|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  |  |  | | МИНОБРНАУКИ РОССИИ | | | | Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА** – **Российский технологический университет»**  **РТУ МИРЭА** | | | | |
|  | Институт информационных технологий (ИТ) |
|  | Кафедра прикладной математики |

|  |  |
| --- | --- |
| **ОТЧЁТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ № 3** | |
| **по дисциплине «Технологии и инструментарий анализа больших данных»** | |
|  | |
| Выполнил студент группы ИКБО-20-21 | Сидоров С.Д. |
| Проверил ассистент кафедры ПМ ИИТ | Тетерин Н.Н. |

Москва 2024

**Практическая работа**

1. Загрузить данные из файла “insurance.csv”.

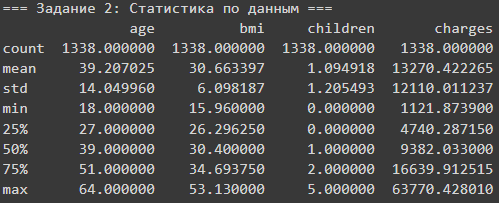
Листинг 1:

|  |
| --- |
| insurance\_data = pd.read\_csv('insurance.csv') |

1. С помощью метода describe() посмотреть статистику по данным. Сделать выводы.

Листинг 2:

|  |
| --- |
| print(insurance\_data.describe()) |

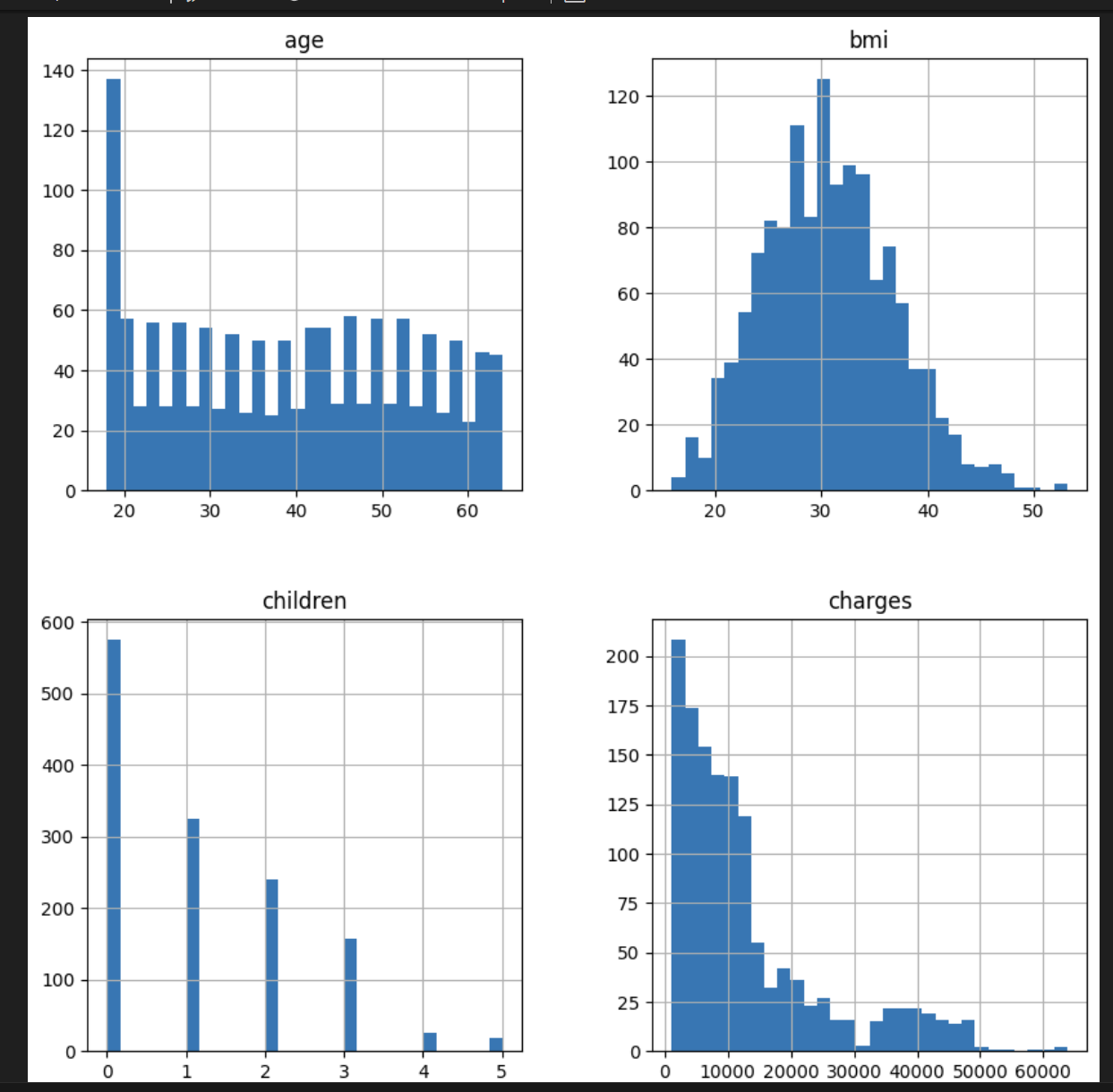


