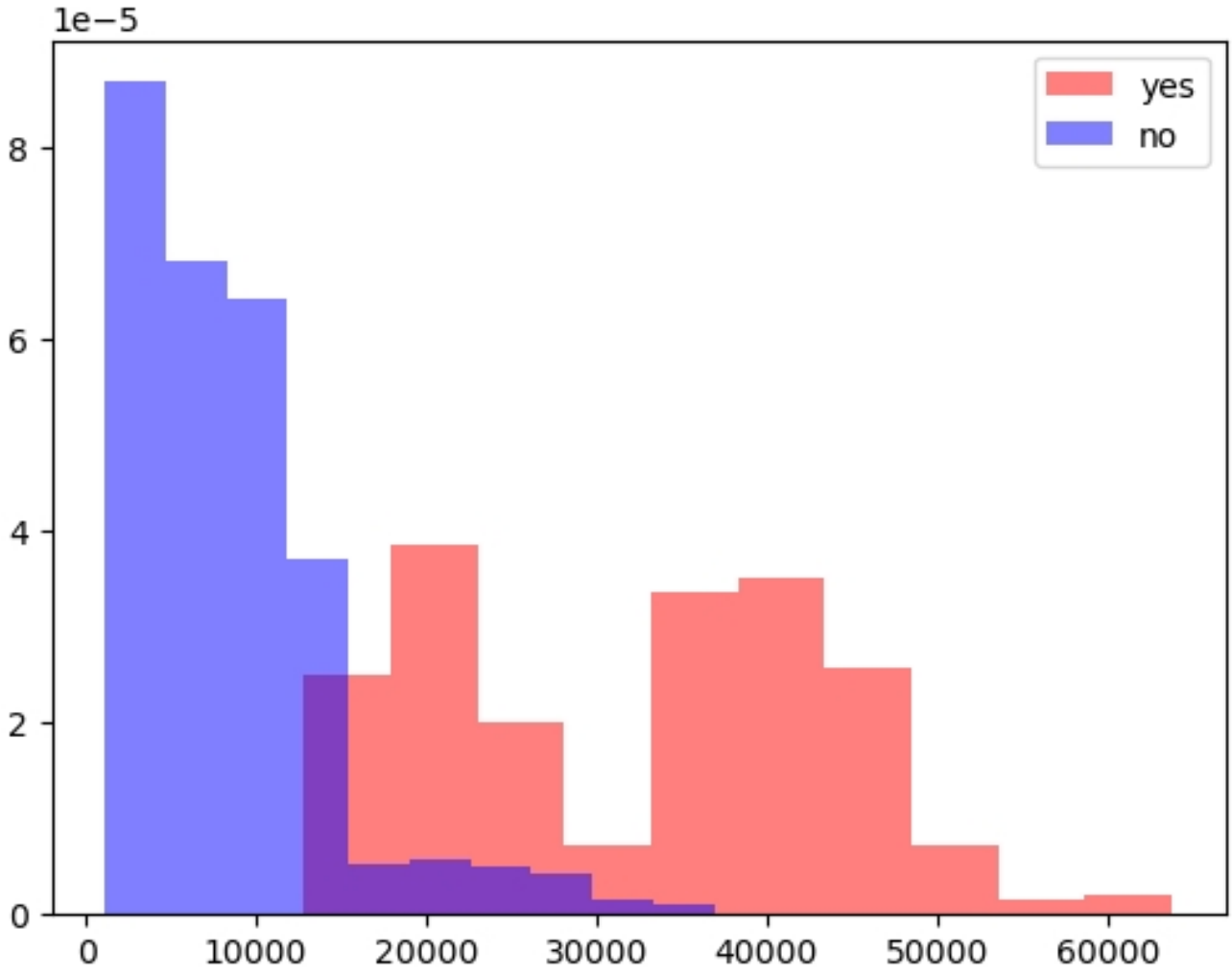


Наблюдение 11. Этот график показывает что большинство людей в датасете имеют 0 или 1 ребенка.

