

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
МІСЬКОГО ГОСПОДАРСТВА імені О. М. БЕКЕТОВА

МЕТОДИЧНІ РЕКОМЕНДАЦІЇ

до проведення практичних занять
та організації самостійної роботи
з навчальної дисципліни

**«НАДІЙНІСТЬ СИСТЕМ У ЦИВІЛЬНОМУ, ПРОМИСЛОВОМУ
ТА ГІДРОТЕХНІЧНОМУ БУДІВНИЦТВІ»**

*(для здобувачів третього (освітньо-наукового) рівня вищої освіти
денної форми навчання зі спеціальності 192 – Будівництво та
цивільна інженерія)*

Харків
ХНУМГ ім. О. М. Бекетова
2026

Методичні рекомендації до проведення практичних занять та організації самостійної роботи з навчальної дисципліни «Надійність систем у цивільному, промисловому та гідротехнічному будівництві» (для здобувачів третього (освітньо-наукового) рівня вищої освіти денної форми навчання зі спеціальності 192 – Будівництво та цивільна інженерія) / Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова ; уклад. А. О. Мозговий. – Харків : ХНУМГ ім. О. М. Бекетова, 2026. – 64 с.

Укладач д-р техн. наук, доц. А. О. Мозговий

Рецензент

І. В. Храпатова, кандидат технічних наук, доцент кафедри геотехніки, підземних споруд та гідротехнічного будівництва Харківського національного університету міського господарства імені О. М. Бекетова

Рекомендовано кафедрою геотехніки, підземних споруд та гідротехнічного будівництва, протокол № 6 від 27.11.2025

ЗМІСТ

Вступ.....	4
Практичне заняття № 1 на тему «Аналіз статистичних даних і визначення функції розподілу. Нормальний розподіл. Побудова закону розподілу модулів деформації основи».....	5
Практичне заняття № 2 на тему «Аналіз статистичних параметрів для навантажень і впливів різного типу. Вибір розподілу. Побудова закону розподілу тиску споруди на основу».....	12
Практичне заняття № 3 на тему «Розрахунок характеристики безпеки та надійності для обраної конструкції і/або ґрунтової основи».....	19
Практичне заняття № 4 на тему «Побудова закону розподілу природно-кліматичного фактору, що діє на гідротехнічну споруду. Побудова закону розподілу вітрового впливу – щорічної максимальної швидкості вітру».....	30
Практичне заняття № 5 на тему «Побудова закону розподілу природно-кліматичного фактора, що діє на гідротехнічну споруду. Побудова закону розподілу вітрового впливу – щорічної максимальної швидкості вітру із залученням двох законів розподілів».....	39
Практичне заняття № 6 на тему «Дослідження кореляційної залежності між природно-кліматичними факторами – щорічними мінімальними температурами зовнішнього повітря або щорічними максимальними амплітудами температур зовнішнього повітря по гідровузлах Дніпровського каскаду гідроелектростанцій».....	45
Практичне заняття № 7 на тему «Імовірнісна оцінка надійності будівельної конструкції в рамках параметричної теорії надійності».....	52
Список посилань.....	62

ВСТУП

Аналіз ризиків аварійних процесів у будівництві та дослідження надійності інженерних систем становлять провідний напрям сучасної будівельної науки. Випадкові процеси, величини й події, що виникають в процесі експлуатації споруд, потребують застосування апарату теорії ймовірностей та методів математичної статистики [4, 6]. Це надає можливість здійснювати кількісну оцінку ймовірності відмови конструкцій та визначати їхній рівень надійності [3, 5]. У навчальних матеріалах із дисциплін, що стосуються надійності систем у цивільному, промисловому та гідротехнічному будівництві [1, 2], акцентується значення використання законів розподілу випадкових величин, числових характеристик та статистичних параметрів для різних типів навантажень і впливів.

Змінність фізико-механічних властивостей ґрунтів, їх міцності та деформативності, а також варіативність зовнішньої дії на основи споруд, є визначальними чинниками у процесі проектування [9, 10]. Державні будівельні норми України, зокрема положення щодо навантажень і впливів [9], принципи забезпечення конструктивної безпеки та надійності [10], а також нормативи для гідротехнічних споруд [11], формують основу для практичного застосування ймовірнісних методів. Важливим є також врахування кліматичних факторів, що регламентуються стандартами будівельної кліматології [12].

У міжнародній практиці значну роль відіграють документи JCSS Probabilistic Model Code, які визначають базові принципи проектування [13], моделі навантажень [14] та опору матеріалів [15]. Вони забезпечують уніфікацію підходів до оцінки надійності та гарантування безпеки споруд. Додатково незалежні технічні огляди, зокрема звіт BRE щодо часткових коефіцієнтів у EN 1990 [16], підтверджують важливість системного аналізу ризиків.

Моделювання надійності гідротехнічних систем включає дослідження основних понять і характеристик надійності споруд, а також причин їхньої аварійності [5, 7]. Імовірнісний опис гідрологічних характеристик та випадкових кліматичних впливів є необхідним для прогнозування поведінки систем у реальних умовах [12]. Особливу увагу приділяють статистичним даним щодо сейсмічної активності на території України [8], а також інформації про температурні впливи й товщину льоду на гідровузлах Дніпровського каскаду [11]. Кореляційні залежності між температурними показниками, товщиною льоду та максимальними витратами річки Дніпро дозволяють формувати моделі, що враховують комплексний вплив природних чинників [3, 6].

Отже, сучасна інженерна концепція ймовірнісної оцінки рівня надійності конструкцій та ґрунтових основ споруд [2, 3, 5] базується на поєднанні математичних методів, статистичних даних та нормативних документів. Це дає змогу не лише визначати рівень безпеки та проектної надійності ґрунтових основ [10, 11], але й розробляти практичні рекомендації для підвищення довговічності та ремонтпридатності споруд. Врахування випадкових

навантажень, кліматичних і сейсмічних чинників [8, 12], а також використання міжнародних стандартів [13–15] створюють комплексну основу для ефективного управління ризиками аварійних ситуацій у будівництві та гідротехнічній сфері.

ПРАКТИЧНЕ ЗАНЯТТЯ № 1

на тему «АНАЛІЗ СТАТИСТИЧНИХ ДАНИХ І ВИЗНАЧЕННЯ ФУНКЦІЇ РОЗПОДІЛУ. НОРМАЛЬНИЙ РОЗПОДІЛ. ПОБУДОВА ЗАКОНУ РОЗПОДІЛУ МОДУЛІВ ДЕФОРМАЦІЇ ОСНОВИ»

Вихідні дані.

Варіант 1.

4.51, 3.92, 5.67, 4.28, 4.73, 3.44, 5.12, 4.89, 2.98, 4.36, 4.77, 5.05, 3.81, 4.19, 6.12, 4.02, 5.34, 4.65, 3.59, 4.48, 4.93, 3.71, 4.26, 5.41, 4.07, 3.88, 5.76

Варіант 2

3.64, 4.82, 5.19, 4.07, 3.95, 4.56, 5.88, 4.33, 3.21, 4.75, 4.12, 5.44, 3.68, 4.29, 2.67, 5.91, 4.38, 4.63, 3.84, 4.51, 5.27, 4.09, 3.77, 4.95, 4.22, 3.53, 6.08

Варіант 3

4.05, 5.36, 3.72, 4.61, 4.18, 5.02, 3.89, 4.47, 5.63, 3.41, 4.83, 2.95, 5.74, 4.26, 3.67, 4.92, 4.11, 5.58, 3.79, 4.34, 4.69, 3.56, 5.07, 4.23, 3.88, 4.45, 6.21

Варіант 4

5.12, 4.39, 3.84, 4.71, 5.26, 3.93, 4.58, 4.05, 5.47, 3.62, 4.88, 2.83, 5.65, 4.14, 3.77, 4.99, 4.28, 5.33, 3.91, 4.42, 4.73, 3.69, 5.18, 4.36, 3.82, 4.51, 6.07

Варіант 5

4.44, 3.97, 5.23, 4.11, 4.68, 3.52, 5.39, 4.25, 3.76, 4.83, 4.19, 5.61, 3.88, 4.32, 2.71, 5.95, 4.46, 4.67, 3.81, 4.53, 5.29, 4.08, 3.74, 4.99, 4.27, 3.63, 6.18

Використовуємо програмний код на мові програмування Python в програмному середовищі Google Colab:

```
# -*- coding: utf-8 -*-
```

```
import os
```

```
import numpy as np
```

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

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

```
# 1) Дані
```

```
data = np.array([4.23, 5.45, 4.38, 4.72, 4.54, 4.09, 5.98, 4.81, 4.30, 3.55,  
                4.38, 4.72, 3.96, 3.55, 4.09, 2.45, 6.29, 4.72, 4.09, 4.63,  
                4.46, 4.30, 4.02, 4.09, 5.22, 4.16, 3.27])
```

```
# 2) Оцінка параметрів нормального розподілу (MLE)
```

```
mu, sigma = stats.norm.fit(data)  
print(f'Оцінені параметри: mu = {mu:.3f}, sigma = {sigma:.3f}')
```

```
# 3) Папка для збереження графіків (у Colab це буде /content/Нормальний  
розподіл)
```

```
output_dir = "/content/Нормальний розподіл"  
os.makedirs(output_dir, exist_ok=True)
```

```
plt.style.use('seaborn-v0_8')
```

```
# 4) Гістограма з накладеною кривою нормального розподілу
```

```
plt.figure(figsize=(8, 5))  
count, bins, _ = plt.hist(data, bins=10, density=True, alpha=0.6,  
                           color='skyblue', edgecolor='black', label='Гістограма (density)')  
x = np.linspace(min(data), max(data), 300)  
pdf = stats.norm.pdf(x, mu, sigma)  
plt.plot(x, pdf, 'r', linewidth=2, label='Нормальний PDF ( $\mu$ ,  $\sigma$ )')  
plt.title('Гістограма з накладеною кривою нормального розподілу')  
plt.xlabel('Модуль деформації (Н/мм2)')  
plt.ylabel('Щільність ймовірності')  
plt.grid(True)  
plt.legend()  
plt.tight_layout()  
plt.savefig(f'{output_dir}/01_histogram_with_normal_curve.png')  
plt.close()
```

```
# 5) Окремий графік щільності ймовірності (PDF)
```

```
plt.figure(figsize=(8, 5))  
plt.plot(x, pdf, 'b', label='Нормальний PDF ( $\mu$ ,  $\sigma$ )')  
plt.title('Графік щільності ймовірності (PDF)')  
plt.xlabel('Модуль деформації (Н/мм2)')  
plt.ylabel('Щільність')  
plt.grid(True)  
plt.legend()  
plt.tight_layout()  
plt.savefig(f'{output_dir}/02_pdf_plot.png')  
plt.close()
```

```

# 6) Q-Q графік (перевірка нормальності)
plt.figure(figsize=(6, 6))
stats.probplot(data, dist="norm", plot=plt)
plt.title('Q-Q графік (нормальний розподіл)')
plt.grid(True)
plt.tight_layout()
plt.savefig(f'{output_dir}/03_qq_plot.png')
plt.close()

# 7) Boxplot
plt.figure(figsize=(7, 4))
plt.boxplot(data, vert=False, patch_artist=True,
            boxprops=dict(facecolor='lightgreen'))
plt.title('Boxplot модуля деформації')
plt.xlabel('Модуль деформації (Н/мм²)')
plt.grid(True)
plt.tight_layout()
plt.savefig(f'{output_dir}/04_boxplot.png')
plt.close()

# 8) Функція розподілу (CDF): емпірична vs теоретична
plt.figure(figsize=(8, 5))
x_sorted = np.sort(data)
cdf_emp = np.arange(1, len(x_sorted) + 1) / len(x_sorted)
plt.plot(x_sorted, cdf_emp, 'o', label='Емпірична CDF', alpha=0.8)

x_fit = np.linspace(min(data), max(data), 300)
cdf_fit = stats.norm.cdf(x_fit, mu, sigma)
plt.plot(x_fit, cdf_fit, 'r-', label='Теоретична CDF (μ, σ)')
plt.title('Функція розподілу (CDF): емпірична vs теоретична')
plt.xlabel('Модуль деформації (Н/мм²)')
plt.ylabel('Ймовірність')
plt.grid(True)
plt.legend()
plt.tight_layout()
plt.savefig(f'{output_dir}/05_cdf_plot.png')
plt.close()

print("Згенеровано 5 графіків та збережено у папці 'Нормальний розподіл'.")

