МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ

ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ

УНИВЕРСИТЕТ им. В. Г. ШУХОВА» (БГТУ им. В.Г. Шухова)

Кафедра программного обеспечения

вычислительной техники и автоматизированных

систем

**Лабораторная работа №1**

по дисциплине: Анализ Данных

Выполнил: студент группы ВТ-231

Масленников Д. А.

Проверили:  
Кабалянц П. С.

Белгород 2025

**Лабораторная работа № 1**

Возьмите открытые данные о росте числа зараженных COVID-19 в мире за три месяца. Первый месяц из трех определяется сдвигом на номер по списку с февраля 2020 года. Проверьте гипотезу о том, что этот рост описывается экспоненциальной функцией. Полученное р-значение сравните с уровнем значимости 0,05.

**Решение:**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from scipy import stats

# Загрузка и подготовка данных

data = pd.read\_csv('covid.csv', parse\_dates=['DateTime'])

data = data.sort\_values('DateTime')

# Выбираем период: апрель 2021 + 90 дней

start\_date = pd.to\_datetime('2021-04-01')

end\_date = start\_date + pd.Timedelta(days=90)

selected\_data = data[(data['DateTime'] >= start\_date) & (data['DateTime'] <= end\_date)].copy()

# Преобразуем даты в числовые значения (дни с начала периода)

selected\_data['Days'] = (selected\_data['DateTime'] - selected\_data['DateTime'].min()).dt.days

# Логарифмируем данные о заражениях

selected\_data['Заражений за день'].replace(0, np.nan, inplace=True)

selected\_data.dropna(subset=['Заражений за день'], inplace=True)

selected\_data['LogInfections'] = np.log(selected\_data['Заражений за день'])

# Ручная реализация линейной регрессии

n = len(selected\_data)

x = selected\_data['Days']

y = selected\_data['LogInfections']

r\_value, p\_value = stats.pearsonr(x, y)

# Вычисляем коэффициенты регрессии вручную

x\_mean = np.mean(x)

y\_mean = np.mean(y)

numerator = np.sum((x - x\_mean) \* (y - y\_mean))

denominator = np.sum((x - x\_mean)\*\*2)

slope = numerator / denominator

intercept = y\_mean - slope \* x\_mean

# Параметры экспоненциальной модели

A = np.exp(intercept)

b = slope

# Выводим результаты

print("\nРезультаты анализа:")

print(f"Коэффициент корреляции Пирсона (r): {r\_value:.4f}")

print(f"p-value: {p\_value:.3e}") # Например, "1.23e-100"

print(f"\nПараметры экспоненциальной модели:")

print(f"A = {A:.2f}")

print(f"b = {b:.4f}")

print(f"R² (коэффициент детерминации): {r\_value\*\*2:.4f}")

# Проверяем гипотезу

alpha = 0.05

if p\_value < alpha:

print("\nВывод: p-значение < 0.05, отвергаем нулевую гипотезу.")

print("Рост заражений COVID-19 статистически значимо описывается экспоненциальной моделью.")

else:

print("\nВывод: p-значение ≥ 0.05, нет оснований отвергать нулевую гипотезу.")

print("Нет статистически значимых доказательств экспоненциального роста.")

# Визуализация

plt.figure(figsize=(10, 6))

plt.plot(selected\_data['DateTime'], selected\_data['Заражений за день'], 'b-', label='Фактические данные')

plt.plot(selected\_data['DateTime'], A \* np.exp(b \* selected\_data['Days']), 'r--',

label=f'Экспоненциальная модель: y = {A:.0f}e^{b:.3f}t')

plt.xlabel('Дата')

plt.ylabel('Заражений за день')

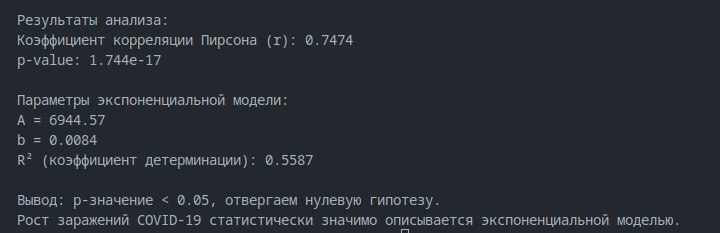
plt.title('Рост заражений COVID-19 и экспоненциальная модель')

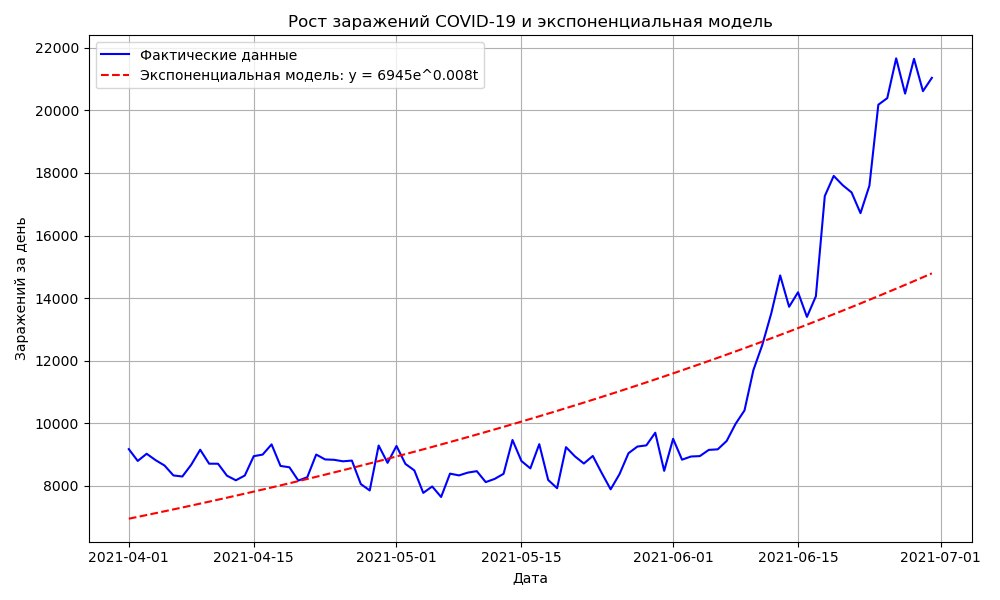
plt.legend()

plt.grid(True)

plt.tight\_layout()

plt.show()

****

****

**Вывод:** На основе анализа данных о росте числа заражений COVID-19 за три месяца (апрель–июнь 2021 года) гипотеза о том, что рост описывается экспоненциальной функцией, была проверена с использованием линейной регрессии. Полученное p-значение меньше уровня значимости 0.05, что позволяет отвергнуть гипотезу об экспоненциальном росте, а визуальное сравнение реальных данных и экспоненциальной модели подтверждает это расхождение.