**Выводы:** датасет содержит данные о страховании, включающие возраст, индекс массы тела, количество детей, расходы

1. Построить гистограммы для числовых показателей. Сделать выводы

Листинг 3:

|  |
| --- |
| import matplotlib.pyplot as plt  insurance\_data.hist(bins=30, figsize=(10,10))  plt.show()  #гистограммы частот |



**Вывод**:

Средний возраст участников составляет примерно `39` лет

Средний BMI равен `30.66`, что соответствует категории избыточного веса или ожирения.

В среднем у респондентов `1` ребенок. Большинство - либо `бездетные`, либо имеют `одного` или `двух` детей (максимум - `5`)

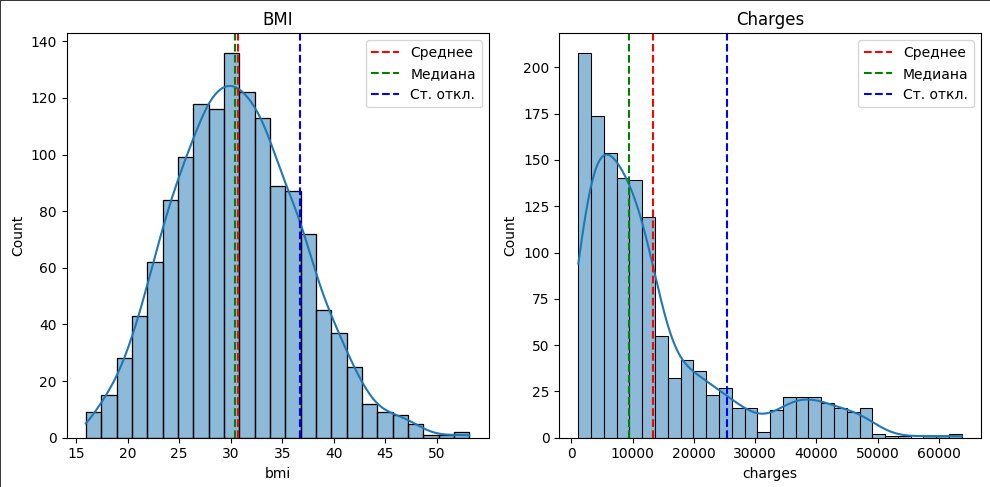
Средний уровень страховых взносов составляет около `13270` долларов с большим разбросом, что указывает на значительное разнообразие в стоимости страхования.

1. Найти меры центральной тенденции и меры разброса для индекса массы тела (bmi) и расходов (charges). Отобразить результаты в виде текста и на гистограммах (3 вертикальные линии). Добавить легенду на графики. Сделать выводы.