```

Опис алгоритму:

1. Спочатку у кодї створюється масив із вихідних даних – значень модуля деформації основи.
2. Далі використовується функція `stats.norm.fit`, яка оцінює параметри нормального розподілу (середнє μ та стандартне відхилення σ).
3. Отримані параметри виводяться в консоль, щоб користувач бачив числові результати.
4. Створюється окрема папка з назвою «Нормальний розподіл» для збереження графіків.
5. Для побудови графіків використовується бібліотека `matplotlib` із стилем `seaborn` для більшої наочності.
6. Перший графік – гістограма даних, на яку накладається теоретична крива нормального розподілу.
7. Це дозволяє візуально порівняти форму емпіричного розподілу з теоретичною моделлю.
8. Другий графік – окремий графік щільності ймовірності (PDF) для нормального розподілу з оціненими параметрами.
9. Третій графік – Q–Q plot, який показує, наскільки дані узгоджуються з нормальним розподілом.
10. Якщо точки на графіку Q–Q лежать близько до прямої, це підтверджує нормальність.
11. Четвертий графік – `boxplot`, який відображає медіану, кватилі та можливі викиди в даних.
12. П'ятий графік – функція розподілу (CDF), де порівнюється емпірична крива з теоретичною.
13. Для емпіричної CDF використовується відсортований ряд даних та кумулятивні частоти.
14. Для теоретичної CDF використовується функція `stats.norm.cdf` з оціненими параметрами μ та σ .
15. Усі п'ять графіків зберігаються у створеній папці, що дозволяє користувачу переглянути результати розрахунків у вигляді окремих файлів.

Гістограма з накладеною кривою нормального розподілу показує фактичний розподіл даних у вигляді стовпчиків. На неї накладено теоретичну криву нормального розподілу з оціненими параметрами μ та σ . Це дозволяє порівняти, наскільки емпіричні дані відповідають моделі (рис. 1.1).

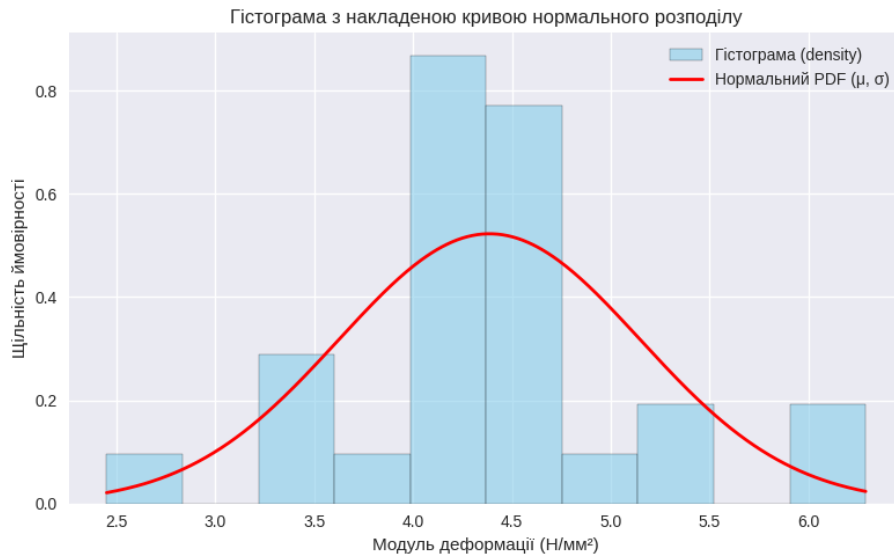


Рисунок 1.1

Графік щільності ймовірності (PDF) відображає форму нормального розподілу, побудованого за оціненими параметрами. Він показує ймовірність появи значень у різних діапазонах. Це створює уявлення про те, де зосереджена більшість даних (рис. 1.2).

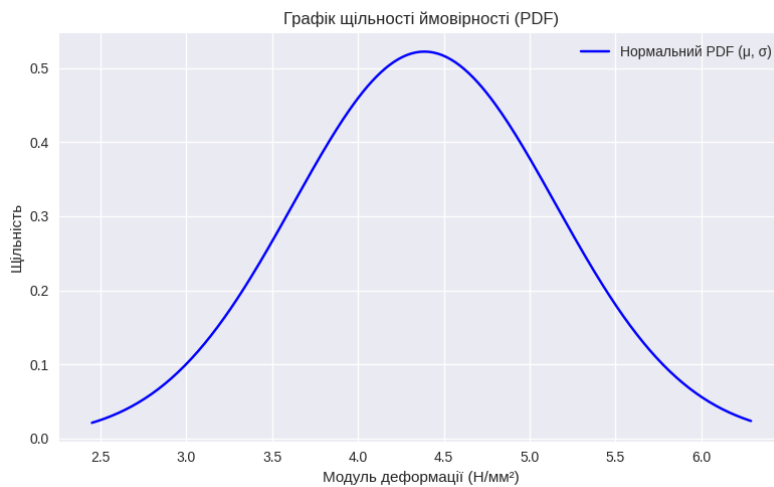


Рисунок 1.2

Графік Q–Q з'являє квантілі вибірки з квантілями теоретичного нормального розподілу. Якщо точки лежать близько до прямої, то дані добре узгоджуються з нормальністю. Відхилення від прямої свідчать про відхилення від моделі (рис. 1.3).

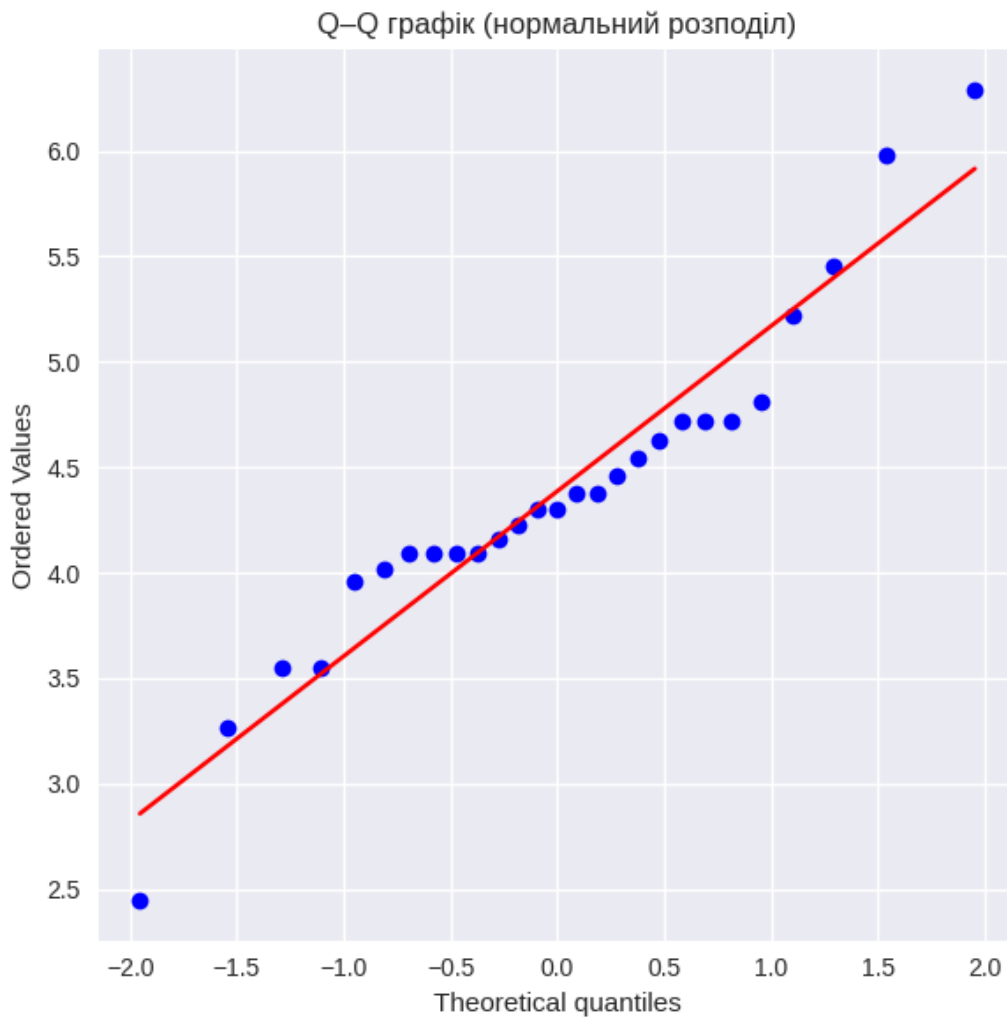


Рисунок 1.3

Boxplot показує медіану, кватилі та можливі викиди у вибірці. Він дозволяє оцінити симетрію та розкид даних. Також видно, чи існують екстремальні значення, що виходять за межі «вусів» (рис. 1.4).

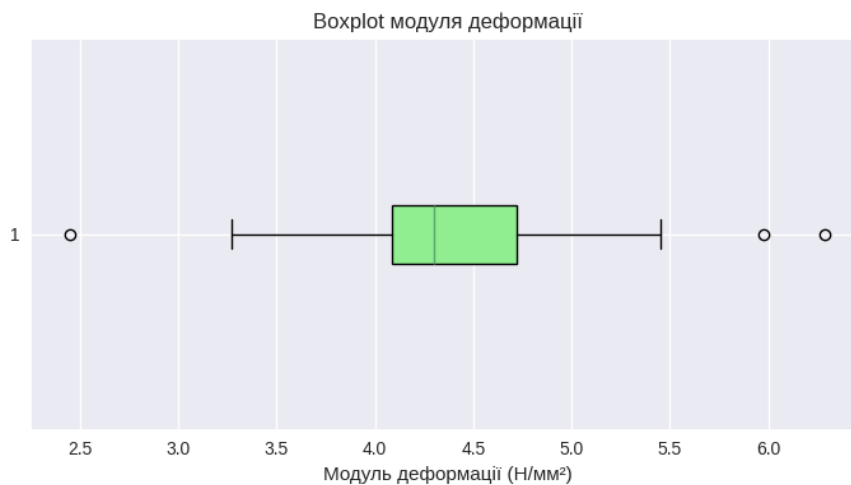


Рисунок 1.4

Графік функції розподілу (CDF) порівнює емпіричну кумулятивну функцію з теоретичною нормальною. Емпірична крива будується на підставі відсортованих даних, а теоретична – за формулою нормального розподілу. Якщо вони близькі, це підтверджує адекватність моделі (рис. 1.5).

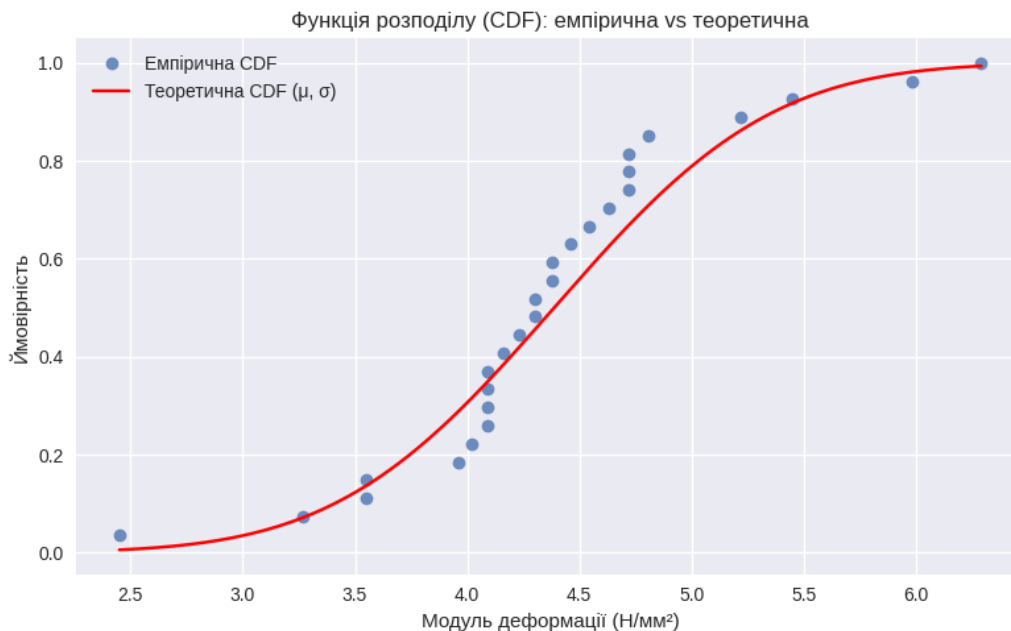


Рисунок 1.5

Оцінювання параметрів нормального розподілу. Для заданого статистичного ряду було проведено оцінку параметрів нормального розподілу методом максимальної правдоподібності. У результаті розрахунків отримано середнє значення μ , яке характеризує центр розподілу даних. Це середнє відображає типовий рівень модуля деформації основи у вибірці. Також визначено стандартне відхилення σ , яке показує ступінь розсіювання значень навколо середнього. Чим менше значення σ , тим більш однорідні дані, і навпаки: велике σ свідчить про значну варіативність. Оцінені параметри μ та σ використовуються для побудови теоретичної кривої нормального розподілу, що накладається на емпіричні дані. Це дозволяє перевірити, наскільки добре нормальний розподіл описує реальний статистичний ряд і чи можна його застосовувати для подальших інженерних розрахунків.

