МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«Дальневосточный федеральный университет»
(ДВФУ)

ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ

Лабораторная работа №9
Линейная регрессия

Дисциплина «Теория вероятностей и математическая статистика»

Студент группы Б9123-01.03.02ии
Моттуева Уруйдана Михайловна

г. Владивосток

2025

Целью данной лабораторной работы является построение модели линейной регрессии.

Уравнение регрессии
Имеется m признаков $X = (X_1, X_2, \dots, X_m)^T$ и зависящей от них целевой признак Y.
Уравнением регрессии Y на X называется уравнение

$$Y(x) = E(Y|X = \mathrm{x}) + \epsilon,$$

где $\epsilon \sim N(0; \sigma^2)$ — случайный остаток(ошибка).

Линейная регрессия
Если условное математическое ожидание $E(Y|X = \mathrm{x})$ является линейной функцией:

$$E(Y|X = \mathrm{x}) = a_1x_1 + a_2x_2 + \dots + a_mx_m = a^Tx.$$

То *уравнение линейной регрессии* выглядит как:

$$Y(x) = \mathrm{a}^Tx + \epsilon.$$

Модель определяется параметрами $a = (a_1, a_2, \dots, a_m)^T$, которые оцениваются с помощью выборки при условии $D(\epsilon) \to min$

Имеется выборка $\left(X_1^i, X_2^i, \dots, X_m^i, Y_i\right)$, $i = \overline{1, n}$ из $(X_1, X_2, \dots, X_m, Y)$. Тогда, подставляя значения в уравнение линейной регрессии, получаем

$$Y_i = \mathrm{a}^TX^i + \epsilon_i.$$

Оценкой $D(\epsilon)$ является

$$\frac{1}{n}\sum_{i=1}^{n} \epsilon_i^2 \to min.$$

Оптимальные значения a, минимизирующие сумму квадратов отклонений, находятся как:

$$\hat{\mathrm{a}} = \arg\min_{\mathrm{a}} \sum_{i=1}^{n} \left(a^TX^i - Y_i\right)^2.$$

Оценки $\hat{a}$ параметров a являются несмещёнными, состоятельными и с наименьшей дисперсией при соблюдении условий *теоремы Гаусса-Маркова*.

Теорема Гаусса-Маркова утверждает, что оценка МНК (метода наименьших квадратов) является наилучшей линейной несмещенной оценкой (BLUE), если:

1. Ошибки распределены нормально: $\epsilon_i \sim N(0; \sigma^2)$;

Данное условие проверяется с помощью одновыборочного t-теста для гипотезы:
$H_0: E(\varepsilon_i) = 0;$
$H_1: E(\varepsilon_i) \neq 0.$

2. Нет автокорреляции между ошибками: $\forall j < i \ \mathrm{Cov}\left(\epsilon_j, \epsilon_i\right) = 0$;

Для проверки автокорреляции использовался тест Дарбина–Уотсона, который позволяет выявить наличие линейной автокорреляции первого порядка. Интерпретация статистики:

$DW \approx 2$ — автокорреляции нет,
$DW < 1.5$ — есть положительная автокорреляция,
$DW > 2.5$ — есть отрицательная автокорреляция.

3. Ранг матрицы X равен числу факторов: $\operatorname{rang} X = m$;

4. Ошибки имеют постоянную дисперсию: $Var(\epsilon_i) = \sigma^2$
Для её проверки использовался тест Бреуша–Пагана:
H_0: дисперсия остатков постоянна
H_1: дисперсия остатков изменяется