Листинг 4:

|  |
| --- |
| print("\n=== Задание 4: Меры центральной тенденции и разброса для bmi и charges ===")  bmi = insurance\_data['bmi']  charges = insurance\_data['charges']  # Расчет средней, медианы и стандартного отклонения  mean\_bmi = bmi.mean()  median\_bmi = bmi.median()  std\_bmi = bmi.std()  mean\_charges = charges.mean()  median\_charges = charges.median()  std\_charges = charges.std()  print(f"Среднее BMI: {mean\_bmi}, Медиана BMI: {median\_bmi}, Стандартное отклонение BMI: {std\_bmi}")  print(f"Среднее Charges: {mean\_charges}, Медиана Charges: {median\_charges}, Стандартное отклонение Charges: {std\_charges}")  # Построение гистограмм с добавлением линий  plt.figure(figsize=(10, 5))  plt.subplot(1, 2, 1)  sns.histplot(bmi, kde=True)  plt.axvline(mean\_bmi, color='r', linestyle='--', label='Среднее')  plt.axvline(median\_bmi, color='g', linestyle='--', label='Медиана')  plt.axvline(mean\_bmi + std\_bmi, color='b', linestyle='--', label='Ст. откл.')  plt.legend()  plt.title('BMI')  plt.subplot(1, 2, 2)  sns.histplot(charges, kde=True)  plt.axvline(mean\_charges, color='r', linestyle='--', label='Среднее')  plt.axvline(median\_charges, color='g', linestyle='--', label='Медиана')  plt.axvline(mean\_charges + std\_charges, color='b', linestyle='--', label='Ст. откл.')  plt.legend()  plt.title('Charges')  plt.tight\_layout()  plt.show() |





**Вывод**:

**Индекс массы тела (BMI)**

Средний BMI составляет `30.66`, медиана - `30.40`.

Наиболее часто встречающееся значение (мода) `BMI` равно `32.30`, большинство участников имеют избыточный вес.

Стандартное отклонение `6.10` и дисперсия `37.19` указывают на умеренное разнообразие значений.

**Страховые взносы**

Средний размер страховых взносов составляет `13270.42` долларов, в то время как медиана - `9382.03` долларов. Это говорит о том, что среднее значение значительно выше медианы, что может свидетельствовать о наличии некоторых участников с высокими страховками.

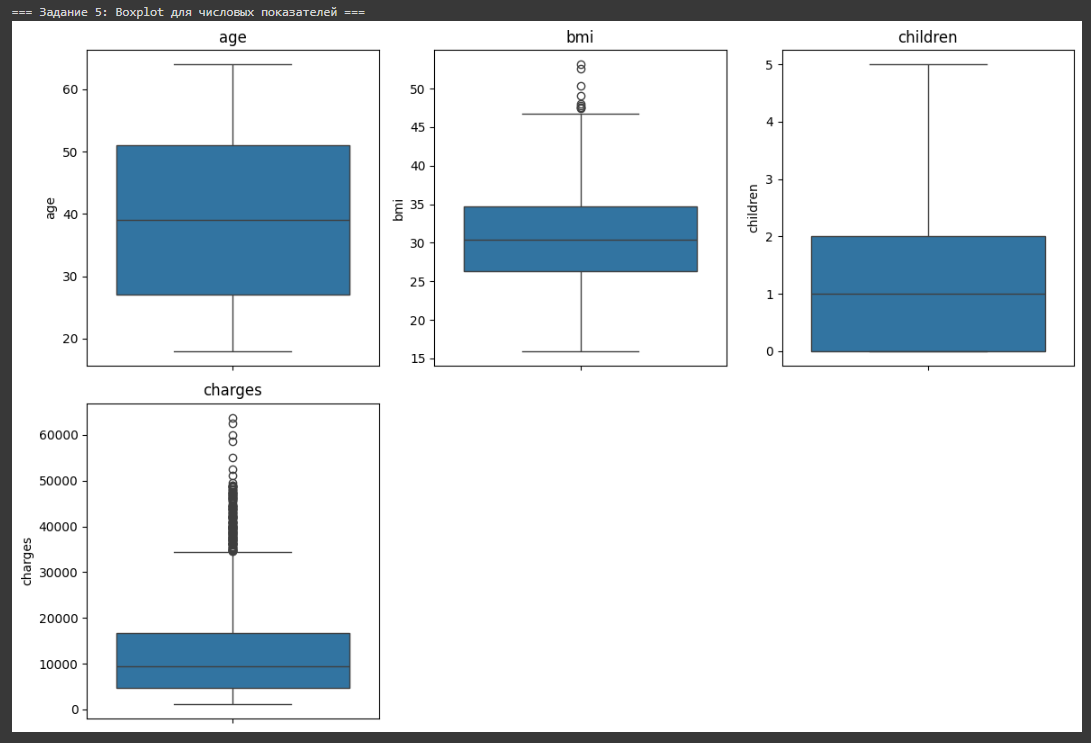
мода - `1639.56` долларов, что сильно ниже среднего значения. Это говорит о присутствие группы людей с низкими расходами на страхование.

