



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«МИРЭА – Российский технологический университет»

РТУ МИРЭА

**Институт информационных технологий (ИИТ)
Кафедра прикладной математики (ПМ)**

ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ
по дисциплине «Технологии и инструментарий анализа больших данных»

Практическая работа № 4

Студент группы *ИКБО-01-22 Прокопчук Роман Олегович*

(подпись)

Ассистент *Тетерин Николай Николаевич*

(подпись)

Отчёт представлен «__» ноября 2025 г.

Москва, 2025 г.

ЦЕЛЬ

Ознакомиться с инструментами на Python для построения корреляции, линейной регрессии и выполнения дисперсионного анализа.

ХОД РАБОТЫ

Задание

1. Определить два вектора, представляющие собой число автомобилей, припаркованных в течении 5 рабочих дней у бизнес-центра на уличной стоянке и в подземном гараже.

День	Улица	Гараж
Понедельник	80	100
Вторник	98	82
Среда	75	105
Четверг	91	89
Пятница	78	102

1.1. Найти и интерпретировать корреляцию между переменными «Улица» и «Гараж» (подсчитать корреляцию по Пирсону).

1.2. Построить диаграмму рассеяния для вышеупомянутых переменных.

2. Найти и выгрузить данные. Вывести, провести предобработку и описать признаки.

2.1. Построить корреляционную матрицу по одной целевой переменной. Определить наиболее коррелирующую переменную, продолжить с ней работу в следующем пункте.

2.2. Реализовать регрессию вручную, отобразить наклон, сдвиг и MSE.

2.3. Визуализировать регрессию на графике.

3. Загрузить данные: 'insurance.csv'. Вывести и провести предобработку. Вывести список уникальных регионов.

3.1. Выполнить однофакторный ANOVA тест, чтобы проверить влияние региона на индекс массы тела (BMI), используя первый способ, через библиотеку Scipy.

3.2. Выполнить однофакторный ANOVA тест, чтобы проверить влияние региона на индекс массы тела (BMI), используя второй способ, с помощью функции `anova_lm()` из библиотеки `statsmodels`.

3.3. С помощью *t* критерия Стьюдента перебрать все пары. Определить поправку Бонферрони. Сделать выводы.

3.4. Выполнить пост-хок тесты Тьюки и построить график.

3.5. Выполнить двухфакторный ANOVA тест, чтобы проверить влияние региона и пола на индекс массы тела (BMI), используя функцию `anova_lm()` из библиотеки `statsmodels`.

3.6. Выполнить пост-хок тесты Тьюки и построить график.

Ход работы

Код для выполнения задания 1 представлен на рисунке 1. Результат его работы – на рисунке 2, построенная диаграмма – на рисунке 3.

```
import matplotlib.pyplot as plt
import pandas as pd

data = {
    'День': ['Понедельник', 'Вторник', 'Среда', 'Четверг', 'Пятница'],
    'Улица': [80, 98, 75, 91, 78],
    'Гараж': [100, 82, 105, 89, 102]
}

df = pd.DataFrame(data)

print("--- Исходные данные: ---")
print(df)
print("\n")

correlation = df['Улица'].corr(df['Гараж'])
abs_corr = abs(correlation)

print(f"Значение r = {correlation:.4f} (по модулю {abs_corr:.4f}).")

plt.figure(figsize=(8, 6))
plt.scatter(df['Улица'], df['Гараж'], color='crimson', marker='o')

plt.title('Диаграмма рассеяния: Уличная стоянка vs. Гараж')
plt.xlabel('Число автомобилей на уличной стоянке')
plt.ylabel('Число автомобилей в подземном гараже')

plt.grid(True)

plt.show()
```

Рисунок 1 – Код для выполнения задания 1

```
Run 1_correlation_and_diagram x
C:\Users\Remsely\dev\mirea\python\big-da
--- Исходные данные: ---
      День  Улица  Гараж
0  Понедельник    80   100
1    Вторник    98    82
2     Среда    75   105
3   Четверг    91    89
4   Пятница    78   102

Значение r = -1.0000 (по модулю 1.0000).

Process finished with exit code 0
```

