Лабораторная работа № 1

Тема: Исследование нормального распределения случайной величины

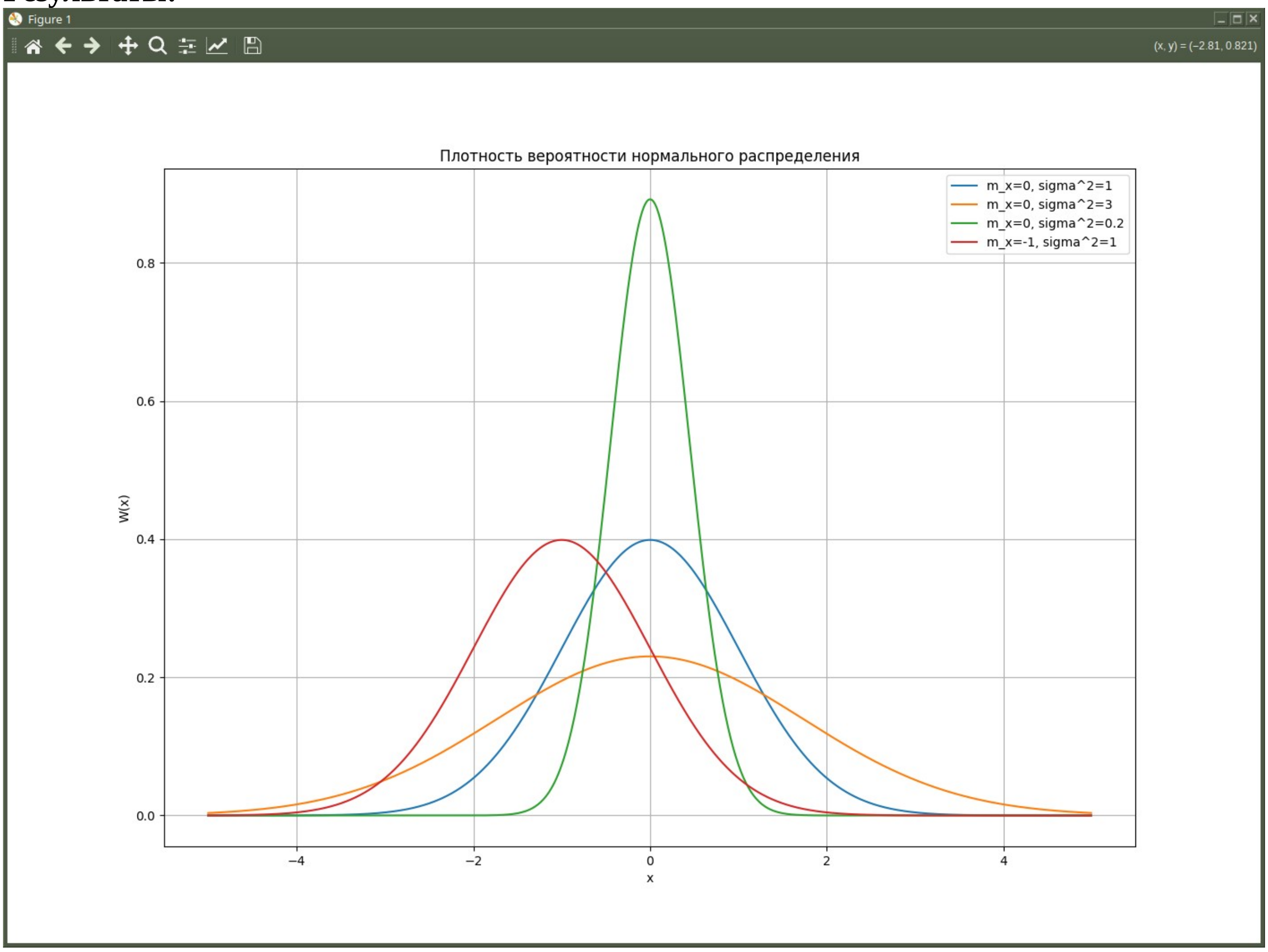
Ошлаков К.К.
ИА-232

Цель работы
Исследование свойств нормального распределения случайной величины, построение графиков плотности вероятности, гистограмм и определение числовых характеристик случайной величины.