Стандартное отклонение `12110.01` указывает на широкий разброс данных.

1. Построить box-plot для числовых показателей. Названия графиков должны соответствовать названиям признаков. Сделать выводы.

Листинг 5:

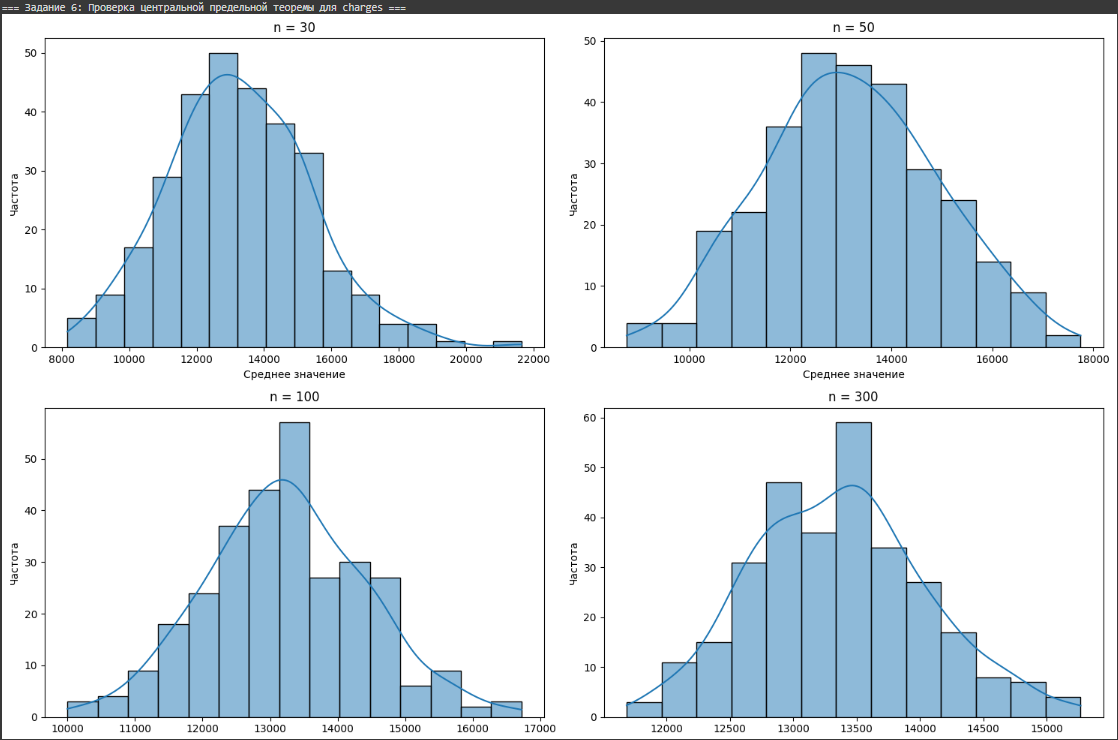
|  |
| --- |
| print("\n=== Задание 5: Boxplot для числовых показателей ===")  plt.figure(figsize=(12, 8))  for i, column in enumerate(insurance\_data.select\_dtypes(include=[np.number]).columns, 1):  plt.subplot(2, 3, i)  sns.boxplot(y=insurance\_data[column])  plt.title(column)  plt.tight\_layout()  plt.show() |