Рисунок 2 – Результат выполнения кода для задания 1

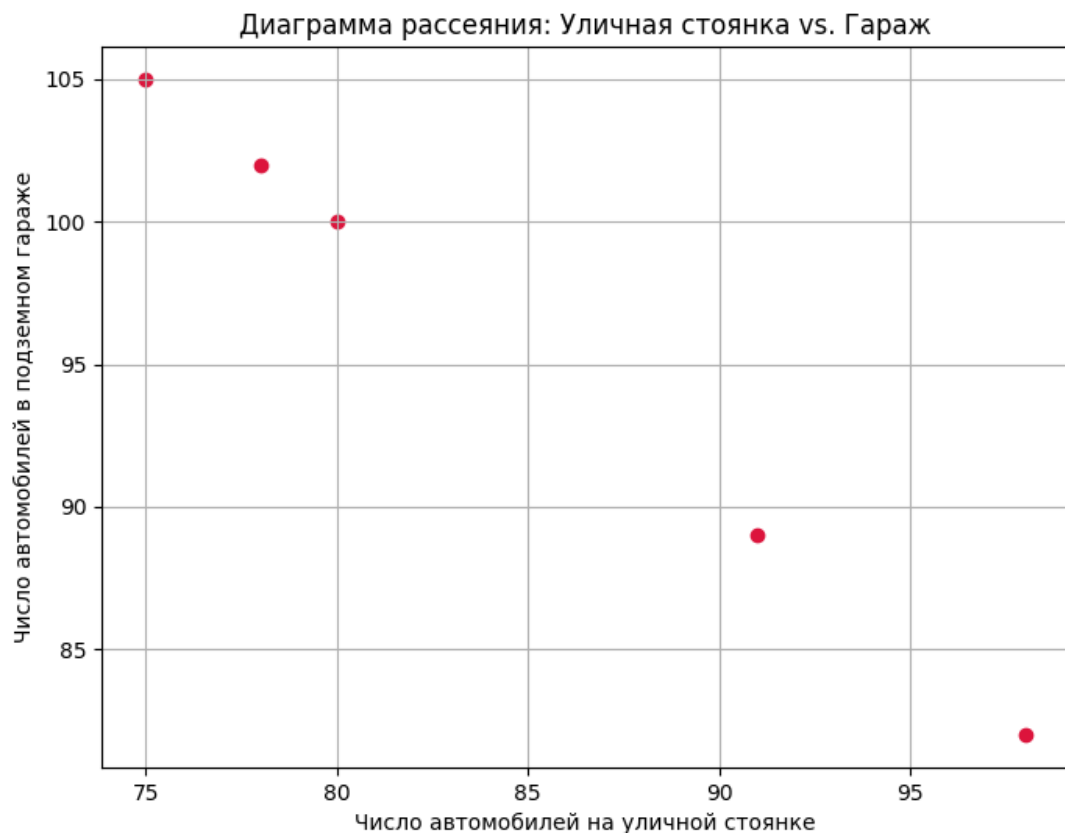


Рисунок 3 – Построенная для задания 1 диаграмма

Вывод: наблюдается идеальная отрицательная линейная зависимость – при росте числа машин на улице число машин в гараже падает на такую же величину.

Код для выполнения задания 2 представлен на рисунках 4-6. Результат его работы – на рисунке 7, построенная диаграмма – на рисунке 8.

```

import matplotlib.pyplot as plt
import numpy as np
import pandas as pd

df = pd.read_csv('data/insurance.csv')

print("--- Данные 'insurance.csv' ---")
print(df.head())
print("\n")

print(f"--- Пропущенные значения: ---\n{df.isnull().sum()}\n")

df['sex'] = df['sex'].map({'female': 0, 'male': 1})
df['smoker'] = df['smoker'].map({'no': 0, 'yes': 1})

df = pd.get_dummies(df, columns=['region'], drop_first=True)

print("--- Данные после предобработки (текст в числа): ---")
print(df.head())
print("\n")

# --- Задание 2.1 ---
corr_matrix = df.corr()

target_corr = corr_matrix['charges'].sort_values(ascending=False)

print(target_corr)
print("\n")

most_correlated_var = target_corr.index[1]
print(f"Наиболее коррелирующая переменная (кроме самой 'charges'): '{most_correlated_var}'")