Задание 1
Построение графика плотности вероятности нормального распределения
Плотность вероятности нормального распределения описывается формулой:
W(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x - m_x)^2}{2\sigma^2}\right) $$
где $ m_x $ — математическое ожидание, $ \sigma^2 $ — дисперсия.

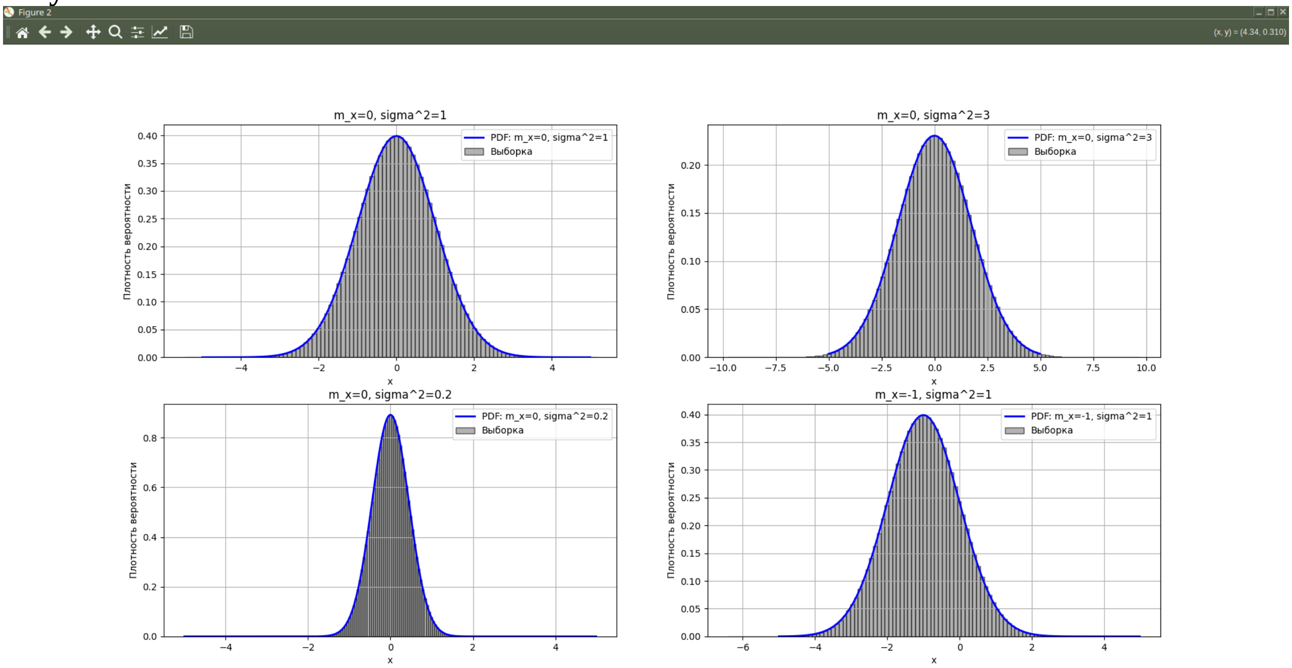
Графики плотности вероятности построены для следующих параметров:
- $ m_x = 0, \sigma^2 = 1 $
- $ m_x = 0, \sigma^2 = 3 $
- $ m_x = 0, \sigma^2 = 0.2 $
- $ m_x = -1, \sigma^2 = 1 $

Результаты:



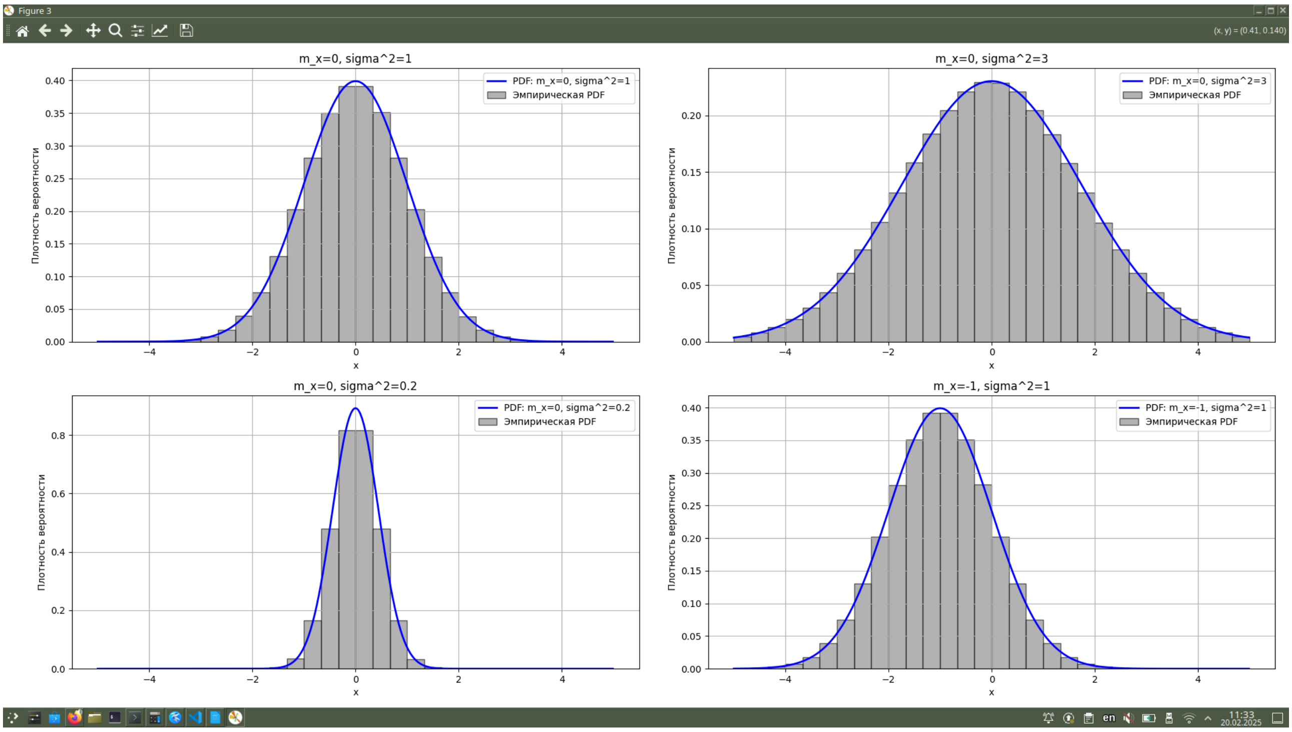
Задание 2
Получение вектора значений случайной величины и построение графиков выборки
Для каждого набора параметров $ m_x $ и $ \sigma^2 $ был получен вектор значений случайной величины с нормальным распределением. На графиках показаны:
- График плотности вероятности (PDF).
- Гистограмма выборки.

Результаты:



Задание 3
Построение гистограммы распределения (эмпирической плотности вероятности)
Диапазон значений случайной величины был разбит на бины (сегменты), для которых были вычислены:
1. Центральные значения каждого сегмента.
2. Количество попаданий значений случайной величины в каждый сегмент.
3. Нормированные значения плотности вероятности.