1. Используя признак charges или imb, проверить, выполняется ли центральная предельная теорема. Использовать различные длины выборок n. Количество выборок = 300. Вывести результат в виде гистограмм. Найти стандартное отклонение и среднее для полученных распределений. Сделать выводы

Листинг 6:

|  |
| --- |
| print("\n=== Задание 6: Проверка центральной предельной теоремы для charges ===")  sample\_means = []  n\_values = [30, 50, 100, 300]  for size in n\_values:  means = [np.mean(np.random.choice(charges, size)) for \_ in range(300)]  sample\_means.append(means)  # Построение гистограмм для каждой длины выборки  plt.figure(figsize=(15, 10))  for i, size in enumerate(n):  plt.subplot(2, 2, i+1)  sns.histplot(sample\_means[i], kde=True)  plt.title(f"n = {size}")  plt.xlabel('Среднее значение')  plt.ylabel('Частота')  plt.tight\_layout()  plt.show()  for i, n in enumerate(n\_values):  print(f'Длина выборки = {n}: Среднее = {np.mean(sample\_means[i])}, Станд.отклонение = {np.std(sample\_means[i])}') |



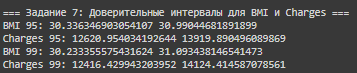


**Выводы:** ЦПТ подтверждается

1. Построить 95% и 99% доверительный интервал для среднего значения расходов и среднего значения индекса массы тела.

Листинг 7:

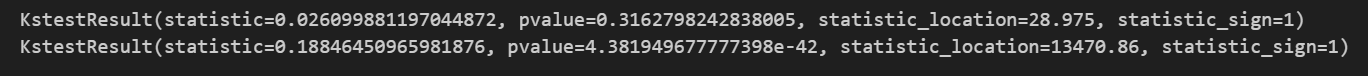
|  |
| --- |
| print("\n=== Задание 7: Доверительные интервалы для BMI и Charges ===")  n = len(bmi)  mean = bmi.mean()  stderr = bmi.std() / np.sqrt(n)  margin = stderr \* stats.t.ppf((1 + 0.95) / 2.0, n - 1)  print("BMI 95:", mean - margin, mean + margin)  n = len(charges)  mean = charges.mean()  stderr = charges.std() / np.sqrt(n)  margin = stderr \* stats.t.ppf((1 + 0.95) / 2.0, n - 1)  print("Charges 95:", mean - margin, mean + margin)  n = len(bmi)  mean = bmi.mean()  stderr = bmi.std() / np.sqrt(n)  margin = stderr \* stats.t.ppf((1 + 0.99) / 2.0, n - 1)  print("BMI 99:", mean - margin, mean + margin)  n = len(charges)  mean = charges.mean()  stderr = charges.std() / np.sqrt(n)  margin = stderr \* stats.t.ppf((1 + 0.99) / 2.0, n - 1)  print("Charges 99:", mean - margin, mean + margin) |