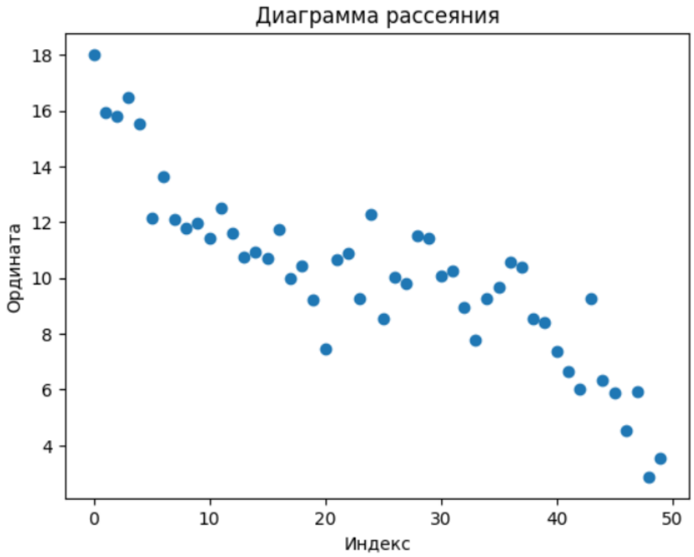
5. Математическое ожидание ошибок равно нулю: $E(\varepsilon_i) = 0$.

Примечание: нормальность остатков **не требуется** для теоремы Гаусса-Маркова, но необходима для проверки значимости коэффициентов (t-тесты, F-тесты).

Реализация

Линейная модель: $y = \beta_0 + \beta_1 x$

1. Построить диаграмму рассеяния



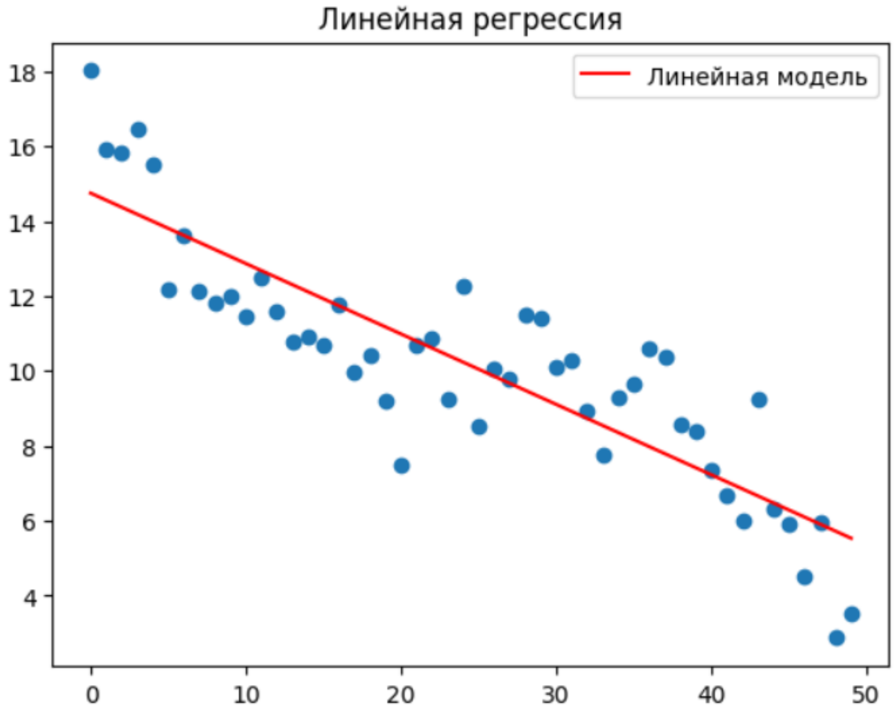
2. Построить модель линейной регрессии с помощью statsmodels.api.OLS взяв в качестве условного матожидания линейную функцию.

```
X = sm.add_constant(x)
model = sm.OLS(y, X).fit()
print(model.summary())

plt.scatter(x, y)
plt.plot(x, model.predict(X), color='red', label='Линейная модель')
plt.title('Линейная регрессия')
plt.legend()
plt.show()
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.757
Model:                            OLS   Adj. R-squared:                  0.752
Method:                 Least Squares   F-statistic:                     149.6
Date:                Mon, 26 May 2025   Prob (F-statistic):           2.34e-16
Time:                        00:16:16   Log-Likelihood:                -92.445
No. Observations:                  50   AIC:                             188.9
Df Residuals:                      48   BIC:                             192.7
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         14.7473      0.437     33.733      0.000      13.868      15.626
x1            -0.1880      0.015    -12.230      0.000      -0.219      -0.157
==============================================================================
Omnibus:                        1.246   Durbin-Watson:                   0.993
Prob(Omnibus):                  0.536   Jarque-Bera (JB):                1.067
Skew:                           0.146   Prob(JB):                        0.587
Kurtosis:                       2.347   Cond. No.                         56.1
==============================================================================
```

$R^2 = 0.757$, коэффициенты значимы ($p - value < 0.05$)

3. На диаграмму рассеяния добавить линию регрессии.