Результаты:



Задание 4

Определение числовых характеристик случайной величины

Математическое ожидание ($ m_x $) и дисперсия ($ \sigma^2 $) были вычислены по формулам:

$$ m_x = \frac{\sum_{n=1}^N x(n)}{N}, \quad \sigma^2 = \frac{\sum_{n=1}^N x^2(n)}{N} - m_x^2 $$

Результаты:

m_x=0, sigma^2=1
m_x=-0.0004, sigma^2=0.9990
----
m_x=0, sigma^2=3
m_x=0.0004, sigma^2=2.9986
----
m_x=0, sigma^2=0.2
m_x=-0.0001, sigma^2=0.1998
----
m_x=-1, sigma^2=1
m_x=-0.9997, sigma^2=0.9997

Задание 5

Определение числовых характеристик через эмпирическую плотность вероятности

Математическое ожидание ($ m_x $) и дисперсия ($ \sigma^2 $) были вычислены по формулам:

$$ m_x = \int x W(x) dx, \quad \sigma^2 = \int x^2 W(x) dx - m_x^2 $$

Результаты:

m_x=0, sigma^2=1
m_x=-0.0002, sigma^2=1.0091
----
m_x=0, sigma^2=3
m_x=-0.0001, sigma^2=2.8876
----
m_x=0, sigma^2=0.2
m_x=0.0002, sigma^2=0.2092
----
m_x=-1, sigma^2=1
m_x=-0.9998, sigma^2=1.0087