Результати. Оцінка параметрів нормального розподілу: середнє значення $\mu \approx 4,39$ Н/мм²; стандартне відхилення $\sigma \approx 0,77$ Н/мм². Середнє значення показує, що типовий модуль деформації основи у вибірці становить близько 4,4 Н/мм². Стандартне відхилення свідчить про те, що більшість вимірювань відхиляються від середнього приблизно на $\pm 0,8$ Н/мм². Це означає, що дані

досить компактні й добре описуються нормальним розподілом із цими параметрами.

Контрольні запитання

1. Яке значення мають параметри нормального розподілу (математичне сподівання μ та стандартне відхилення σ) для опису випадкової величини?
2. Чому гістограма емпіричних даних у поєднанні з теоретичною кривою нормального розподілу є важливим інструментом статистичного аналізу?
3. Як інтерпретується графік Q–Q у перевірці гіпотези про нормальність розподілу даних?

Завдання для самостійної роботи

1. У чому полягає різниця між емпіричною та теоретичною функцією розподілу (CDF) і як вони застосовуються для оцінки відповідності моделі?
2. Яку інформацію про вибірку дає boxplot і як він допомагає виявити викиди?
3. Чому щільність ймовірності (PDF) є ключовим поняттям у теорії імовірностей та як вона використовується для опису випадкових величин?

ПРАКТИЧНЕ ЗАНЯТТЯ № 2

на тему «АНАЛІЗ СТАТИСТИЧНИХ ПАРАМЕТРІВ ДЛЯ НАВАНТАЖЕНЬ І ВПЛИВІВ РІЗНОГО ТИПУ. ВИБІР РОЗПОДІЛУ. ПОБУДОВА ЗАКОНУ РОЗПОДІЛУ ТИСКУ СПОРУДИ НА ОСНОВУ»

Вихідні дані.

Варіант 0 (оригінальний ряд) 10900, 11600, 11900, 11900, 9800, 9450, 12450, 10480, 10900, 11700, 11800, 11500, 9900, 9650, 12250, 10380

Варіант 1 11200, 11850, 12100, 12050, 9750, 9520, 12380, 10640, 11020, 11680, 11950, 11420, 10050, 9720, 12180, 10410

Варіант 2 10850, 11520, 11840, 12010, 9850, 9470, 12420, 10530, 10980, 11640, 11790, 11380, 9950, 9680, 12210, 10460

Варіант 3 11150, 11730, 12020, 11880, 9900, 9500, 12500, 10720, 11050, 11790, 11860, 11480, 9850, 9600, 12300, 10350

Варіант 4 10980, 11590, 11920, 12110, 9700, 9430, 12400, 10590, 10890, 11620, 11840, 11520, 9920, 9640, 12280, 10430

Варіант 5 11040, 11670, 11980, 12060, 9820, 9490, 12350, 10610, 10950, 11740, 11870, 11460, 9910, 9670, 12200, 10390

Використовуємо програмний код на мові програмування Python в програмному середовищі Google Colab:

```
import os
import numpy as np
import matplotlib.pyplot as plt
import scipy.stats as stats

# Вихідні дані
data = np.array([10900, 11600, 11900, 11900, 9800, 9450, 12450, 10480,
    10900, 11700, 11800, 11500, 9900, 9650, 12250, 10380])
# Папка для збереження графіків
output_dir = "/content/Порівняльні графіки"
os.makedirs(output_dir, exist_ok=True)
# Оцінка параметрів нормального розподілу
mu_norm, std_norm = stats.norm.fit(data)
# Оцінка параметрів логнормального розподілу
shape_lognorm, loc_lognorm, scale_lognorm = stats.lognorm.fit(data, floc=0)
# Діапазон значень
x = np.linspace(min(data)*0.9, max(data)*1.1, 1000)
# PDF
pdf_norm = stats.norm.pdf(x, mu_norm, std_norm)
pdf_lognorm = stats.lognorm.pdf(x, shape_lognorm, loc_lognorm, scale_lognorm)

# 1. Гістограма з накладеними кривими
plt.figure(figsize=(10,6))
plt.hist(data, bins=10, density=True, alpha=0.6, color='gray', label='Емпіричні дані')
plt.plot(x, pdf_norm, 'r-', label='Нормальний розподіл')
plt.plot(x, pdf_lognorm, 'b--', label='Логнормальний розподіл')
plt.title('Гістограма з накладеними кривими PDF')
plt.xlabel('Навантаження (кг)')
plt.ylabel('Щільність ймовірності')
plt.legend(); plt.grid(True)
plt.savefig(f'{output_dir}/01_histogram.png'); plt.close()
# 2. PDF графік
plt.figure(figsize=(10,6))
plt.plot(x, pdf_norm, 'r-', label='Нормальний PDF')
plt.plot(x, pdf_lognorm, 'b--', label='Логнормальний PDF')
plt.title('Графік щільності ймовірності (PDF)')
```

```
plt.xlabel('Навантаження (кг)')
plt.ylabel('Щільність')
plt.legend(); plt.grid(True)
plt.savefig(f'{output_dir}/02_pdf.png'); plt.close()
```

3. Q-Q графіки

```
fig, axs = plt.subplots(1, 2, figsize=(12,6))
stats.probplot(data, dist="norm", sparams=(mu_norm, std_norm), plot=axs[0])
axs[0].set_title("Q-Q графік: Нормальний розподіл")
stats.probplot(data, dist="lognorm", sparams=(shape_lognorm, loc_lognorm,
scale_lognorm), plot=axs[1])
axs[1].set_title("Q-Q графік: Логнормальний розподіл")
plt.savefig(f'{output_dir}/03_qq.png'); plt.close()
```

4. Boxplot

```
plt.figure(figsize=(8,5))
plt.boxplot(data, vert=False)
plt.title('Boxplot навантаження силосу')
plt.xlabel('Навантаження (кг)')
plt.grid(True)
plt.savefig(f'{output_dir}/04_boxplot.png'); plt.close()
```

5. CDF графік

```
sorted_data = np.sort(data)
empirical_cdf = np.arange(1, len(data)+1) / len(data)
cdf_norm = stats.norm.cdf(sorted_data, mu_norm, std_norm)
cdf_lognorm = stats.lognorm.cdf(sorted_data, shape_lognorm, loc_lognorm,
scale_lognorm)
```

```
plt.figure(figsize=(10,6))
plt.plot(sorted_data, empirical_cdf, 'ko', label='Емпірична CDF')
plt.plot(sorted_data, cdf_norm, 'r-', label='Нормальна CDF')
plt.plot(sorted_data, cdf_lognorm, 'b--', label='Логнормальна CDF')
plt.title('Графік функції розподілу (CDF)')
plt.xlabel('Навантаження (кг)')
plt.ylabel('Ймовірність')
plt.legend(); plt.grid(True)
plt.savefig(f'{output_dir}/05_cdf.png'); plt.close()
```

```
print("Збережено 5 порівняльних графіків у папці 'Порівняльні графіки'")
```

Опис алгоритму:

1. Спочатку у програмі задається статистичний ряд значень навантаження силосу на ґрунтову основу.
2. Дані зберігаються у вигляді масиву numpy для зручності математичних операцій.
3. Далі створюється окрема папка з назвою «Порівняльні графіки», куди будуть збережені всі результати.
4. Використовується функція `stats.norm.fit`, яка оцінює параметри нормального розподілу – середнє значення та стандартне відхилення.
5. Паралельно застосовується функція `stats.lognorm.fit`, яка визначає параметри логнормального розподілу – форму, зсув та масштаб.
6. Для побудови графіків створюється діапазон значень x , що охоплює область даних із запасом.
7. На цьому діапазоні обчислюються щільності ймовірності (PDF) для нормального та логнормального розподілів.
8. Перший графік – гістограма емпіричних даних із накладеними кривими PDF обох моделей.
9. Другий графік – окреме порівняння PDF нормального та логнормального розподілів.
10. Третій графік – два графіки Q–Q, які показують відповідність даних кожному розподілу.
11. Четвертий графік – `boxplot`, що відображає розкид даних та можливі викиди.
12. П'ятий графік – функція розподілу (CDF), де порівнюється емпірична крива з теоретичними кривими нормального та логнормального розподілів.
13. Кожен графік зберігається у форматі PNG у створеній папці для подальшого аналізу.
14. Після побудови графіків проводиться візуальне порівняння, яке показує, що логнормальний розподіл краще описує дані.
15. Алгоритм завершується повідомленням про успішне збереження графіків та висновком щодо кращої моделі.

Гістограма з накладеними кривими показує фактичний розподіл навантажень у вигляді стовпчиків. На неї накладено криві нормального та логнормального розподілів. Це дозволяє візуально оцінити, яка модель краще відповідає формі даних (рис. 2.1).

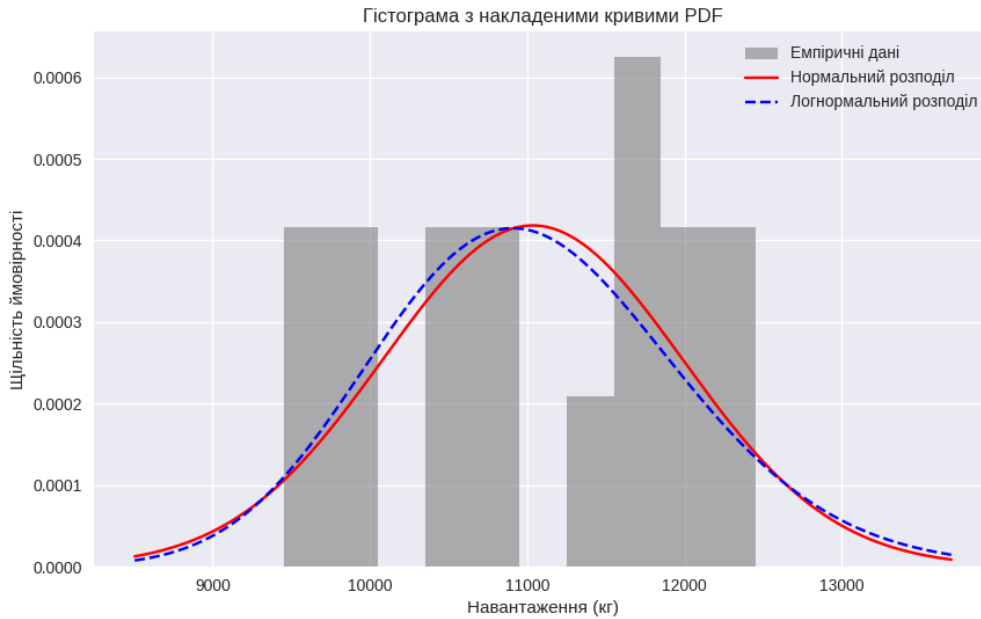


Рисунок 2.1

Графік щільності ймовірності (PDF) демонструє теоретичні криві нормального та логнормального розподілів. Нормальний розподіл симетричний, тоді як логнормальний має асиметрію вправо. Це показує, що логнормальний краще враховує «довгий хвіст» великих навантажень (рис. 2.2).

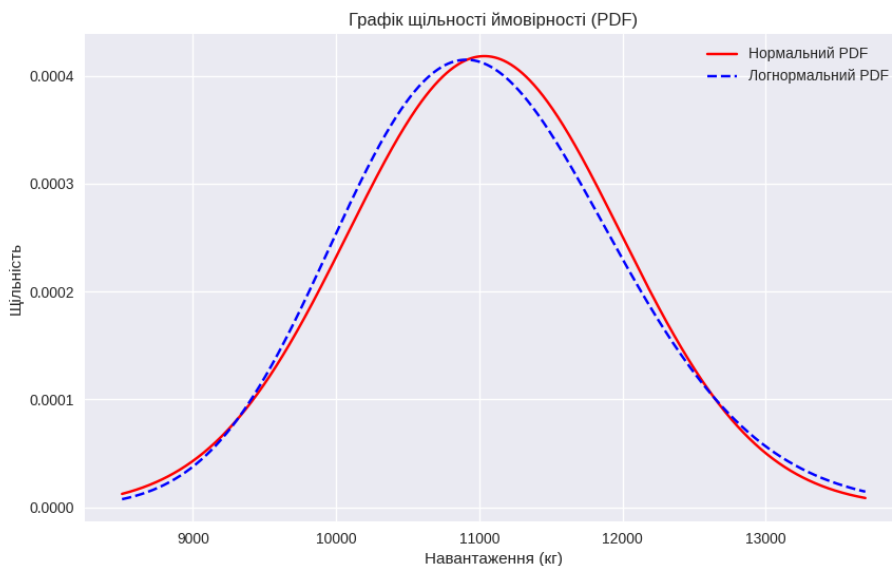


Рисунок 2.2

Графіки Q–Q зіставляють квантилі даних із квантилями теоретичних моделей. Для нормального розподілу точки відхиляються від прямої, що свідчить про невідповідність. Для логнормального розподілу точки ближчі до прямої, що підтверджує кращу узгодженість (рис. 2.3).