```

Рисунок 4 – Код для преобразования данных и выполнения пункта 2.1

```

# --- Задание 2.2: Реализация регрессии вручную (Градиентный спуск) ---
X = np.array(df[most_correlated_var])
y = np.array(df['charges'])

def mserror(X, y, w1, w0): 1 usage new *
    y_pred = w1 * X + w0
    n = len(y)
    return np.sum((y - y_pred) ** 2) / n

def gr_mserror(X, y, w1, w0): 1 usage new *
    y_pred = w1 * X + w0
    n = len(y)

    grad_w1 = (-2 / n) * np.sum(X * (y - y_pred))
    grad_w0 = (-2 / n) * np.sum(y - y_pred)

    return grad_w1, grad_w0

w1 = 0.0
w0 = 0.0
learning_rate = 0.1
n_iterations = 10000
eps = 0.0001

for i in range(n_iterations):
    grad_w1, grad_w0 = gr_mserror(X, y, w1, w0)

    next_w1 = w1 - learning_rate * grad_w1
    next_w0 = w0 - learning_rate * grad_w0

    if abs(next_w1 - w1) < eps and abs(next_w0 - w0) < eps:
        print(f"\nАлгоритм сошелся на итерации {i}")
        break

    w1, w0 = next_w1, next_w0

final_mse = mserror(X, y, w1, w0)

print("\n--- Результаты ручной регрессии: ---")
print(f"Наклон (w1): {w1:.2f}")
print(f"Сдвиг (w0): {w0:.2f}")
print(f"MSE (среднеквадратичная ошибка): {final_mse:.2f}")
print("\n")

```

Рисунок 5 – Код для выполнения пункта 2.2

```

# --- Задание 2.3: Визуализация регрессии ---
plt.figure(figsize=(10, 6))
plt.scatter(X, y, alpha=0.3, label='Исходные данные')

X_line = np.array([0, 1])
y_line = w1 * X_line + w0

plt.plot(*args=X_line, y_line, color='red', linewidth=3,
         label=f'Модель: y = {w1:.2f}*x + {w0:.2f}')

plt.title(f'Линейная регрессия: {most_correlated_var} vs. charges')
plt.xlabel(most_correlated_var.capitalize())
plt.ylabel('Charges (Стоимость страховки)')
plt.legend()
plt.grid(True)
plt.show()