1. Оценить значимость коэффициентов. Проверить положения теоремы Гаусса-Маркова.

```
def gauss_markov(model, X, flag=False):
    residuals = model.resid

    # 1. Нормальность остатков (дополнительная проверка)
    stat_shapiro, p_shapiro = shapiro(residuals)
    normal_ok = p_shapiro > 0.05

    # 2. Автокорреляция (Дарбин-Уотсон)
    dw_stat = durbin_watson(residuals)
    autocorr_ok = 1.5 < dw_stat < 2.5

    # 3. Полный ранг матрицы X
    rank_X = la.matrix_rank(X)
    num_params = X.shape[1]
    rank_ok = (rank_X == num_params)

    # 4. Гомоскедастичность (Бреуш-Паган) постоянность дисперсии
    try:
        bp_test = het_breuschpagan(residuals, X)
        p_bp = bp_test[1]
        homoskedasticity_ok = p_bp > 0.05
    except:
        homoskedasticity_ok = True
        p_bp = np.nan

    # 5. Нулевое матожидание остатков
    t_stat, p_zero_mean = ttest_1samp(residuals, 0)
    mean_ok = p_zero_mean > 0.05
```

```python
    if flag:
        print("\n1. Нормальность остатков (Шапиро-Уилк):")
        print(f"    p-value = {p_shapiro:.3f}",
              "Нормальны" if normal_ok else "Не нормальны")

        print("\n2. Автокорреляция (Дарбин-Уотсон):")
        print(f"    DW = {dw_stat:.3f}",
              "Нет автокорреляции" if autocorr_ok else "Есть автокорреляция")

        print("\n3. Ранг матрицы X:")
        print(f"    Ранг = {rank_X}, Параметры = {num_params}",
              "Полный ранг" if rank_ok else "Неполный ранг")

        print("\n4. Нулевое среднее остатков:")
        print(f"    p-value = {p_zero_mean:.3f}",
              "E[ε]=0" if mean_ok else "E[ε]≠0")

        print("\n5. Гомоскедастичность (Бреуш-Паган):")
        print(f"    p-value = {p_bp:.3f}",
              "Дисперсия постоянна" if homoskedasticity_ok else "Гетероскедастичность")

        print("\nИтог:")
        print("Теорема Гаусса-Маркова выполнена" if all(
            [autocorr_ok, rank_ok, mean_ok, homoskedasticity_ok]
        ) else "Теорема Гаусса-Маркова нарушена")

    return {
        'normality': normal_ok,
        'autocorrelation': autocorr_ok,
        'full_rank': rank_ok,
        'zero_mean': mean_ok,
        'homoskedasticity': homoskedasticity_ok
    }
```

```python
gauss_markov(model, X, True)
```