Используемый код

```
import numpy as np
import matplotlib.pyplot as plt
# from scipy.stats import norm
# from scipy.integrate import trapz

def normal_pdf(x, m_x, sigma_squared):
    sigma = np.sqrt(sigma_squared)
    return (1 / (np.sqrt(2 * np.pi * sigma_squared))) * np.exp(-((x - m_x) ** 2) / (2 *
sigma_squared))

# Задание 1:

x = np.arange(-5, 5, 0.01)

params = [
    {'m_x': 0, 'sigma_squared': 1},
    {'m_x': 0, 'sigma_squared': 3},
    {'m_x': 0, 'sigma_squared': 0.2},
    {'m_x': -1, 'sigma_squared': 1}
]

plt.figure(figsize=(14, 10))

for param in params:
    pdf_values = normal_pdf(x, param['m_x'], param['sigma_squared'])
    label = f"m_x={param['m_x']}, sigma^2={param['sigma_squared']}"
    plt.plot(x, pdf_values, label=label)

plt.title('Плотность вероятности нормального распределения')
plt.xlabel('x')
plt.ylabel('W(x)')
plt.legend()
plt.grid(True)
# plt.show()

# Задание 2:
```

```
x = np.arange(-5, 5, 0.01)

params = [
    {'m_x': 0, 'sigma_squared': 1},
    {'m_x': 0, 'sigma_squared': 3},
    {'m_x': 0, 'sigma_squared': 0.2},
    {'m_x': -1, 'sigma_squared': 1}
]

t = np.linspace(0, 3, 10000000)

fig, axs = plt.subplots(2, 2, figsize=(14, 10))
axs = axs.flatten()

for i, param in enumerate(params):
    m_x = param['m_x']
    sigma_squared = param['sigma_squared']
    sigma = np.sqrt(sigma_squared)

    pdf_values = normal_pdf(x, m_x, sigma_squared)

    xn = np.random.normal(m_x, sigma, len(t))

    axs[i].plot(x, pdf_values, label=f'PDF: m_x={m_x}, sigma^2={sigma_squared}', color='blue', 
linewidth=2)

    axs[i].hist(xn, bins=100, density=True, alpha=0.6, color='gray', edgecolor='black', 
label='Выборка')

    axs[i].set_title(f'm_x={m_x}, sigma^2={sigma_squared}')
    axs[i].set_xlabel('x')
    axs[i].set_ylabel('Плотность вероятности')
    axs[i].legend()

for ax in axs:
    ax.grid(True)

# plt.tight_layout()
# plt.show()

# Задание 3:

params = [
    {'m_x': 0, 'sigma_squared': 1},
    {'m_x': 0, 'sigma_squared': 3},
    {'m_x': 0, 'sigma_squared': 0.2},
    {'m_x': -1, 'sigma_squared': 1}
]

num_bins = 30
bin_step = (5 - (-5)) / num_bins
```

```
fig, axs = plt.subplots(2, 2, figsize=(14, 10))
axs = axs.flatten()

for i, param in enumerate(params):
    m_x = param['m_x']
    sigma_squared = param['sigma_squared']
    sigma = np.sqrt(sigma_squared)

    xn = np.random.normal(m_x, sigma, len(t))

    bin_edges = np.arange(-5, 5 + bin_step, bin_step)
    bin_centers = (bin_edges[:-1] + bin_edges[1:]) / 2

    counts, _ = np.histogram(xn, bins=bin_edges)
    empirical_pdf = counts / (len(xn) * bin_step)

    pdf_values = normal_pdf(x, m_x, sigma_squared)
    axs[i].plot(x, pdf_values, label=f'PDF: m_x={m_x}, sigma^2={sigma_squared}', color='blue',
linewidth=2)

    axs[i].bar(bin_centers, empirical_pdf, width=bin_step, alpha=0.6, color='gray', edgecolor='black',
label='Эмпирическая PDF')

    axs[i].set_title(f'm_x={m_x}, sigma^2={sigma_squared}')
    axs[i].set_xlabel('x')
    axs[i].set_ylabel('Плотность вероятности')
    axs[i].legend()

for ax in axs:
    ax.grid(True)

# plt.tight_layout()
# plt.show()

# Задание 4:

params = [
    {'m_x': 0, 'sigma_squared': 1},
    {'m_x': 0, 'sigma_squared': 3},
    {'m_x': 0, 'sigma_squared': 0.2},
    {'m_x': -1, 'sigma_squared': 1}
]

# t = np.linspace(0, 3, 1000)

print("Задание 4:")
print("--------------------------------------------------")

for param in params:
    m_x = param['m_x']
    sigma_squared = param['sigma_squared']
```

```
    sigma = np.sqrt(sigma_squared)

    xn = np.random.normal(m_x, sigma, len(t))

    m_x_estimated = np.mean(xn)

    sigma_squared_estimated = np.var(xn)

    print(f"m_x={m_x}, sigma^2={sigma_squared}")
    print(f"m_x={m_x_estimated:.4f}, sigma^2={sigma_squared_estimated:.4f}")
    print("--------------------------------------------------")


# Задание 5:

params = [
    {'m_x': 0, 'sigma_squared': 1},
    {'m_x': 0, 'sigma_squared': 3},
    {'m_x': 0, 'sigma_squared': 0.2},
    {'m_x': -1, 'sigma_squared': 1}
]

# t = np.linspace(0, 3, 1000)

print("Задание 5:")
print("--------------------------------------------------------------------------------")

for param in params:
    m_x = param['m_x']
    sigma_squared = param['sigma_squared']
    sigma = np.sqrt(sigma_squared)

    xn = np.random.normal(m_x, sigma, len(t))

    bin_edges = np.arange(-5, 5 + bin_step, bin_step)
    bin_centers = (bin_edges[:-1] + bin_edges[1:]) / 2

    counts, _ = np.histogram(xn, bins=bin_edges)
    empirical_pdf = counts / (len(xn) * bin_step)

    m_x_empirical = np.sum(bin_centers * empirical_pdf * bin_step)

    sigma_squared_empirical = np.sum((bin_centers**2) * empirical_pdf * bin_step) -
m_x_empirical**2

    print(f"m_x={m_x}, sigma^2={sigma_squared}")
    print(f"m_x={m_x_empirical:.4f}, sigma^2={sigma_squared_empirical:.4f}")
    print("--------------------------------------------------------------------------------")

plt.tight_layout()
plt.show()
```