```

Рисунок 6 – Код для выполнения пункта 2.3

Run 2_manual_linear_regression

```

C:\Users\Remsely\dev\mirea\python\big-data\.venv\Scripts\python.exe C:\Users\Re
--- Данные 'insurance.csv' ---
   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

--- Пропущенные значения: ---
age      0
sex      0
bmi      0
children 0
smoker   0
region   0
charges  0
dtype: int64

--- Данные после предобработки (текст в числа): ---
   age  sex    bmi  ...  region_northwest  region_southeast  region_southwest
0   19    0  27.900  ...                False                False                True
1   18    1  33.770  ...                False                 True                False
2   28    1  33.000  ...                False                 True                False
3   33    1  22.705  ...                 True                False                False
4   32    1  28.880  ...                 True                False                False

[5 rows x 9 columns]

charges      1.000000
smoker       0.787251
age          0.299008
bmi          0.198341
region_southeast  0.073982
children     0.067998
sex          0.057292
region_northwest -0.039905
region_southwest -0.043210
Name: charges, dtype: float64

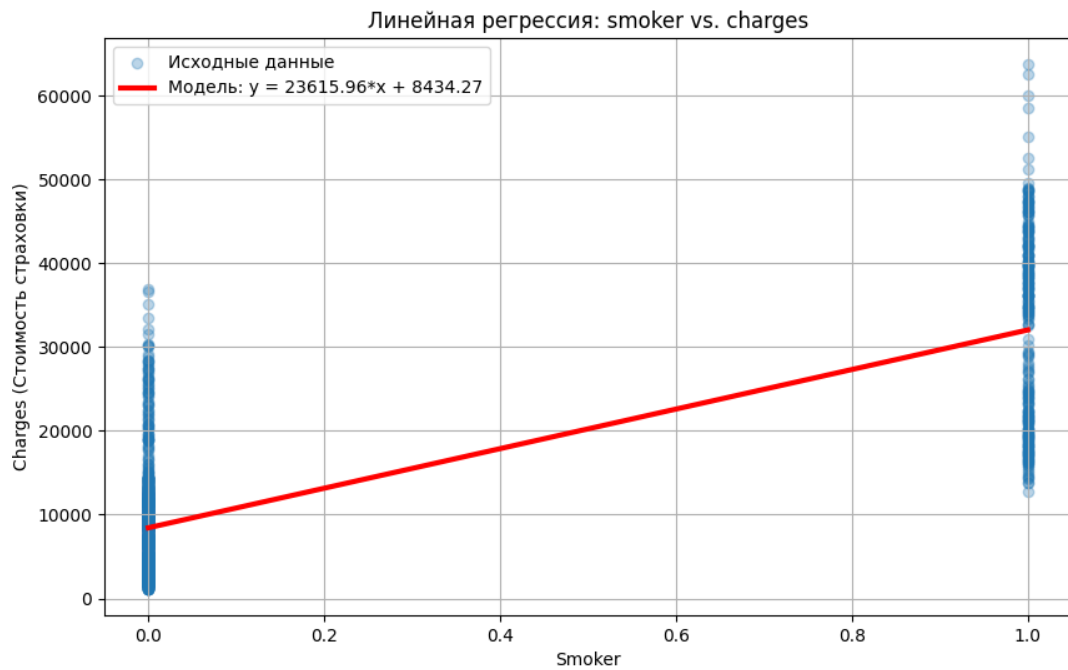
Наиболее коррелирующая переменная (кроме самой 'charges'): 'smoker'

Алгоритм сошелся на итерации 497

--- Результаты ручной регрессии: ---
Наклон (w1): 23615.96
Сдвиг (w0): 8434.27
MSE (среднеквадратичная ошибка): 55720715.95

```

Рисунок 7 – Результат работы кода для задания 2



```
import warnings
```

1

```
import matplotlib.pyplot as plt
```

```
import pandas as pd
```

```
import scipy.stats as stats
```

```
import statsmodels.api as sm
```

```
from statsmodels.formula.api import ols
```

```
from statsmodels.stats.multicomp import pairwise_tukeyhsd
```

```
warnings.filterwarnings('ignore')
```

```
df = pd.read_csv('data/insurance.csv')
```

```
print("--- Данные 'insurance.csv' ---")
```

```
print(df.head())
```

```
print("\n")
```

```
unique_regions = df['region'].unique()
```

```
print(f"--- Уникальные регионы в датасете: {unique_regions} ---")
```

```
print("\n")
```

```
# --- Задание 3.1 ---
```

```
regions = df['region'].unique()
```

```
bmi_by_region = []
```

```
for region in regions:
```

```
    bmi_by_region.append(df[df['region'] == region]['bmi'])
```

```
f_stat, p_value = stats.f_oneway(*bmi_by_region)
```

```
print(f"F-статистика: {f_stat:.4f}")
```

```
print(f"P-значение: {p_value:.4f}")
```

```
alpha = 0.05
```

```
if p_value < alpha:
```

```
    print(f"Вывод: P-значение ({p_value:.4f}) < {alpha}.")
```

```
    print("-> Существуют статистически значимые различия в BMI между регионами.")
```

```
else:
```

```
    print(f"Вывод: P-значение ({p_value:.4f}) > {alpha}, не можем отклонить нулевую гипотезу.")
```

```
    print("-> Статистически значимых различий в BMI между регионами не найдено.")
```

```
print("\n")
```