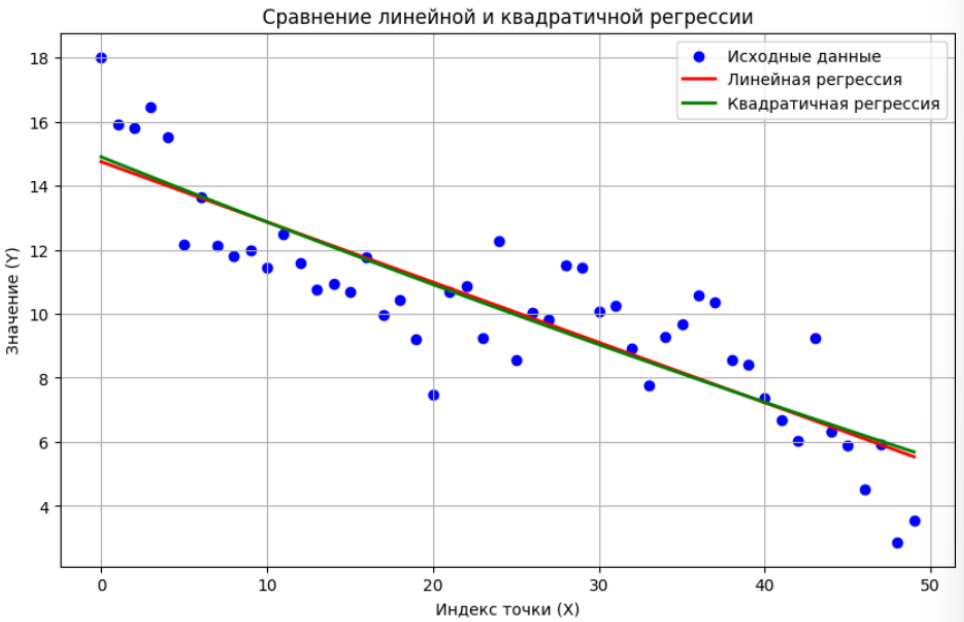
```

1. Нормальность остатков (Шапиро-Уилк):
   p-value = 0.329 Нормальны

2. Автокорреляция (Дарбин-Уотсон):
   DW = 0.993 Есть автокорреляция

3. Ранг матрицы X:
   Ранг = 2, Параметры = 2 Полный ранг

4. Нулевое среднее остатков:
   p-value = 1.000 E[ε]=0

5. Гомоскедастичность (Бреуш-Паган:
   p-value = 0.768 Дисперсия постоянна

Итог:
Теорема Гаусса-Маркова нарушена
{'normality': np.True_,
 'autocorrelation': np.False_,
 'full_rank': np.True_,
 'zero_mean': np.True_,
 'homoskedasticity': np.True_}
```

Квадратичная модель: $y = \beta_0 + \beta_1 x + \beta_2 x$

```python
from scipy.optimize import curve_fit

# Определяем квадратичную функцию
def quadratic_func(x, a, b, c):
    return a * x**2 + b * x + c

params, _ = curve_fit(quadratic_func, x, y)
a, b, c = params
y_pred_quad = quadratic_func(x, a, b, c)

plt.figure(figsize=(10, 6))
plt.scatter(x, y, color='blue', label='Исходные данные')
plt.plot(x, y_pred, color='red', linewidth=2, label='Линейная регрессия')
plt.plot(x, y_pred_quad, color='green', linewidth=2, label='Квадратичная регрессия')
plt.title('Сравнение линейной и квадратичной регрессии')
plt.xlabel('Индекс точки (X)')
plt.ylabel('Значение (Y)')
plt.grid(True)
plt.legend()
plt.show()
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.758
Model:                            OLS   Adj. R-squared:                  0.747
Method:                 Least Squares   F-statistic:                     73.44
Date:                Mon, 12 May 2025   Prob (F-statistic):           3.45e-15
Time:                        04:27:24   Log-Likelihood:                -92.391
No. Observations:                  50   AIC:                             190.8
Df Residuals:                      47   BIC:                             196.5
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         14.8978      0.646     23.064      0.000      13.598      16.197
x1            -0.2068      0.061     -3.393      0.001      -0.329      -0.084
x2             0.0004      0.001      0.319      0.751      -0.002       0.003
==============================================================================
Omnibus:                        1.390   Durbin-Watson:                   0.995
Prob(Omnibus):                  0.499   Jarque-Bera (JB):                1.098
Skew:                           0.115   Prob(JB):                        0.578
Kurtosis:                       2.312   Cond. No.                     3.16e+03
==============================================================================
```

$R^2 = 0.758$, не все коэффициенты значимы ($p_2 - value > 0.05$)



```
results_poly = gauss_markov(model_poly, X_poly, flag=True)
```

```
1. Нормальность остатков (Шапиро-Уилк):
   p-value = 0.355 Нормальны

2. Автокорреляция (Дарбин-Уотсон):
   DW = 0.995 Есть автокорреляция

3. Ранг матрицы X:
   Ранг = 3, Параметры = 3 Полный ранг

4. Нулевое среднее остатков:
   p-value = 1.000 E[ε]=0

5. Гомоскедастичность (Бреуш-Паган:
   p-value = 0.395 Дисперсия постоянна

Итог:
Теорема Гаусса-Маркова нарушена
```