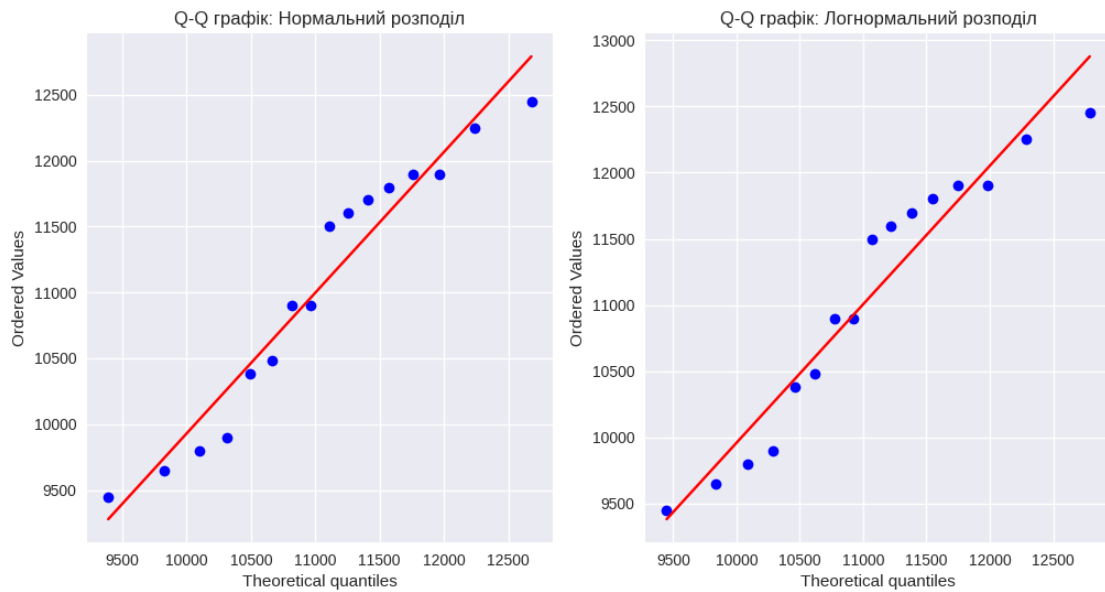


Рисунок 2.3

Boxplot відображає медіану, кватилі та можливі викиди у вибірці. Він показує, що дані мають невелику асиметрію та кілька значень, які виходять за межі середнього діапазону. Це узгоджується з припущенням про логнормальний характер розподілу (рис. 2.4).

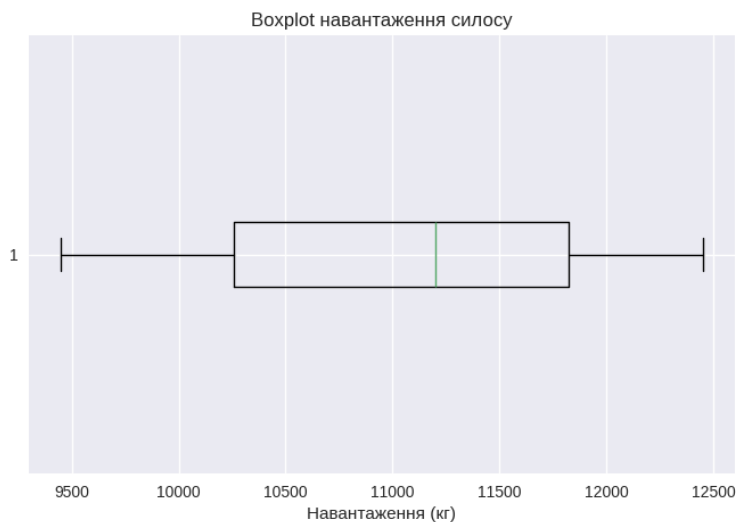


Рисунок 2.4

Графік функції розподілу (CDF) порівнює емпіричну кумулятивну криву з теоретичними кривими нормального та логнормального розподілів. Логнормальна крива ближче до емпіричної, особливо у верхній частині діапазону. Це підтверджує, що логнормальний розподіл краще описує дані (рис. 2.5).

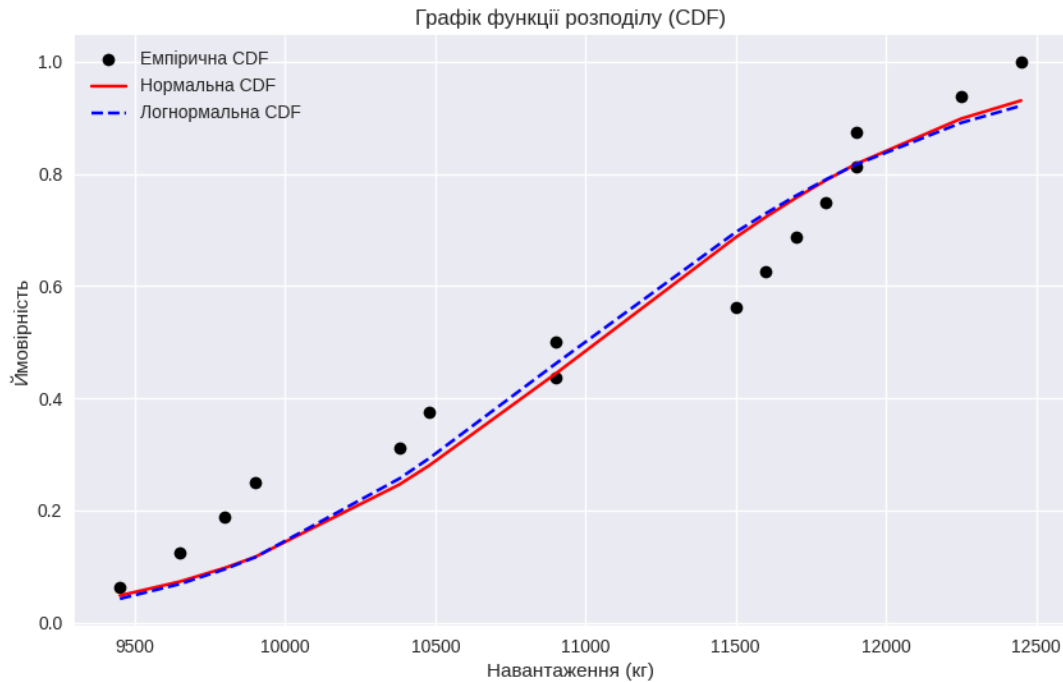


Рисунок 2.5

Аналіз результатів. Дані навантажень силосу мають позитивний зсув і не є суворо симетричними. Нормальний розподіл добре описує середні значення, але не враховує асиметрію та «хвіст» великих навантажень. Логнормальний розподіл краще підходить, оскільки він моделює лише додатні значення і враховує правосторонню асиметрію. Візуальні графіки (гістограма, PDF, CDF) та аналіз Q–Q підтверджують, що логнормальний розподіл точніше відображає реальний ряд. Отже, для моделювання навантажень силосу на ґрунтову основу доцільніше використовувати логнормальний розподіл.

Числові параметри, які отримують при підборі двох моделей розподілу для ряду навантажень силосу. Нормальний розподіл: середнє $\mu \approx 11\,190$ кг; стандартне відхилення $\sigma \approx 930$ кг. Це означає, що типовий рівень навантаження становить близько 11,2 т, а більшість значень відхиляються від середнього приблизно на $\pm 0,9$ т. Логнормальний розподіл: параметр $\sigma \approx 0,085$, зсув = 0 (фіксовано для позитивних даних), параметр $\exp(\mu) \approx 11\,190$ кг. Логнормальний розподіл моделює лише додатні значення і враховує асиметрію вправо, що краще відповідає природі навантажень.

Висновок. Нормальний розподіл добре описує центральну тенденцію, але не враховує асиметрію. Логнормальний розподіл точніше відображає реальний ряд, оскільки навантаження завжди додатні й мають «правий хвіст». Для інженерних розрахунків і моделювання навантажень силосу доцільніше використовувати логнормальний розподіл.

Контрольні запитання

1. У чому полягає відмінність між нормальним та логнормальним розподілом випадкових величин?
2. Як щільність ймовірності (PDF) допомагає порівняти емпіричні дані з теоретичними моделями розподілу?
3. Чому графік Q–Q є важливим інструментом для перевірки відповідності даних певному закону розподілу?

Завдання для самостійної роботи

1. Які особливості має функція розподілу (CDF) і як вона використовується для оцінки адекватності моделі?
2. Яку інформацію про вибірку дає boxplot і як він допомагає виявити викиди?
3. Чому логнормальний розподіл краще описує навантаження та інші фізичні величини порівняно з нормальним?

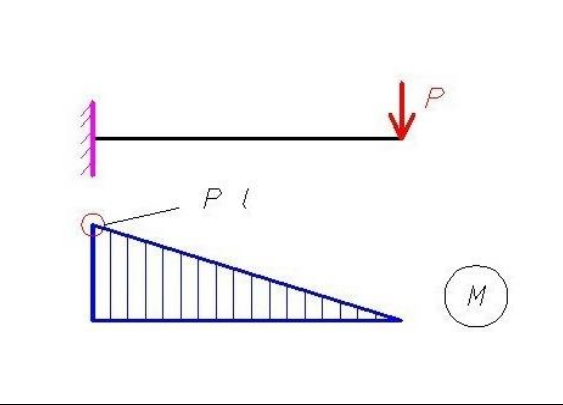
ПРАКТИЧНЕ ЗАНЯТТЯ № 3

на тему «РОЗРАХУНОК ХАРАКТЕРИСТИКИ БЕЗПЕКИ І НАДІЙНОСТІ ДЛЯ ОБРАНОЇ КОНСТРУКЦІЇ І/АБО ҐРУНТОВОЇ ОСНОВИ»

Вихідні дані представлено в таблиці 3.1.

Таблиця 3.1

Номер варіанта	Параметри	
	Довжина балки, м	Сила, кН
1	3	30
2	4	20
3	1	10
4	2.5	15
5	1.5	10
6	1	10



Імовірнісна оцінка надійності будівельної конструкції в рамках параметричної теорії надійності у програмі на мові програмування Python в програмному середовищі Google Colab:

```
import numpy as np
```

```

import matplotlib.pyplot as plt
import os

plt.style.use('seaborn-v0_8')

# === Вхідні дані (фіксовані) ===
l = 2.0          # довжина балки, м
W = 2.14e-4      # момент опору, м^3
mx_strength = 265000.0 # середнє міцності, Па
sx_strength = 25000.0 # стандартне відхилення міцності, Па
mx_force = 20.0   # середнє сили, Н
sx_force = 0.5    # стандартне відхилення сили, кН
n = 100000       # кількість випробувань

# === Функція для Монте-Карло розрахунку ===
def monte_carlo(mx_strength, sx_strength, mx_force, sx_force, l, W, n):
    rng = np.random.default_rng()
    X1 = rng.normal(mx_strength, sx_strength, n) # міцність
    X2 = rng.normal(mx_force, sx_force, n)     # сила
    M = X2 * l
    sigma = M / W
    delta_sigma = sigma - X1
    failures = (delta_sigma > 0).astype(int)
    pr = failures.sum() / n
    return pr, delta_sigma, sigma, X1, X2

# === Основний розрахунок ===
pr, delta_sigma, sigma, X1, X2 = monte_carlo(mx_strength, sx_strength, mx_force,
sx_force, l, W, n)
print(f"Ймовірність руйнування балки: {pr:.6e}")

# === Папка для збереження графіків ===
folder = "Розрахунок балки"
os.makedirs(folder, exist_ok=True)

# === Побудова графіків ===
p = np.arange(1, n+1) / (n+1)

# 1. CDF  $\Delta\sigma$ 