Рисунок 9 – Код для выполнения пункта 3.1

```

# --- Задание 3.2 ---
model = ols(formula='bmi ~ C(region)', data=df).fit()
anova_table = sm.stats.anova_lm(*args=model, type=2)

print(anova_table)

# --- Задание 3.3 ---
pairs = []
for i in range(len(regions)):
    for j in range(i + 1, len(regions)):
        pairs.append((regions[i], regions[j]))

print(f"Всего {len(pairs)} пар для сравнения.")

alpha = 0.05
bonferroni_alpha = alpha / len(pairs)
print(f"Уровень значимости (alpha): {alpha}")
print(f"Новый уровень значимости (поправка Бонферрони): {alpha} / {len(pairs)} = {bonferroni_alpha:.4f}\n")

print("Результаты t-тестов:")
for r1, r2 in pairs:
    group1 = df[df['region'] == r1]['bmi']
    group2 = df[df['region'] == r2]['bmi']

    t_stat, p_value = stats.ttest_ind(group1, group2)

    print(f"   Пара: {r1} vs {r2}")
    print(f"   P-значение: {p_value:.4f}")

    if p_value < bonferroni_alpha:
        print(f"       Вывод: ЗНАЧИМО ({p_value:.4f} < {bonferroni_alpha:.4f})")
    else:
        print(f"       Вывод: НЕ ЗНАЧИМО ({p_value:.4f} > {bonferroni_alpha:.4f})")
print("\n")

```

Рисунок 10 – Код для выполнения пунктов 3.2 и 3.3

```

# --- Задание 3.4 ---
print("Задание 3.4\n")

tukey_result = pairwise_tukeyhsd(
    endog=df['bmi'],
    groups=df['region'],
    alpha=0.05
)

print(tukey_result)

print("Построение графика для теста Тьюки...")
tukey_result.plot_simultaneous()
plt.title("Тест Тьюки (Tukey HSD) для BMI по Регионам")
plt.show()

```

Рисунок 11 – Код для выполнения пункта 3.4

```
# --- Задание 3.5 ---
model_2way = ols( formula: 'bmi ~ C(region) + C(sex) + C(region):C(sex)', data=df).fit()
anova_table_2way = sm.stats.anova_lm( *args: model_2way, type=2)

print(anova_table_2way)

print("\n--- Выводы по двухфакторному ANOVA: ---")
p_region = anova_table_2way.loc['C(region)', 'PR(>F)']
p_sex = anova_table_2way.loc['C(sex)', 'PR(>F)']
p_interaction = anova_table_2way.loc['C(region):C(sex)', 'PR(>F)']

if p_region < 0.05:
    print(f"1. 'region': {p_region:.4f} (< 0.05) - ЗНАЧИМО. Регион влияет на BMI.")
else:
    print(f"1. 'region': {p_region:.4f} (> 0.05) - НЕ ЗНАЧИМО.")

if p_sex < 0.05:
    print(f"2. 'sex': {p_sex:.4f} (< 0.05) - ЗНАЧИМО. Пол влияет на BMI.")
else:
    print(f"2. 'sex': {p_sex:.4f} (> 0.05) - НЕ ЗНАЧИМО.")

if p_interaction < 0.05:
    print(f"3. 'interaction': {p_interaction:.4f} (< 0.05) - ЗНАЧИМО. Эффект региона зависит от пола.")
else:
    print(f"3. 'interaction': {p_interaction:.4f} (> 0.05) - НЕ ЗНАЧИМО.")
print("\n")
```

Рисунок 12 – Код для выполнения пункта 3.5

```
# --- Задание 3.6 ---
df['region_sex_group'] = df['region'] + '_' + df['sex']

print("--- Новые группы для теста (первые 5): ---")
print(df['region_sex_group'].head())
print("\n")

tukey_2way_result = (
    pairwise_tukeyhsd(
        endog=df['bmi'],
        groups=df['region_sex_group'],
        alpha=0.05
    )
)

print(tukey_2way_result)

print("--- Построение графика для теста Тьюки... ---")
tukey_2way_result.plot_simultaneous()
plt.title("Тест Тьюки (Tukey HSD) для BMI по (Регион + Пол)")
plt.show()
```