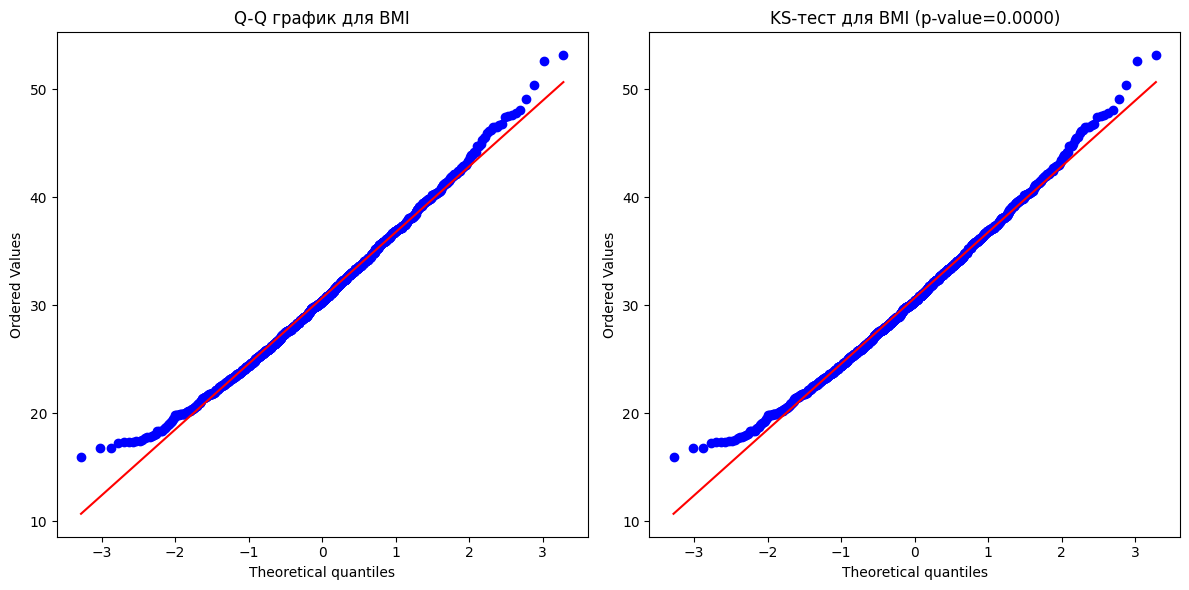


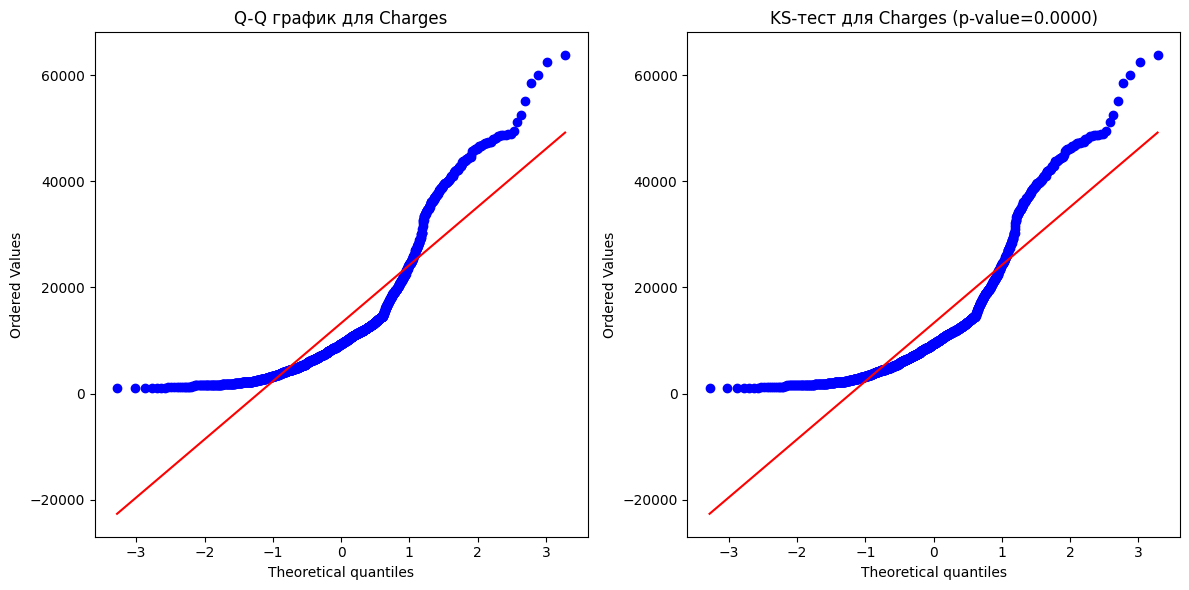
1. Проверить распределения следующих признаков на нормальность: индекс массы тела, расходы. Сформулировать нулевую и альтернативную гипотезы. Для каждого признака использовать KS-тест и q-q plot. Сделать выводы на основе полученных p-значений.