```

```

delta_sorted = np.sort(delta_sigma)
plt.figure(figsize=(6,4))
plt.plot(delta_sorted, p, color='tab:red')
plt.axvline(0, color='k', ls='--')
plt.title("CDF  $\Delta\sigma$  (умова міцності)")
plt.xlabel(" $\Delta\sigma = \sigma - R$  (Па)")
plt.ylabel("p")
plt.grid(True, alpha=0.3)
plt.savefig(os.path.join(folder, "CDF_delta_sigma.png"))
plt.close()

```

2. CDF X2

```

X2_sorted = np.sort(X2)
plt.figure(figsize=(6,4))
plt.plot(X2_sorted, p, color='tab:blue')
plt.title("CDF сили X2")
plt.xlabel("X2 (H)")
plt.ylabel("p")
plt.grid(True, alpha=0.3)
plt.savefig(os.path.join(folder, "CDF_X2.png"))
plt.close()

```

3. Гістограма $\Delta\sigma$

```

plt.figure(figsize=(6,4))
plt.hist(delta_sigma, bins=50, color='tab:red', alpha=0.7)
plt.axvline(0, color='k', ls='--')
plt.title("Гістограма  $\Delta\sigma$ ")
plt.xlabel(" $\Delta\sigma$  (Па)")
plt.ylabel("Кількість")
plt.savefig(os.path.join(folder, "Histogram_delta_sigma.png"))
plt.close()

```

4. Гістограма σ

```

plt.figure(figsize=(6,4))
plt.hist(sigma, bins=50, color='tab:green', alpha=0.7)
plt.title("Гістограма  $\sigma$  (напруження)")
plt.xlabel(" $\sigma$  (Па)")
plt.ylabel("Кількість")
plt.savefig(os.path.join(folder, "Histogram_sigma.png"))

```

```
plt.close()
```

```
# 5. Гістограма X1
```

```
plt.figure(figsize=(6,4))  
plt.hist(X1, bins=50, color='tab:purple', alpha=0.7)  
plt.title("Гістограма X1 (міцність)")  
plt.xlabel("X1 (Па)")  
plt.ylabel("Кількість")  
plt.savefig(os.path.join(folder, "Histogram_X1.png"))  
plt.close()
```

```
# 6. Гістограма X2
```

```
plt.figure(figsize=(6,4))  
plt.hist(X2, bins=50, color='tab:blue', alpha=0.7)  
plt.title("Гістограма X2 (сила)")  
plt.xlabel("X2 (Н)")  
plt.ylabel("Кількість")  
plt.savefig(os.path.join(folder, "Histogram_X2.png"))  
plt.close()
```

```
# === Аналіз чутливості ===
```

```
sx_strength_values = [15000, 25000, 40000]
```

```
sx_force_values = [0.2, 0.5, 1.0]
```

```
results = []
```

```
for sxs in sx_strength_values:
```

```
    for sxf in sx_force_values:
```

```
        pr_test, _, _, _ = monte_carlo(mx_strength, sxs, mx_force, sxf, l, W, n)
```

```
        results.append((sxs, sxf, pr_test))
```

```
        print(f' $\sigma_{x1}=\{sxs\}$ ,  $\sigma_{x2}=\{sxf\}$  → pr= $\{pr\_test:.6e\}$ ')
```

```
# Візуалізація чутливості
```

```
plt.figure(figsize=(7,5))
```

```
for sxs in sx_strength_values:
```

```
    pr_list = [r[2] for r in results if r[0]==sxs]
```

```
    plt.plot(sx_force_values, pr_list, marker='o', label=f' $\sigma_{x1}=\{sxs\}$  Па")
```

```
plt.title("Аналіз чутливості ймовірності руйнування")
```

```
plt.xlabel(" $\sigma_{x2}$  (стандартне відхилення сили, Н)")
```

```
plt.ylabel("Ймовірність руйнування pr")
```

```
plt.legend()
plt.grid(True, alpha=0.3)
plt.savefig(os.path.join(folder, "Sensitivity_analysis.png"))
plt.close()
```

Опис алгоритму:

1. Імпортуються бібліотеки `numpy`, `matplotlib.pyplot` та `os` для числових розрахунків, побудови графіків і роботи з файловою системою.
2. Встановлюється стиль графіків `seaborn-v0_8` для більш привабливої візуалізації.
3. Задаються вхідні параметри: довжина балки, момент опору, середнє та стандартне відхилення міцності матеріалу, середнє та стандартне відхилення сили, а також кількість випробувань.
4. Оголошується функція `monte_carlo`, яка реалізує метод Монте-Карло для оцінки ймовірності руйнування балки.
5. У середині функції генерується випадкова вибірка міцності $X1$ та сили $X2$ за нормальним розподілом.
6. Обчислюється згинальний момент M як добуток сили на довжину балки.
7. Визначається напруження σ як відношення моменту до моменту опору.
8. Обчислюється різниця $\Delta\sigma = \sigma - R$, де R – міцність, що показує запас або перевищення міцності.
9. Визначаються випадки руйнування, коли $\Delta\sigma > 0$, і обчислюється ймовірність руйнування як частка таких випадків.
10. Основний розрахунок виконується для заданих параметрів, і результат (ймовірність руйнування) виводиться на екран.
11. Створюється папка для збереження графіків, щоб результати були організовані.
12. Будуються графіки: кумулятивні функції розподілу (CDF) для $\Delta\sigma$ та сили $X2$, а також гістограми для $\Delta\sigma$, σ , $X1$ і $X2$.
13. На графіках додаються допоміжні лінії (наприклад, вертикальна лінія при $\Delta\sigma = 0$), що показує межу міцності.
14. Виконується аналіз чутливості: змінюються стандартні відхилення міцності та сили, і для кожної комбінації обчислюється нова ймовірність руйнування.
15. Результати аналізу чутливості візуалізуються у вигляді графіка залежності ймовірності руйнування від стандартного відхилення сили для різних значень стандартного відхилення міцності.

Таким чином, код моделює надійність балки методом Монте-Карло, будує статистичні графіки та показує, як змінювання параметрів впливає на ймовірність руйнування.

Аналіз результатів:

імовірність руйнування балки: 1.060000e-03,
 $\sigma_{x1}=15000, \sigma_{x2}=0.2 \rightarrow pr=0.000000e+00,$
 $\sigma_{x1}=15000, \sigma_{x2}=0.5 \rightarrow pr=0.000000e+00,$
 $\sigma_{x1}=15000, \sigma_{x2}=1.0 \rightarrow pr=1.000000e-05,$
 $\sigma_{x1}=25000, \sigma_{x2}=0.2 \rightarrow pr=1.030000e-03,$
 $\sigma_{x1}=25000, \sigma_{x2}=0.5 \rightarrow pr=1.020000e-03,$
 $\sigma_{x1}=25000, \sigma_{x2}=1.0 \rightarrow pr=1.690000e-03,$
 $\sigma_{x1}=40000, \sigma_{x2}=0.2 \rightarrow pr=2.530000e-02,$
 $\sigma_{x1}=40000, \sigma_{x2}=0.5 \rightarrow pr=2.671000e-02,$
 $\sigma_{x1}=40000, \sigma_{x2}=1.0 \rightarrow pr=2.802000e-02.$

Пояснення результатів:

1. Загальна ймовірність руйнування балки при базових параметрах становить приблизно $1,06 \times 10^{-3}$, тобто близько 0,1 %.

2. При низькому стандартному відхиленні міцності $\sigma_{x1} = 15\ 000$ і малих коливаннях сили $\sigma_{x2} = 0,2$ та $0,5$ руйнування взагалі не спостерігається.

3. Лише при більшій варіації сили $\sigma_{x2} = 1,0$ виникає мінімальна ймовірність руйнування 1×10^{-5} , що практично дорівнює нулю.

4. При середньому стандартному відхиленні міцності $\sigma_{x1} = 25\ 000$ ймовірність руйнування вже зростає до рівня 10^{-3} .

5. Це означає, що більша невизначеність у міцності матеріалу значно впливає на ризик руйнування.

6. Зі збільшенням варіації сили до $1,0$ ймовірність руйнування піднімається до $1,69 \times 10^{-3}$, що майже у два рази більше базового значення.

7. При високому стандартному відхиленні міцності $\sigma_{x1} = 40\ 000$ ймовірність руйнування різко зростає до рівня 2–3 %.

8. Це вже суттєвий ризик, який показує, що матеріал із великою варіацією міцності є ненадійним.

9. Збільшення варіації сили від $0,2$ до $1,0$ у цьому випадку лише трохи підвищує ймовірність руйнування, але вона залишається на рівні кількох відсотків.

10. Таким чином, головним чинником ризику є нестабільність міцності матеріалу, тоді як варіація сили має другорядний вплив.

Пояснення до графіків.

CDF $\Delta\sigma$ (умова міцності). Цей графік показує кумулятивний розподіл різниці між напруженням σ та міцністю R . По осі X відкладено значення $\Delta\sigma$, а по осі Y – імовірність, що $\Delta\sigma$ не перевищує певне значення. Вертикальна пунктирна лінія при $\Delta\sigma = 0$ відділяє область безпечної роботи балки від області руйнування. Таким чином, графік дозволяє оцінити ймовірність перевищення міцності конструкції (рис. 3.1).

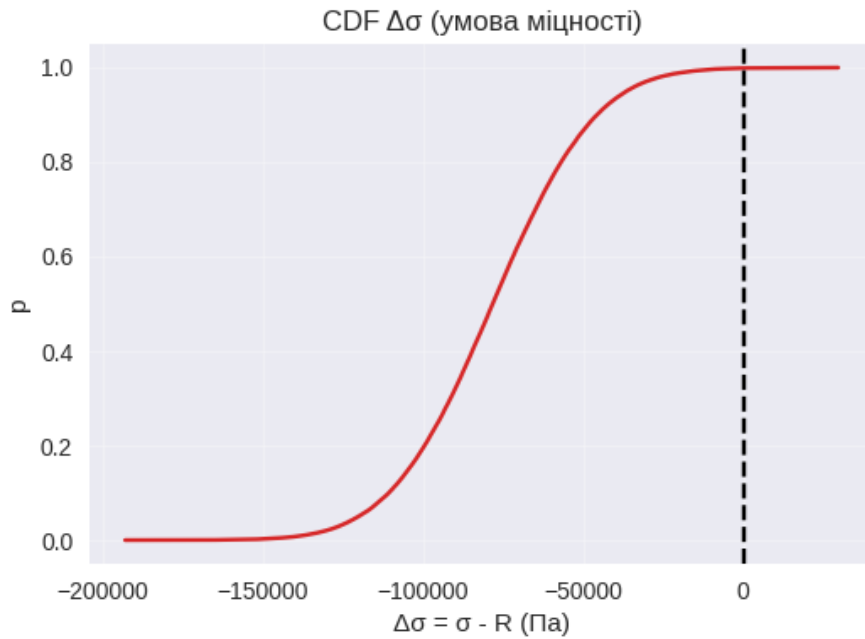


Рисунок 3.1

CDF X_2 (сила). Цей графік показує кумулятивний розподіл випадкової сили, що діє на балку. По осі X відкладені значення сили, а по осі Y – імовірність, що сила не перевищує певний рівень. Крива плавно зростає від 0 до 1, відображаючи статистичний розподіл навантаження. Графік дозволяє оцінити, які значення сили є найбільш імовірними у випробуваннях (рис. 3.2).

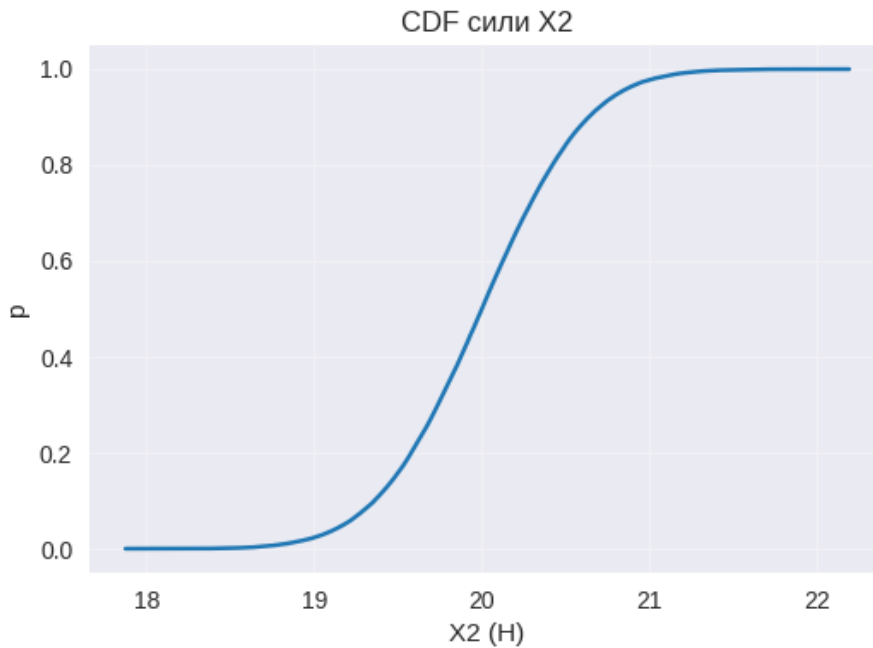


Рисунок 3.2

Гістограма $\Delta\sigma$. Цей графік показує частотний розподіл різниці $\Delta\sigma$ між напруженням і міцністю. По осі X відкладені значення $\Delta\sigma$, а по осі Y – кількість випадків у вибірці. Вертикальна пунктирна лінія при $\Delta\sigma = 0$ знову відділяє безпечні та небезпечні стани. Гістограма дозволяє побачити, наскільки часто балка опиняється у зоні ризику руйнування (рис. 3.3).