Рисунок 13 – Код для выполнения пункта 3.6

```
Run 3_variance_analysis x
C:\Users\Remsely\dev\mirea\python\big-data\.venv\Scripts\python.exe C:\Users\Remsely\dev\
--- Данные 'insurance.csv' ---
   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

--- Уникальные регионы в датасете: ['southwest' 'southeast' 'northwest' 'northeast'] ---

Задание 3.1

F-статистика: 39.4951
P-значение: 0.0000
Вывод: P-значение (0.0000) < 0.05.
-> Существуют статистически значимые различия в BMI между регионами.

Задание 3.2

           df      sum_sq    mean_sq      F      PR(>F)
C(region)   3.0   4055.880631  1351.960210  39.495057  1.881839e-24
Residual  1334.0  45664.319755   34.231124      NaN      NaN

Задание 3.3

Всего 6 пар для сравнения.
Уровень значимости (alpha): 0.05
Новый уровень значимости (поправка Бонферрони): 0.05 / 6 = 0.0083

Результаты t-тестов:
Пара: southwest vs southeast
P-значение: 0.0000
Вывод: ЗНАЧИМО (0.0000 < 0.0083)
Пара: southwest vs northwest
P-значение: 0.0011
Вывод: ЗНАЧИМО (0.0011 < 0.0083)
Пара: southwest vs northeast
P-значение: 0.0019
Вывод: ЗНАЧИМО (0.0019 < 0.0083)
Пара: southeast vs northwest
P-значение: 0.0000
Вывод: ЗНАЧИМО (0.0000 < 0.0083)
Пара: southeast vs northeast
P-значение: 0.0000
Вывод: ЗНАЧИМО (0.0000 < 0.0083)
Пара: northwest vs northeast
P-значение: 0.9519
Вывод: НЕ ЗНАЧИМО (0.9519 > 0.0083)
```

Рисунок 14 – Результат работы программы для пунктов 3.1-3.3

```
Run 3_variance_analysis x
```

```
↑ Задание 3.4
↓
Multiple Comparison of Means - Tukey HSD, FWER=0.05
=====
group1 group2 meandiff p-adj lower upper reject
-----
northeast northwest 0.0263 0.9999 -1.1552 1.2078 False
northeast southeast 4.1825 0.0 3.033 5.332 True
northeast southwest 1.4231 0.0107 0.2416 2.6046 True
northwest southeast 4.1562 0.0 3.0077 5.3047 True
northwest southwest 1.3968 0.0127 0.2162 2.5774 True
southeast southwest -2.7594 0.0 -3.9079 -1.6108 True
-----

Построение графика для теста Тьюки...
Задание 3.5

df sum_sq mean_sq F PR(>F)
C(region) 3.0 4055.880631 1351.960210 39.602259 1.636858e-24
C(sex) 1.0 86.007035 86.007035 2.519359 1.126940e-01
C(region):C(sex) 3.0 174.157808 58.052603 1.700504 1.650655e-01
Residual 1330.0 45404.154911 34.138462 NaN NaN

--- Выводы по двухфакторному ANOVA: ---
1. 'region': 0.0000 (< 0.05) - ЗНАЧИМО. Регион влияет на BMI.
2. 'sex': 0.1127 (> 0.05) - НЕ ЗНАЧИМО.
3. 'interaction': 0.1651 (> 0.05) - НЕ ЗНАЧИМО.
```