7. **Как между собой соотносятся smoker и charges?**

The difference between charges mean and smoker mean: 23615.963533676637
The difference between charges median and smoker median: 27110.943150000006



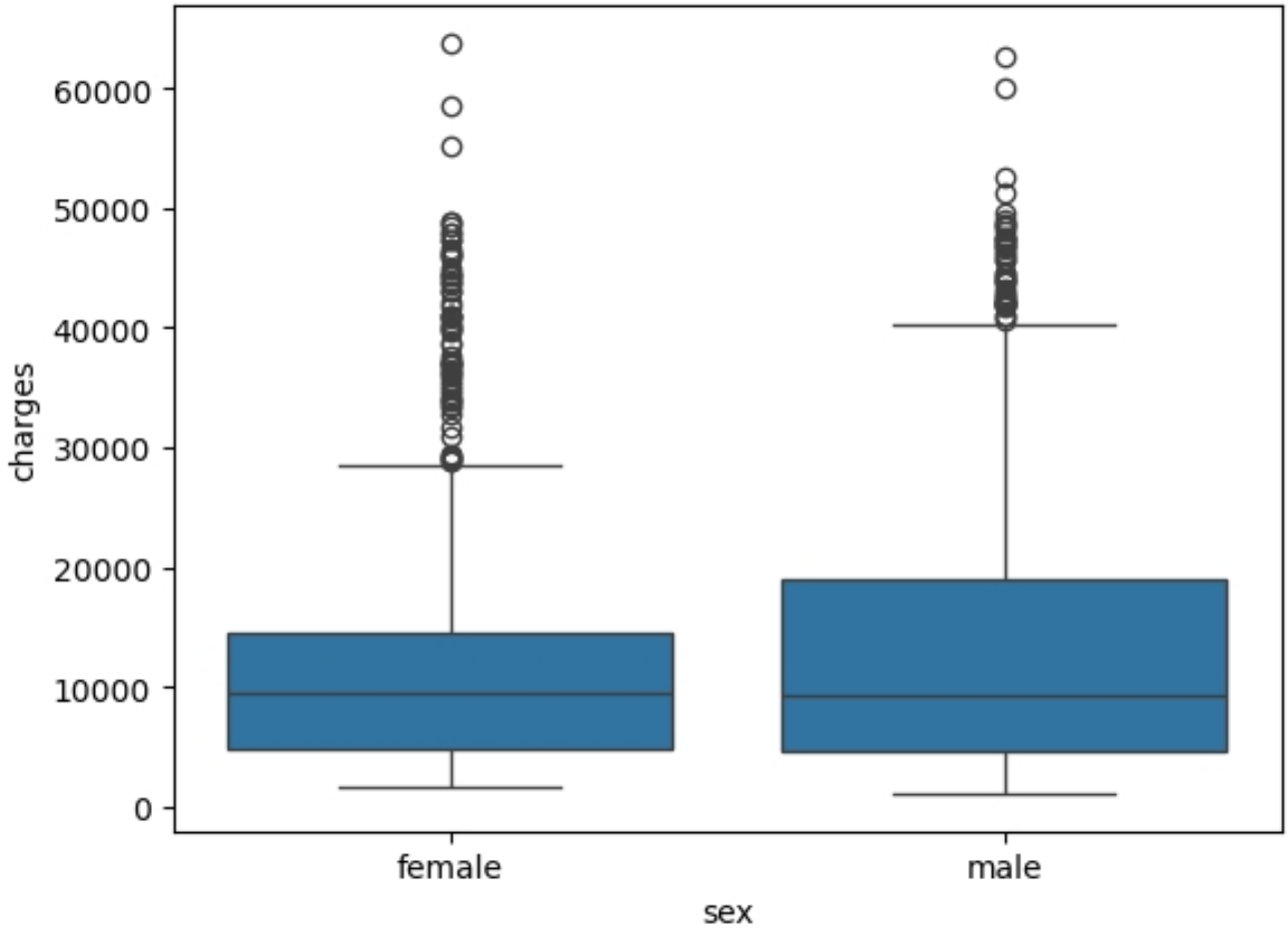


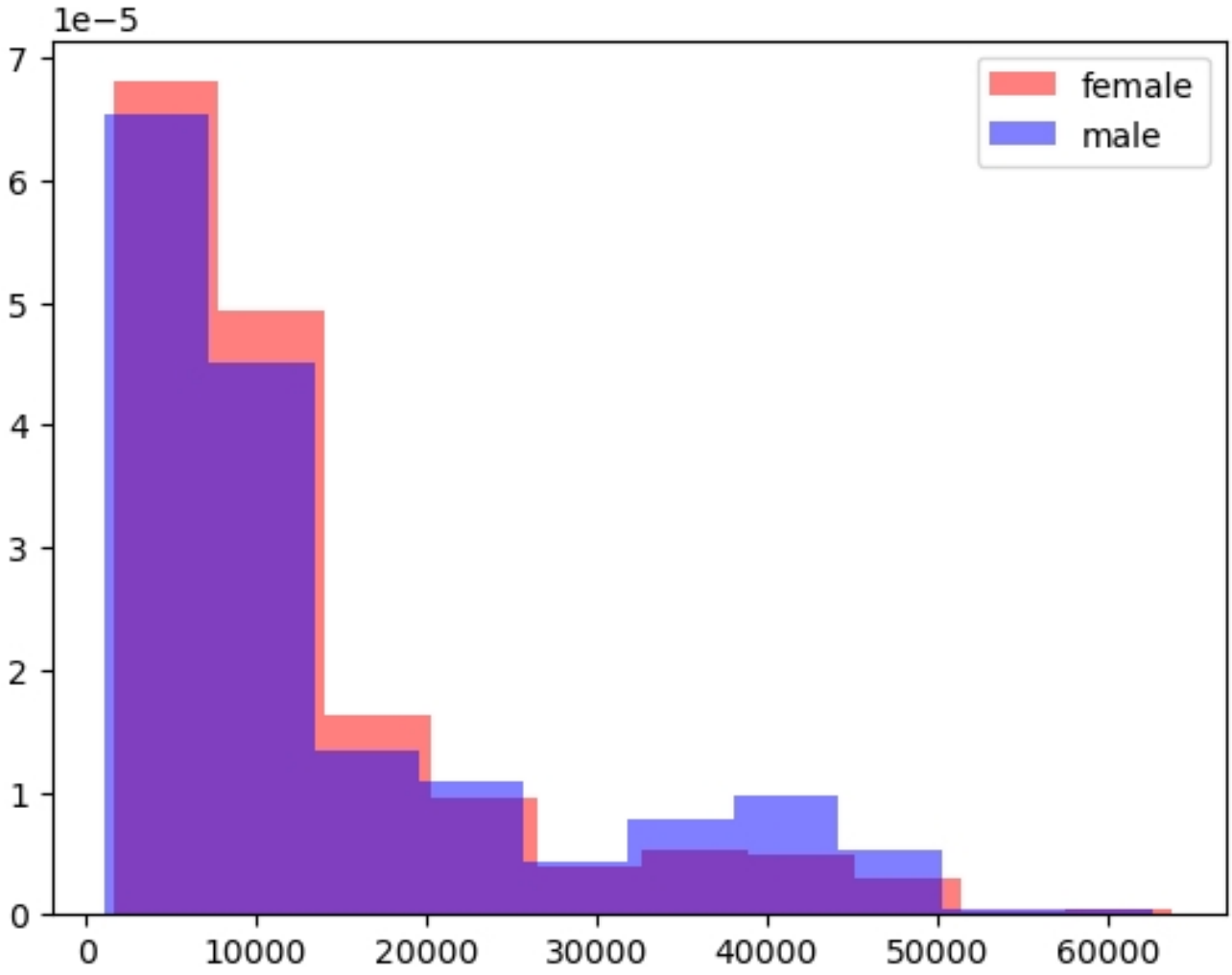
Наблюдение 12. Разница является слишком значительной, следовательно, мы можем предположить, что курение существенно увеличивает стоимость страховки.

Наблюдение 13. Наши boxplot и гистограмма показывают, что между этими категориями отсутствует перекрытие, поэтому мы можем сделать вывод, что курение значительно увеличивает стоимость страховых выплат.