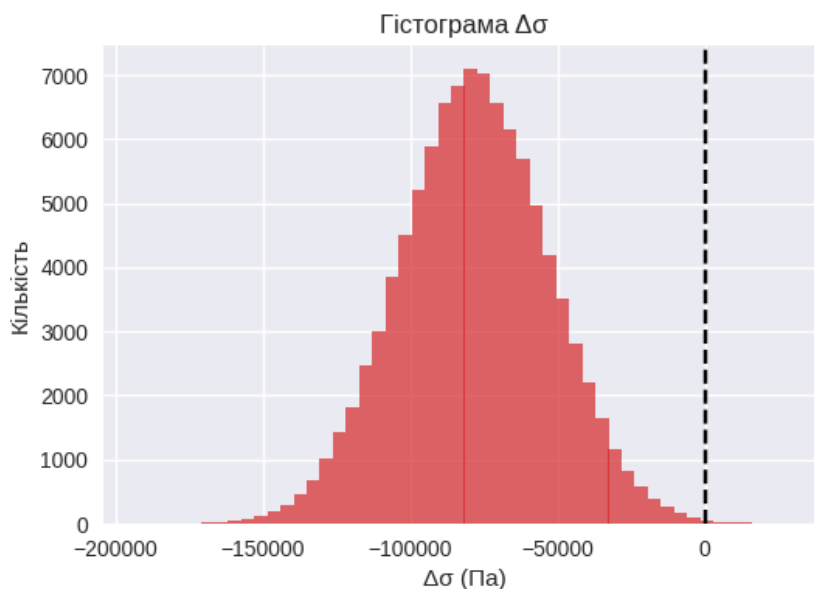


Рисунок 3.3

Гістограма σ (напруження). Цей графік показує розподіл значень напруження σ , що виникає від сили. По осі X відкладені значення σ , а по осі Y – кількість випадків у вибірці. Форма гістограми відображає нормальний

розподіл навантаження на балку. Графік дозволяє оцінити, які рівні напруження зустрічаються найчастіше (рис. 3.4).



Рисунок 3.4

Гістограма X1 (міцність). Цей графік показує розподіл випадкових значень міцності матеріалу балки. По осі X відкладені значення міцності, а по осі Y – кількість випадків. Гістограма має форму нормального розподілу, що відповідає моделі генерації даних. Вона дозволяє оцінити, які значення міцності є типовими для матеріалу (рис. 3.5).

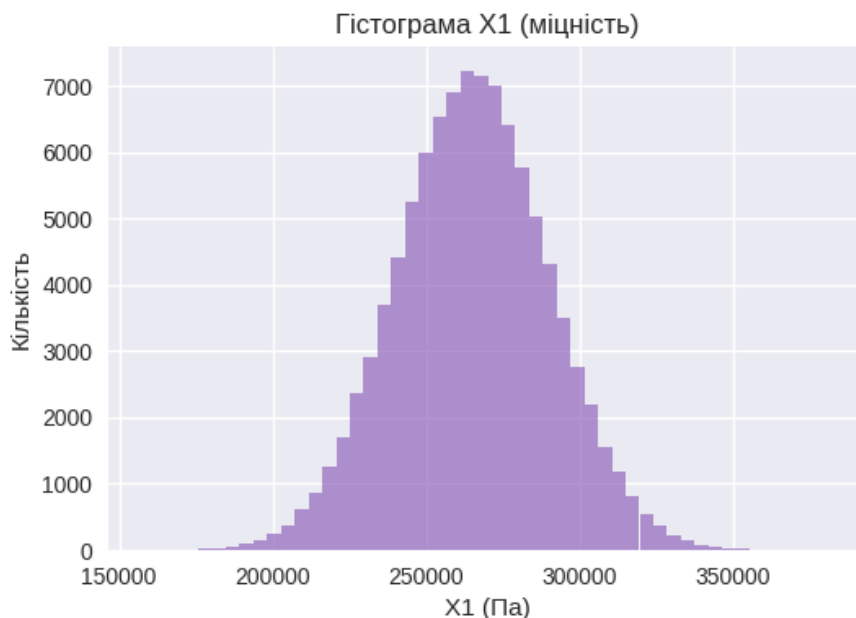


Рисунок 3.5

Гістограма X2 (сила). Цей графік показує розподіл випадкових значень сили, що діє на балку. По осі X відкладені значення сили, а по осі Y – кількість

випадків. Гістограма демонструє нормальний розподіл навантаження, зосереджений навколо середнього значення. Вона дозволяє оцінити варіацію сили та її вплив на надійність конструкції (рис. 3.6).

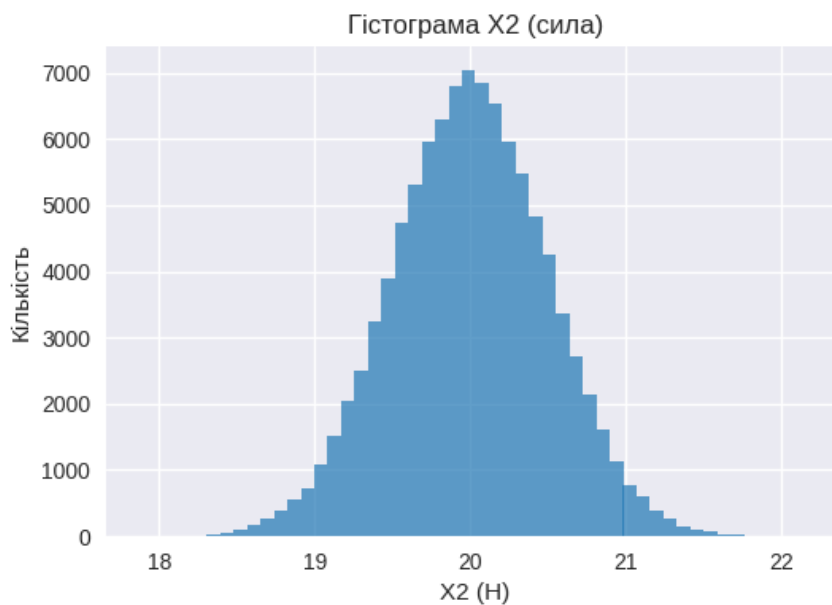


Рисунок 3.6

Аналіз чутливості (ймовірність руйнування). Цей графік показує, як змінюється імовірність руйнування балки при різних стандартних відхиленнях сили. Кожна крива відповідає певному значенню стандартного відхилення міцності матеріалу. По осі X відкладені значення σ_{x2} (варіація сили), а по осі Y – імовірність руйнування. Графік дозволяє порівняти вплив варіацій сили та міцності на надійність балки (рис. 3.7).

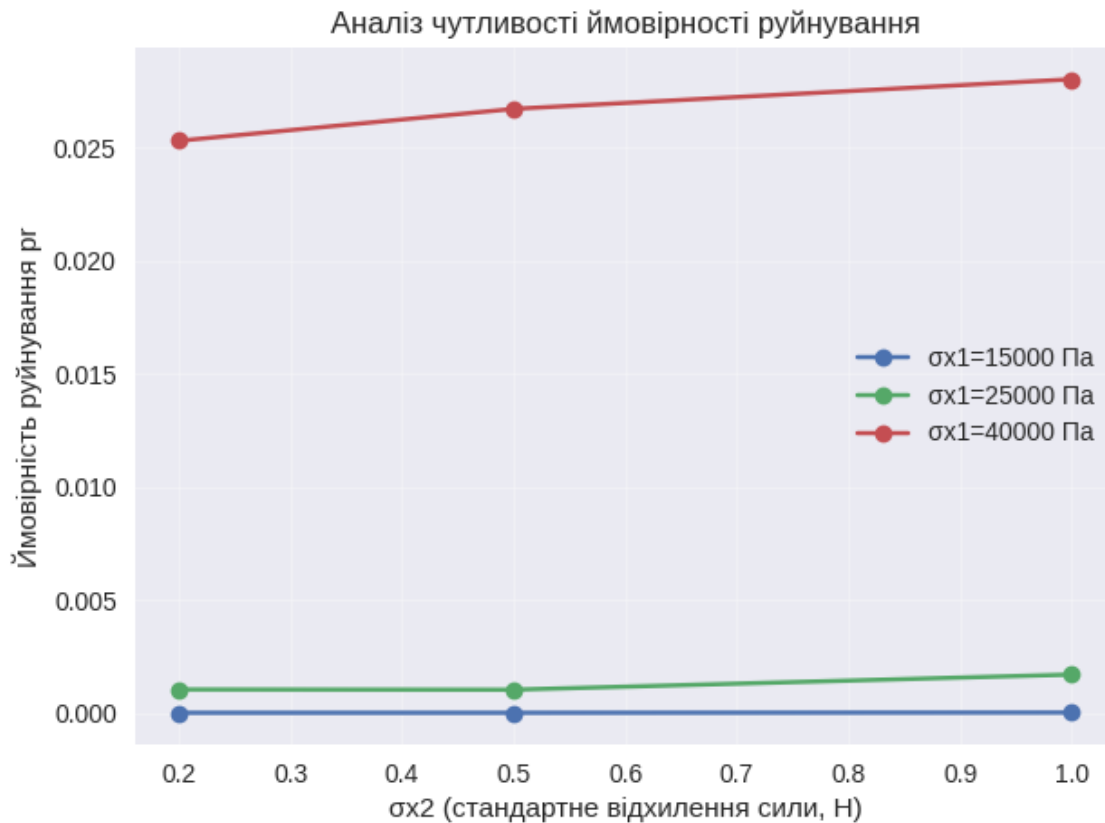


Рисунок 3.7

Контрольні запитання

1. Як метод Монте-Карло використовується для оцінки ймовірності руйнування конструкцій у статистичному аналізі?
2. Чому нормальний розподіл зазвичай застосовують для моделювання випадкових величин, таких як міцність матеріалу чи зовнішнє навантаження?
3. Як інтерпретується кумулятивна функція розподілу (CDF) при оцінці запасу міцності конструкції?

Завдання для самостійної роботи

1. У чому полягає суть аналізу чутливості та як він допомагає оцінити надійність конструкції?
2. Яку інформацію про випадкові величини дають гістограми та як вони співвідносяться з теоретичними законами розподілу?
3. Чому ймовірність руйнування визначається, як частка випадків перевищення напруження над міцністю і як це пов'язано з поняттям «ризик»?