Рисунок 15 – Результат работы программы для пунктов 3.4-3.5

Run 3_variance_analysis x

↶

■

⋮

↑

↓

⇌

⇅

📄

🗑

Задание 3.6

```

--- Новые группы для теста (первые 5): ---
0    southwest_female
1    southeast_male
2    southeast_male
3    northwest_male
4    northwest_male
Name: region_sex_group, dtype: object

Multiple Comparison of Means - Tukey HSD, FWER=0.05
=====
  group1      group2  meandiff p-adj  lower  upper  reject
-----
northeast_female  northeast_male -0.2998 0.9998 -2.2706  1.6711  False
northeast_female  northwest_female -0.0464  1.0 -2.0142  1.9215  False
northeast_female  northwest_male -0.2042  1.0 -2.1811  1.7728  False
northeast_female  southeast_female  3.3469  0.0  1.41  5.2839  True
northeast_female  southeast_male  4.6657  0.0  2.7634  6.568  True
northeast_female  southwest_female  0.7362 0.9497 -1.2377  2.71  False
northeast_female  southwest_male  1.8051 0.1007 -0.1657  3.776  False
northeast_male  northwest_female  0.2534 0.9999 -1.7083  2.2152  False
northeast_male  northwest_male  0.0956  1.0 -1.8752  2.0665  False
northeast_male  southeast_female  3.6467  0.0  1.7159  5.5775  True
northeast_male  southeast_male  4.9655  0.0  3.0695  6.8614  True
northeast_male  southwest_female  1.036 0.7515 -0.9318  3.0037  False
northeast_male  southwest_male  2.1049 0.0258  0.1402  4.0697  True
northwest_female  northwest_male -0.1578  1.0 -2.1257  1.81  False
northwest_female  southeast_female  3.3933  0.0  1.4656  5.321  True
northwest_female  southeast_male  4.712  0.0  2.8192  6.6049  True
northwest_female  southwest_female  0.7825 0.9294 -1.1822  2.7473  False
northwest_female  southwest_male  1.8515 0.0806 -0.1103  3.8132  False
northwest_male  southeast_female  3.5511  0.0  1.6141  5.4881  True
northwest_male  southeast_male  4.8698  0.0  2.9676  6.7721  True
northwest_male  southwest_female  0.9403 0.8354 -1.0335  2.9142  False
northwest_male  southwest_male  2.0093 0.042  0.0385  3.9801  True
southeast_female  southeast_male  1.3187 0.3823 -0.542  3.1795  False
southeast_female  southwest_female -2.6108 0.0011 -4.5446 -0.6769  True
southeast_female  southwest_male -1.5418 0.2304 -3.4726  0.389  False
southeast_male  southwest_female -3.9295  0.0 -5.8286 -2.0304  True
southeast_male  southwest_male -2.8606 0.0001 -4.7565 -0.9646  True
southwest_female  southwest_male  1.069 0.7201 -0.8988  3.0367  False
=====
--- Построение графика для теста Тьюки... ---