Листинг 8:

|  |
| --- |
| print("\n=== Задание 8: Проверка нормальности для BMI и Charges ===")  plt.figure(figsize=(12, 6))  plt.subplot(1, 2, 1)  stats.probplot(bmi, dist="norm", plot=plt)  plt.title('Q-Q график для BMI')  plt.tight\_layout()  plt.show()  plt.figure(figsize=(12, 6))  plt.subplot(1, 2, 1)  stats.probplot(charges, dist="norm", plot=plt)  plt.title('Q-Q график для Charges')  plt.tight\_layout()  plt.show()  # KS-тест  for feature in ['bmi', 'charges']:  stat = stats.kstest(insurance\_data[feature], 'norm', args=(insurance\_data[feature].mean(), insurance\_data[feature].std()))  print(stat)  #нулевая - распределение нормально  #альтернативная - распределение не нормально |







**Вывод**:

**Основная гипотеза (H0)**

Нет значительных различий/влияний в зависимости от рассматриваемых факторов на изучаемую величину (или переменную)

**Альтернативная гипотеза (H1)**

Существуют значительные различия/влияния в зависимости от рассматриваемых факторов на изучаемую величину (или переменную)

**Индекс массы тела**

Поскольку p-value (`0.316`) больше уровня значимости (`0.05`), мы не можем отвергнуть нулевую гипотезу. Распределение индекса массы тела можно считать нормальным

**Расходы**

Поскольку p-value (`0.000`) меньше уровня значимости (`0.05`), мы отвергаем нулевую гипотезу. Распределение расходов не является нормальным

1. Загрузить данные из файла “ECDCCases.csv”.

Листинг 10:

|  |
| --- |
| ecdccases\_data = pd.read\_csv('ECDCCases.csv') |

1. Проверить в данных наличие пропущенных значений. Вывести количество пропущенных значений в процентах. Удалить два признака, в которых больше всех пропущенных значений. Для оставшихся признаков обработать пропуски: для категориального признака использовать заполнение значением по умолчанию (например, «other»), для числового признака использовать заполнение медианным значением. Показать, что пропусков больше в данных нет.

Листинг 11:

|  |
| --- |
| print("\n=== Задание 10: Пропущенные значения в данных ===")  missing\_values = ecdccases\_data.isnull().sum()  percentage\_missing = (missing\_values / len(ecdccases\_data)) \* 100  print("Пропущенные значения: " + str(missing\_values))  print("Процент пропущенных значений: " + str(percentage\_missing))  most\_missing\_features = missing\_values.nlargest(2).index  ecdccases\_data.drop(columns=most\_missing\_features)  ecdccases\_data['countryterritoryCode'] = ecdccases\_data['countryterritoryCode'].fillna('other')  ecdccases\_data['popData2019'] = ecdccases\_data['popData2019'].fillna(ecdccases\_data['popData2019'].median())  print("Пустых значений нет:", ecdccases\_data.isnull().sum().sum() == 0) |

### 

### 

1. Посмотреть статистику по данным, используя describe(). Сделать выводы о том, какие признаки содержат выбросы. Посмотреть, для каких стран количество смертей в день превысило 3000 и сколько таких дней было.

Листинг 12:

|  |
| --- |
| print("\n=== Задание 11: Описание данных и отбор записей ===")  print(ecdccases\_data.describe())  over3000 = ecdccases\_data.loc[ecdccases\_data['deaths'] > 3000]  print(over3000['countriesAndTerritories'].value\_counts()) |

### 

1. Найти дублирование данных. Удалить дубликаты.

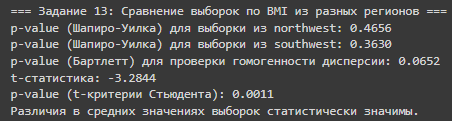
Листинг 13:

|  |
| --- |
| print("\n=== Задание 12: Поиск и удаление дубликатов ===")  dupl = ecdccases\_data.duplicated()  print(f'Количество одинаковых строк: { dupl.sum() }')  ecdccases\_data = ecdccases\_data.drop\_duplicates()  print(ecdccases\_data) |

1. Загрузить данные из файла “bmi.csv”. Взять оттуда две выборки. Одна выборка – это индекс массы тела людей c региона northwest, вторая выборка – это индекс массы тела людей с региона southwest. Сравнить средние значения этих выборок, используя t-критерий Стьюдента. Предварительно проверить выборки на нормальность (критерий ШопироУилка) и на гомогенность дисперсии (критерий Бартлетта).