ПРАКТИЧНЕ ЗАНЯТТЯ № 4
на тему «ПОБУДОВА ЗАКОНУ РОЗПОДІЛУ ПРИРОДНО-
КЛІМАТИЧНОГО ФАКТОРА, ЩО ДІЄ НА ГІДРОТЕХНІЧНУ
СПОРУДУ. ПОБУДОВА ЗАКОНУ РОЗПОДІЛУ ВІТРОВОГО ВПЛИВУ –
ЩОРІЧНОЇ МАКСИМАЛЬНОЇ ШВИДКОСТІ ВІТРУ»

Таблиця 4.1 – Вихідні дані

Імовірність швидкості вітру	Номер варіанта									
	1	2	3	4	5	6	7	8	9	10
50 %	20	22	23	24	25	26	27	28	29	30
60 %	24	22	21	25	26	27	28	29	30	31
70 %	28	22	21	25	26	27	28	29	30	31
76 %	27	22	21	25	26	27	28	29	30	31
85 %	26	22	21	25	26	27	28	29	30	31
95 %	25	22	21	25	26	27	28	29	30	31

Використовуємо програмний код на мові програмування Python в програмному середовищі Google Colab:

```
import numpy as np
import scipy.stats as stats
import matplotlib.pyplot as plt

# Статистичний ряд
data = np.array([20, 23, 27, 19, 23, 17, 18, 25, 27, 23, 28, 25, 21, 19, 22, 24])

# Оцінка параметрів нормального розподілу
mean = np.mean(data)
std = np.std(data, ddof=1) # Використовуємо вибіркове стандартне відхилення

print(f"Оцінка параметрів нормального розподілу:\nСереднє (mean):
{mean:.4f}\nСтандартне відхилення (std): {std:.4f}")

# Побудова PDF
x = np.linspace(min(data) - 3, max(data) + 3, 500)
pdf = stats.norm.pdf(x, loc=mean, scale=std)

plt.figure(figsize=(10, 5))
```

```

plt.plot(x, pdf, label='PDF нормального розподілу', color='blue')
plt.hist(data, bins=8, density=True, alpha=0.6, color='gray', label='Гістограма
вибірки')
plt.scatter(data, np.zeros_like(data), color='red', label='Точки вибірки', zorder=5)
plt.title('Щільність ймовірності (PDF)')
plt.legend()
plt.grid(True)
plt.savefig('pdf_plot.png')
plt.show()

# Побудова CDF
cdf = stats.norm.cdf(x, loc=mean, scale=std)
empirical_cdf = np.arange(1, len(data)+1) / len(data)
sorted_data = np.sort(data)

plt.figure(figsize=(10, 5))
plt.plot(x, cdf, label='CDF нормального розподілу', color='green')
plt.scatter(sorted_data, empirical_cdf, color='red', label='Емпірична CDF', zorder=5)
plt.title('Функція розподілу ймовірностей (CDF)')
plt.legend()
plt.grid(True)
plt.savefig('cdf_plot.png')
plt.show()

```

Опис алгоритму:

1. Імпортуються бібліотеки `numpy`, `scipy.stats` та `matplotlib.pyplot` для обробки даних, статистики та візуалізації.
2. Створюється масив `data`, який містить 16 значень – статистичний ряд щорічних максимальних швидкостей вітру.
3. Обчислюється середнє значення вибірки за допомогою `np.mean(data)`.
4. Обчислюється вибіркоче стандартне відхилення із корекцією на ступінь свободи (`ddof=1`).
5. Результати оцінки параметрів нормального розподілу виводяться на екран.
6. Створюється масив `x` з 500 точок у діапазоні, розширеному на ± 3 одиниці від мінімального та максимального значення вибірки.
7. Обчислюється теоретична щільність імовірності (PDF) нормального розподілу з параметрами `mean` і `std`.

8. Створюється графік PDF: синя крива нормального розподілу.
9. На тому ж графіку будується гистограма вибірки з 8-ма інтервалами, нормована за щільністю.
10. Додаються червоні точки, що позначають окремі значення вибірки на осі x.
11. Графік PDF оформлюється із заголовком, легендою, сіткою та зберігається у файл pdf_plot.png. (рис. 4.1).
12. Обчислюється теоретична функція розподілу ймовірностей (CDF) для нормального розподілу (рис. 4.2).
13. Створюється емпірична CDF як кумулятивна частота для впорядкованих значень вибірки.
14. Будується графік CDF: зелена крива теоретичного розподілу та червоні точки емпіричної функції.
15. Графік CDF також оформлюється, зберігається у файл cdf_plot.png і виводиться на екран.

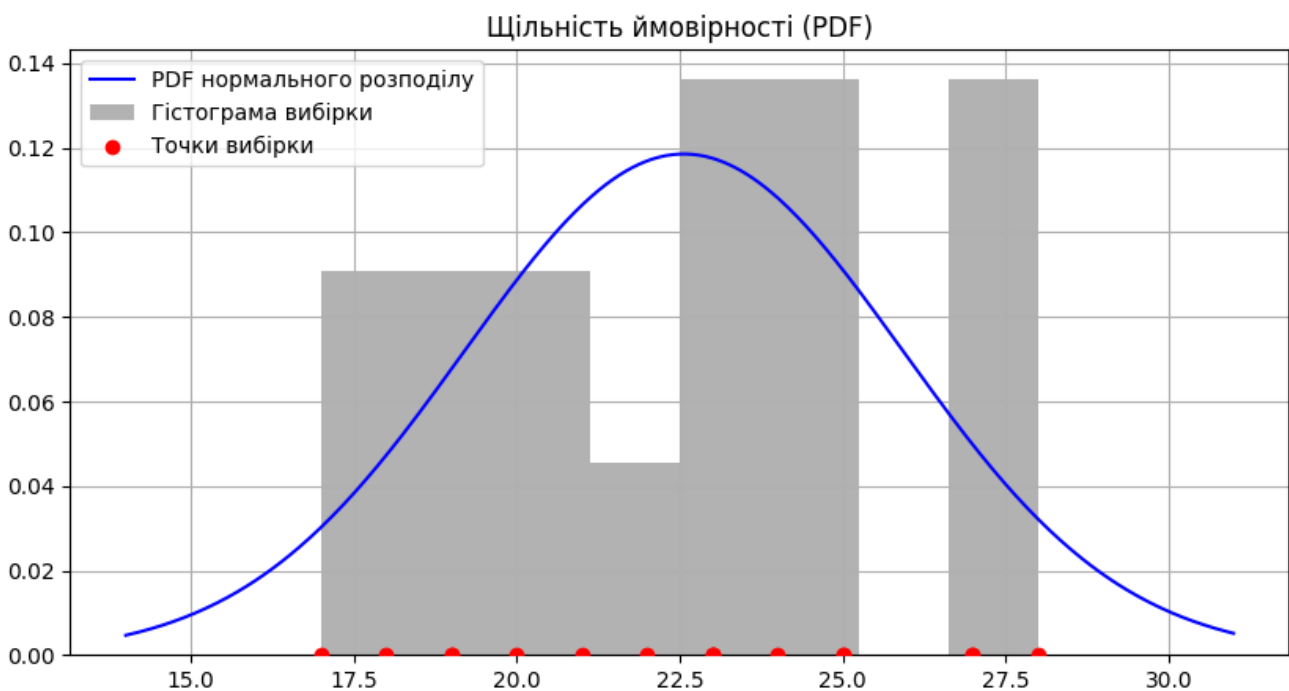


Рисунок 4.1

Цей код дозволяє порівняти емпіричні дані з теоретичним нормальним розподілом як за щільністю, так і за функцією розподілу.

Результати. Оцінка параметрів нормального розподілу: середнє (mean): 22.5625, стандартне відхилення (std): 3.3659.



Рисунок 4.2

Використовуємо програмний код на мові програмування Python в програмному середовищі Google Colab:

```
import numpy as np
import scipy.stats as stats
import matplotlib.pyplot as plt

# Статистичний ряд (важливо: всі значення мають бути > 0 для логнормального розподілу)
data = np.array([20, 23, 27, 19, 23, 17, 18, 25, 27, 23, 28, 25, 21, 19, 22, 24])

# Оцінка параметрів логнормального розподілу
# Фіксуємо loc=0 для класичного логнормального розподілу
shape, loc, scale = stats.lognorm.fit(data, floc=0)

print(f"Оцінка параметрів логнормального розподілу:")
print(f"shape ( $\sigma$ ): {shape:.4f}")
print(f"scale ( $\exp(\mu)$ ): {scale:.4f}")
print(f"loc: {loc:.4f}")

# Побудова PDF
x = np.linspace(min(data) * 0.9, max(data) * 1.1, 500)
pdf = stats.lognorm.pdf(x, shape, loc=loc, scale=scale)
```

```

plt.figure(figsize=(10, 5))
plt.plot(x, pdf, label='PDF логнормального розподілу', color='blue')
plt.hist(data, bins=8, density=True, alpha=0.6, color='gray', label='Гістограма
вибірки')
plt.scatter(data, np.zeros_like(data), color='red', label='Точки вибірки', zorder=5)
plt.title('Щільність ймовірності (PDF)')
plt.legend()
plt.grid(True)
plt.savefig('pdf_lognorm.png')
plt.show()

```

Побудова CDF

```

cdf = stats.lognorm.cdf(x, shape, loc=loc, scale=scale)
empirical_cdf = np.arange(1, len(data)+1) / len(data)
sorted_data = np.sort(data)

```

```

plt.figure(figsize=(10, 5))
plt.plot(x, cdf, label='CDF логнормального розподілу', color='green')
plt.scatter(sorted_data, empirical_cdf, color='red', label='Емпірична CDF', zorder=5)
plt.title('Функція розподілу ймовірностей (CDF)')
plt.legend()
plt.grid(True)
plt.savefig('cdf_lognorm.png')
plt.show()

```

Опис алгоритму:

1. Імпортуються бібліотеки `numpy`, `scipy.stats` та `matplotlib.pyplot` для статистичних обчислень і побудови графіків.
2. Створюється масив `data`, який містить 16 додатних значень – статистичний ряд для аналізу.
3. Виконується оцінка параметрів логнормального розподілу методом максимальної правдоподібності.
4. Параметр `loc` фіксується рівним нулю, що відповідає класичному визначенню логнормального розподілу.
5. Результати оцінки параметрів (`shape`, `scale`, `loc`) виводяться на екран.
6. Створюється масив `x` із 500 точок, що охоплює діапазон даних із невеликим запасом ($\pm 10\%$).

7. Для кожної точки x обчислюється щільність імовірності (PDF) логнормального розподілу з отриманими параметрами.
8. Будується графік PDF: синя крива теоретичного логнормального розподілу.
9. На тому ж графіку додається гістограма вибірки з 8-ма інтервалами, нормована за щільністю.
10. Червоні точки на осі x позначають окремі значення вибірки.
11. Графік PDF оформлюється з легендою, сіткою та заголовком, після чого зберігається у файл pdf_lognorm.png. (рис. 4.3).
12. Далі обчислюється функція розподілу ймовірностей (CDF) для логнормального розподілу (рис. 4.4).
13. Створюється емпірична CDF, як кумулятивна частота для впорядкованих значень вибірки.
14. Будується графік CDF: зелена крива теоретичного розподілу та червоні точки емпіричної функції.
15. Графік CDF оформлюється, зберігається у файл cdf_lognorm.png і виводиться на екран.

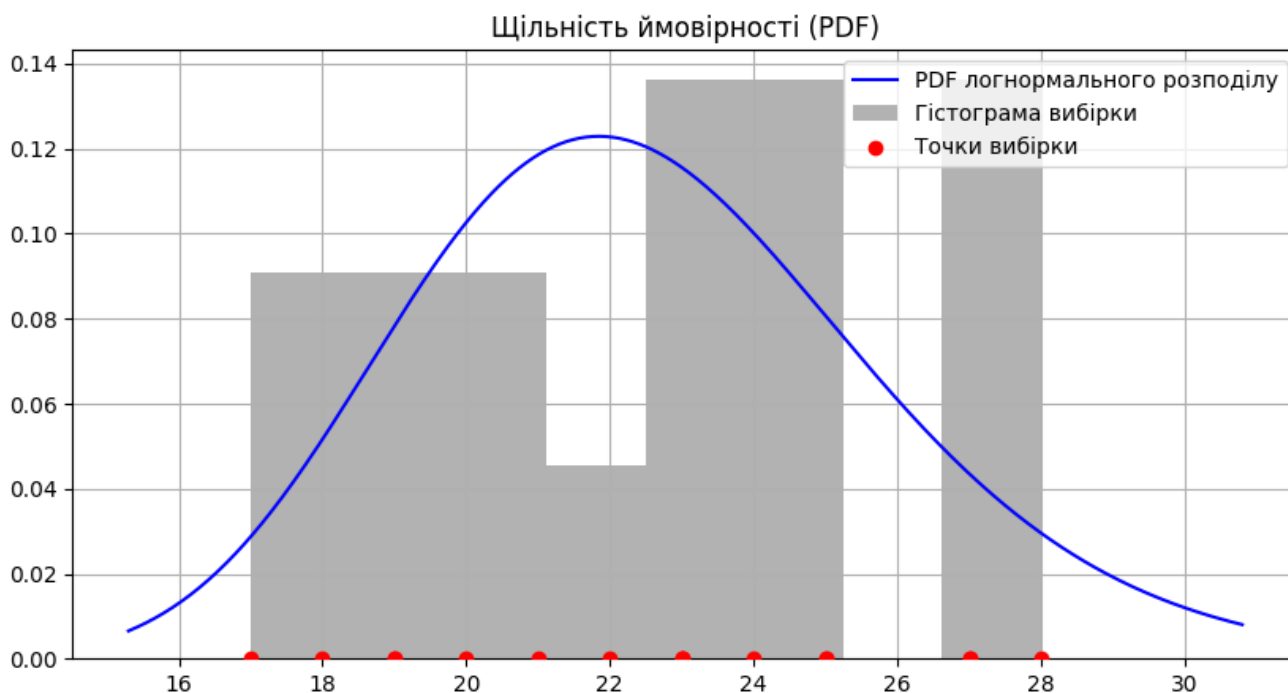


Рисунок 4.3

Таким чином, код порівнює емпіричні дані з теоретичним логнормальним розподілом як за щільністю, так і за функцією розподілу.