```

Рисунок 16 – Результат работы программы для пункта 3.6

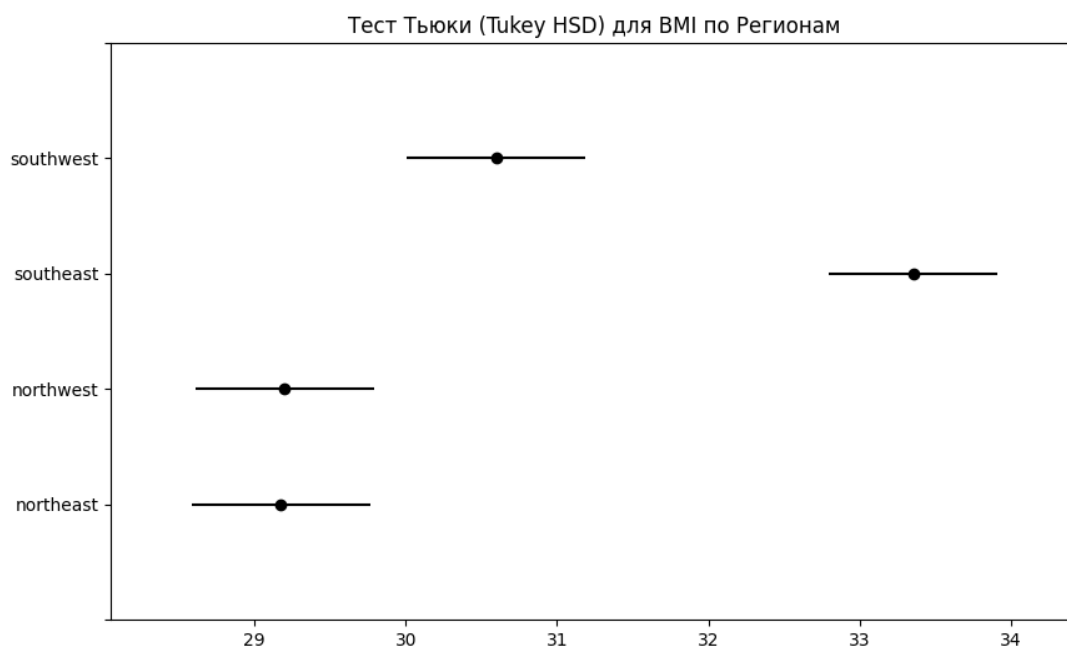


Рисунок 17 – Построенный график для BMI по регионам

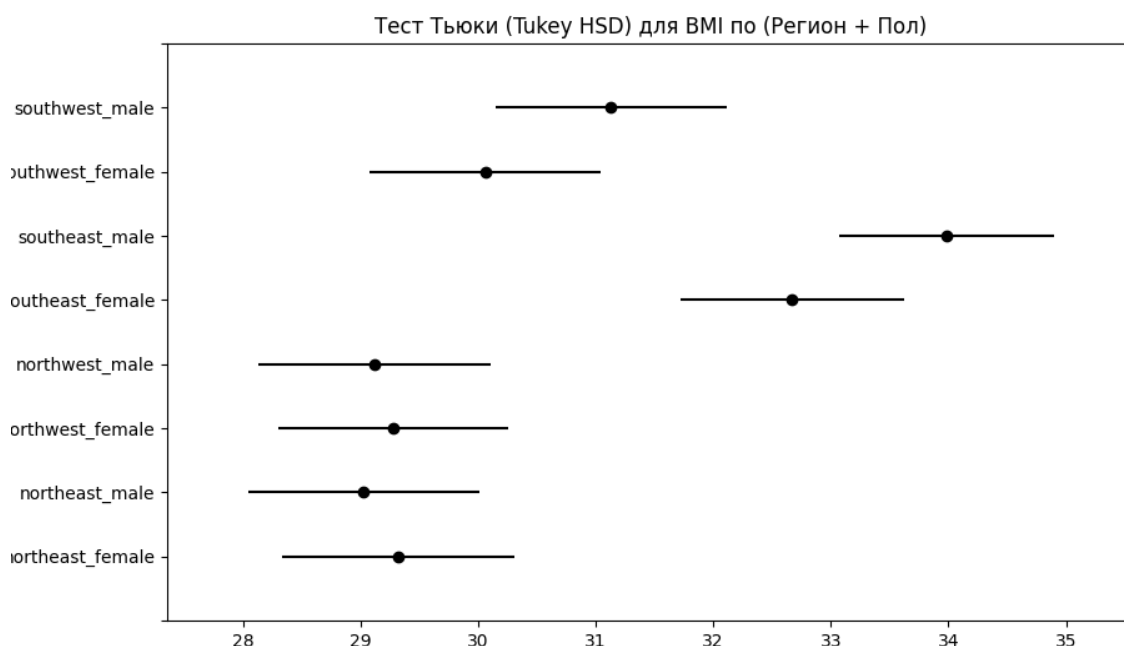


Рисунок 18 – Построенный график для BMI по регионам и полу

Выводы:

- Между регионами существуют статистически значимые различия в BMI.
- Фактор region значим ($p < 0.05$).
- Все пары регионов значимы, кроме northwest vs northeast. Ранг средних BMI: southeast < southwest < (northwest \approx northeast).
- Тест Тьюки подтверждает – southeast ниже всех; southwest выше southeast, но ниже northwest и northeast; northwest \approx northeast (различий нет).

- Двухфакторный ANOVA: region значим, sex не значим, взаимодействие region:sex не значимо.

- Пост-хок по группам region+sex: различия формируются регионом; внутри каждого региона различий между полами нет. Сочетания с southeast сильно отличаются от большинства других групп.

ВЫВОД

В ходе выполненной практической работы проведено ознакомление с инструментами на Python для построения корреляции, линейной регрессии и выполнения дисперсионного анализа.