8. Как между собой соотносятся sex и charges?

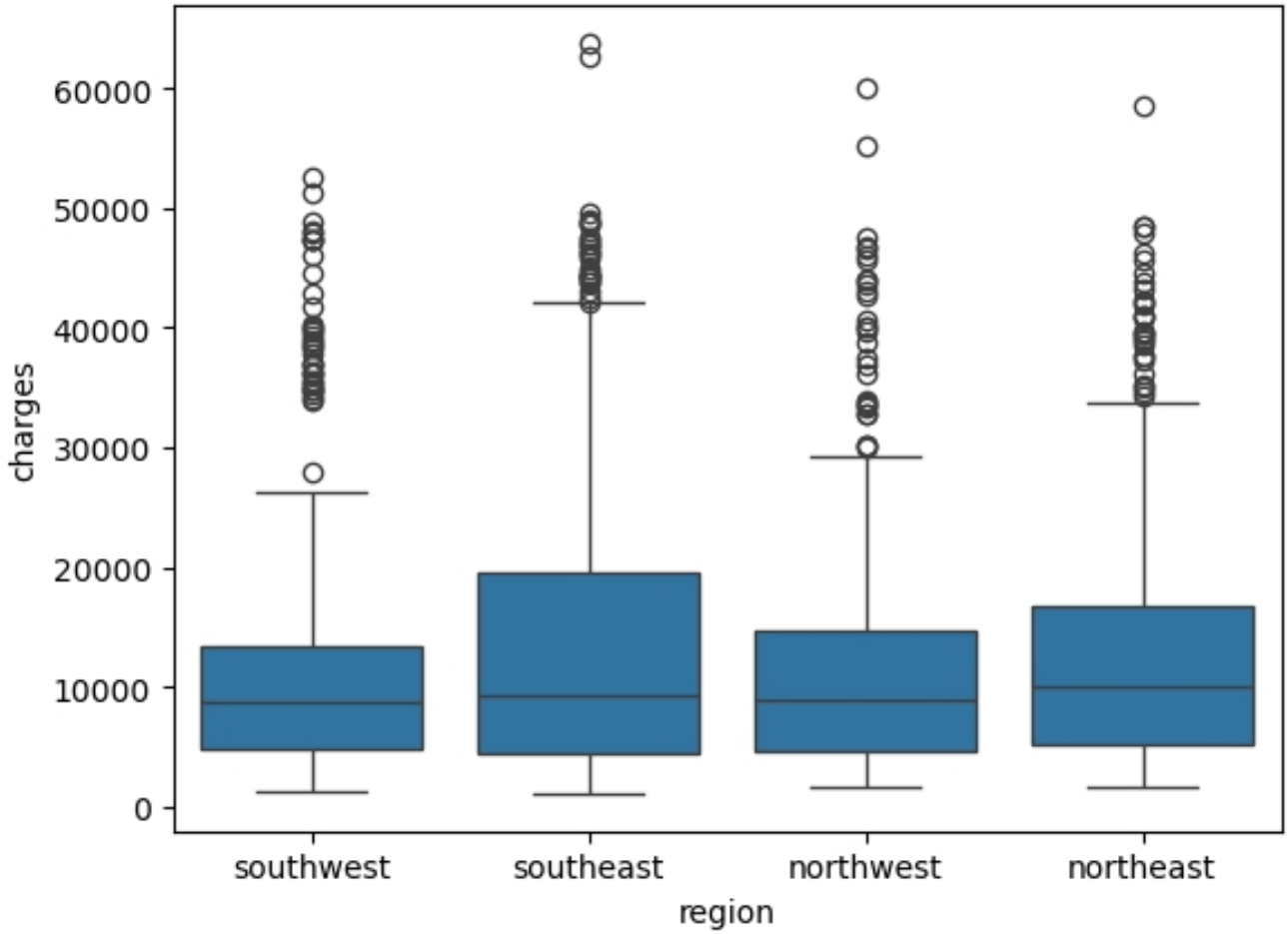
The difference between male mean and female mean: 1387.1723338865468
The difference between male median and female median: -43.34674999999879