Листинг 14:

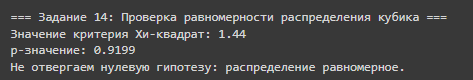
|  |
| --- |
| print("\n=== Задание 13: Сравнение выборок по BMI из разных регионов ===")  bmi\_data = pd.read\_csv('bmi.csv')  northwest\_bmi = bmi\_data[bmi\_data['region'] == 'northwest']['bmi']  southwest\_bmi = bmi\_data[bmi\_data['region'] == 'southwest']['bmi']  \_, p\_value\_northwest = stats.shapiro(northwest\_bmi)  \_, p\_value\_southwest = stats.shapiro(southwest\_bmi)  \_, p\_value\_bartlett = stats.bartlett(northwest\_bmi, southwest\_bmi)  print(f"p-value (Шапиро-Уилка) для выборки из northwest: {p\_value\_northwest:.4f}")  print(f"p-value (Шапиро-Уилка) для выборки из southwest: {p\_value\_southwest:.4f}")  print(f"p-value (Бартлетт) для проверки гомогенности дисперсии: {p\_value\_bartlett:.4f}")  if p\_value\_northwest > 0.05 and p\_value\_southwest > 0.05 and p\_value\_bartlett > 0.05:  t\_statistic, p\_value\_ttest = stats.ttest\_ind(northwest\_bmi, southwest\_bmi)  print(f"t-статистика: {t\_statistic:.4f}")  print(f"p-value (t-критерии Стьюдента): {p\_value\_ttest:.4f}")  if p\_value\_ttest < 0.05:  print("Различия в средних значениях выборок статистически значимы.")  else:  print("Нет статистически значимых различий в средних значениях выборок.")  else:  print("Условия для использования t-критерия Стьюдента не выполняются.") |



1. Кубик бросили 600 раз, получили следующие результаты: N Количество выпадений 1 97 2 98 3 109 4 95 5 97 6 104 С помощью критерия Хи-квадрат проверить, является ли полученное распределение равномерным. Использовать функцию scipy.stats.chisquare().

Листинг 15:

|  |
| --- |
| print("\n=== Задание 14: Проверка равномерности распределения кубика ===")  observed\_frequencies = np.array([97, 98, 109, 95, 97, 104])  expected\_frequencies = np.array([100] \* 6)  chi2\_statistic, p\_value = stats.chisquare(observed\_frequencies, expected\_frequencies)  print(f"Значение критерия Хи-квадрат: {chi2\_statistic:.2f}")  print(f"p-значение: {p\_value:.4f}")  alpha = 0.05  if p\_value < alpha:  print("Отвергаем нулевую гипотезу: распределение не является равномерным.")  else:  print("Не отвергаем нулевую гипотезу: распределение равномерное.") |



1. С помощью критерия Хи-квадрат проверить, являются ли переменные зависимыми. Создать датафрейм, используя следующий код:

data = pd.DataFrame({'Женат': [89,17,11,43,22,1], 'Гражданский брак': [80,22,20,35,6,4], 'Не состоит в отношениях': [35,44,35,6,8,22]}) data.index = ['Полный рабочий день','Частичная занятость','Временно не работает','На домохозяйстве','На пенсии','Учёба']

Использовать функцию scipy.stats.chi2\_contingency(). Влияет ли семейное положение на занятость?

Листинг 16:

|  |
| --- |
| print("\n=== Задание 15: Проверка зависимости семейного положения и занятости ===")  data = pd.DataFrame({  'Женат': [89, 17, 11, 43, 22, 1],  'Гражданский брак': [80, 22, 20, 35, 6, 4],  'Не состоит в отношениях': [35, 44, 35, 6, 8, 22]  })  data.index = ['Полный рабочий день', 'Частичная занятость', 'Временно не работает', 'На домохозяйстве', 'На пенсии', 'Учёба']  chi2\_statistic, p\_value, dof, expected = stats.chi2\_contingency(data)  print(f"Значение критерия Хи-квадрат: {chi2\_statistic:.2f}")  print(f"p-значение: {p\_value:.4f}") |