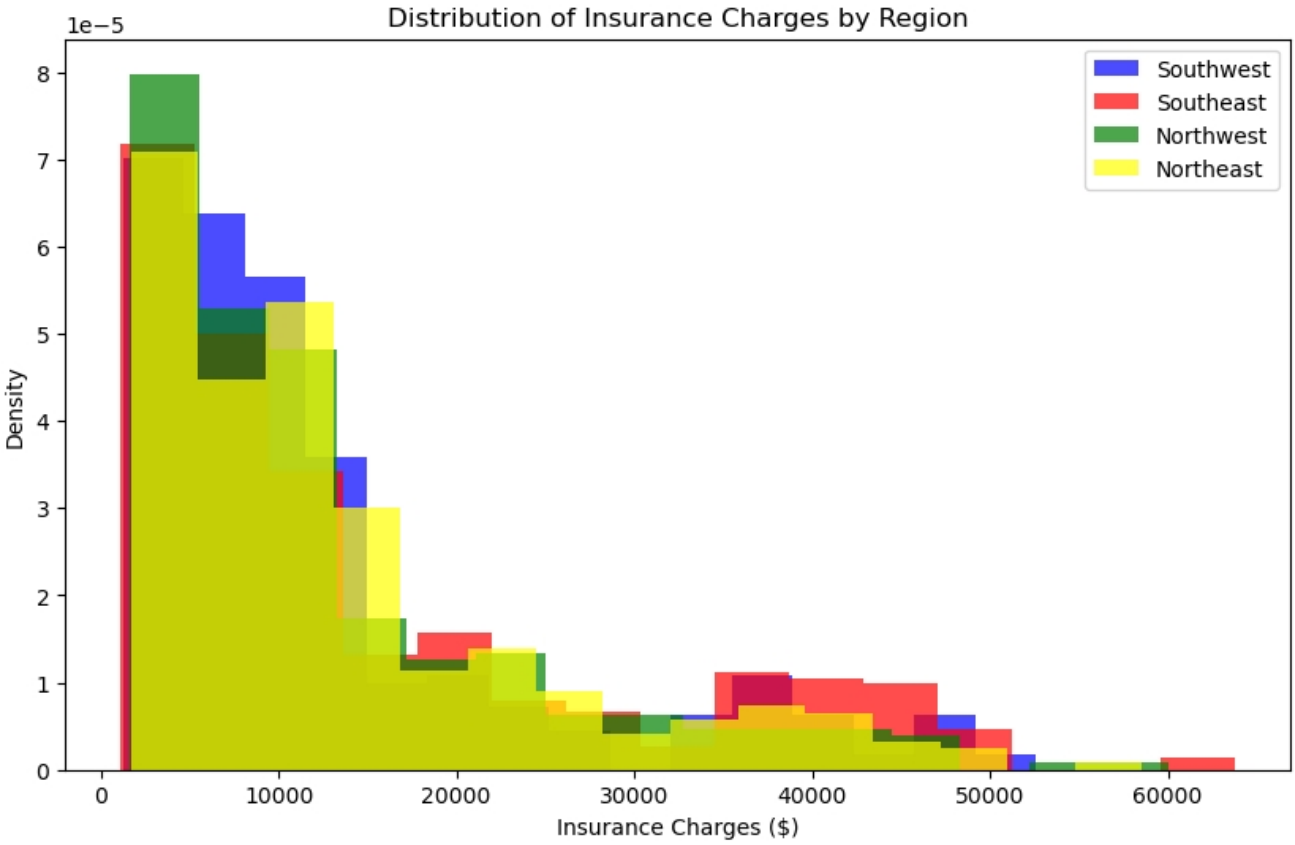




Наблюдение 14. Разница в между средними значениями, медианами и графики показывают что между мужчинами и женщинами не существует явной разницы в стоимости страхования.

9. Как между собой соотносятся region и charges?





Наблюдение 15. Из этих двух графиков видно, что разница между регионами не имеет значительного влияния на повышение стоимости страховых выплат.

1.5. Линейная регрессия. Множественная линейная регрессия.