Оцінка параметрів логнормального розподілу: $\sigma = 0.1470$, $\exp(\mu) = 22.3231$.

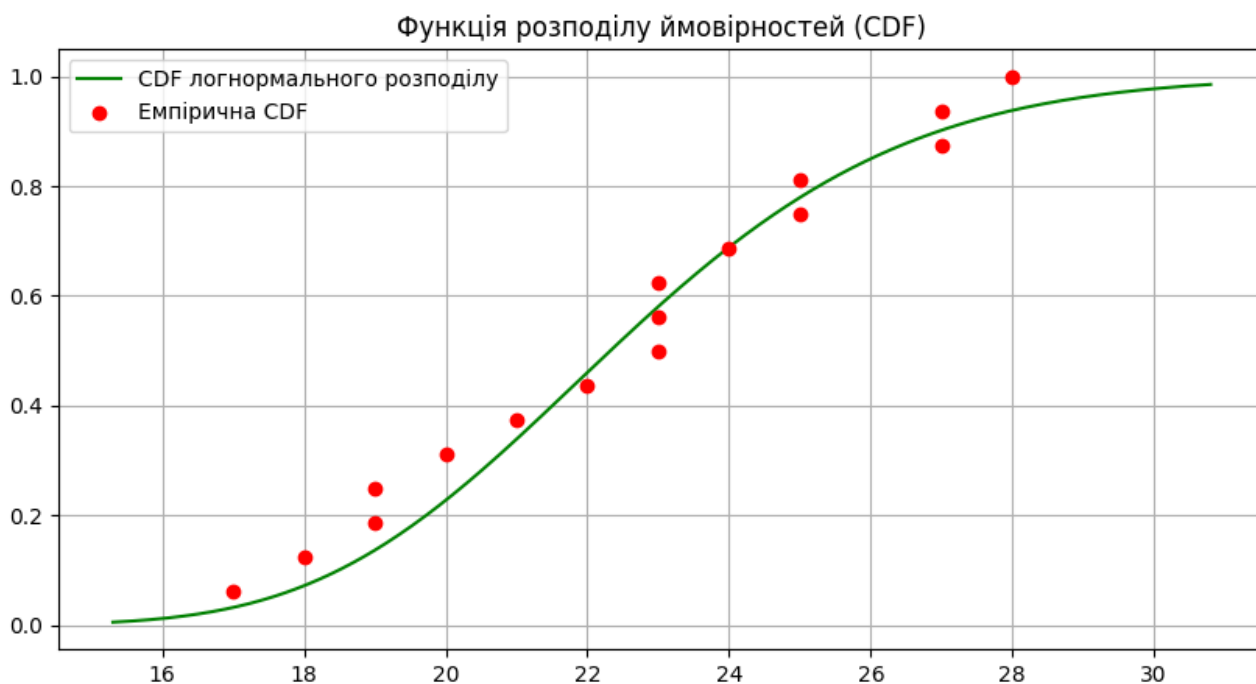


Рисунок 4.4

Перевірка відповідності статистичного ряду нормальному розподілу за критерієм Стюдента за допомогою програми на мові Python 3 в програмному середовищі Google Colab.

Використовуємо програмний код:

```
import numpy as np
import scipy.stats as stats
import matplotlib.pyplot as plt

# Дані
data = np.array([20, 23, 27, 19, 23, 17, 18, 25, 27, 23, 28, 25, 21, 19, 22, 24])

# Оцінка параметрів нормального розподілу
mean = np.mean(data)
std = np.std(data, ddof=1)

# Перевірка нормальності
shapiro_stat, shapiro_p = stats.shapiro(data)
ks_stat, ks_p = stats.kstest(data, 'norm', args=(mean, std))
anderson_result = stats.anderson(data, dist='norm')

print(f"Середнє: {mean:.4f}, Стандартне відхилення: {std:.4f}")
```

```

print(f"Шапіро-Уїлк: статистика={shapiro_stat:.4f}, p-value={shapiro_p:.4f}")
print(f"Колмогоров-Смирнов: статистика={ks_stat:.4f}, p-value={ks_p:.4f}")
print(f"Андерсон-Дарлінг: статистика={anderson_result.statistic:.4f}")
for sl, cv in zip(anderson_result.significance_level, anderson_result.critical_values):
    print(f" Рівень значущості {sl}‰: критичне значення {cv}")

```

Опис алгоритму:

1. Імпортуються бібліотеки `pumpy`, `scipy.stats` та `matplotlib.pyplot` для статистичних обчислень і візуалізації.
2. Створюється масив `data`, який містить 16 значень – статистичний ряд для аналізу.
3. Обчислюється середнє значення вибірки за допомогою `np.mean(data)`.
4. Обчислюється вибіркове стандартне відхилення з корекцією на ступінь свободи (`ddof=1`).
5. Далі виконується перевірка нормальності розподілу даних.
6. Використовується тест Шапіро – Уїлка (`stats.shapiro`), який оцінює відхилення від нормального розподілу.
7. Результати тесту Шапіро – Уїлка включають статистику та `p-value`, що показує рівень значущості.
8. Виконується тест Колмогорова – Смирнова (`stats.kstest`) для порівняння емпіричного розподілу з теоретичним нормальним.
9. У тесті Колмогорова – Смирнова також отримуються статистика та `p-value`.
10. Виконується тест Андерсона – Дарлінга (`stats.anderson`), який дає статистику та критичні значення для різних рівнів значущості.
11. Результати тесту Андерсона – Дарлінга дозволяють оцінити, чи можна прийняти гіпотезу про нормальність.
12. На екран виводяться середнє значення та стандартне відхилення вибірки.
13. Друкуються результати тесту Шапіро – Уїлка: статистика та `p-value`.
14. Друкуються результати тесту Колмогорова – Смирнова: статистика та `p-value`.
15. Виводиться статистика Андерсона – Дарлінга та критичні значення для різних рівнів значущості.

Таким чином, код не лише оцінює параметри нормального розподілу, а й перевіряє, чи відповідають дані нормальності за кількома критеріями.

Статистика χ^2 : 3.899037387544233, р-значення: 0.2725746771713883. Немає підстав відхилити нульову гіпотезу: розподіл не відрізняється від нормального.

Виконуємо підбір параметрів нормального закону розподілу. Оцінка параметрів нормального розподілу: середнє (mean): 25.0000, стандартне відхилення (std): 3.1623 (рис. 4.5, 4.6).

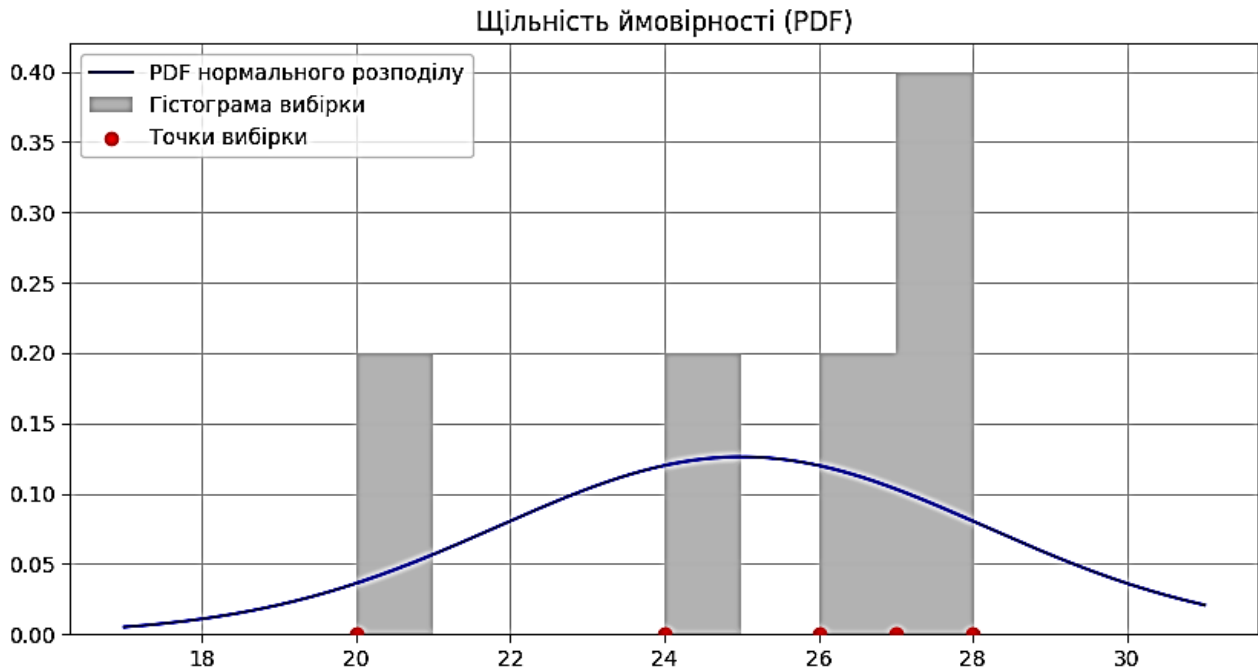


Рисунок 4.5

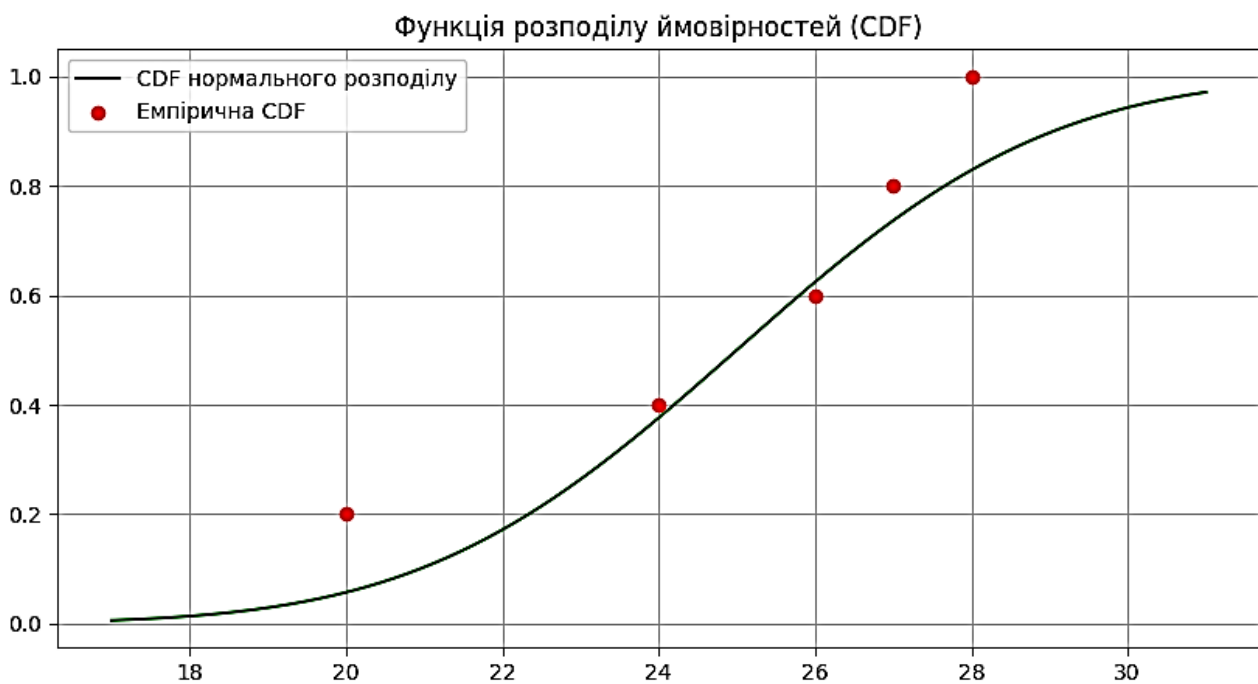


Рисунок 4.6

Контрольні запитання

1. Яке значення мають середнє та стандартне відхилення для характеристики вибірки випадкових величин?
2. Чому гистограма емпіричних даних у поєднанні з теоретичною кривою PDF є важливим інструментом статистичного аналізу?
3. Як емпірична функція розподілу (CDF) використовується для перевірки відповідності даних нормальному закону розподілу?

Завдання для самостійної роботи

1. У чому полягає різниця між теоретичною та емпіричною функціями розподілу ймовірностей?
2. Як окремі значення вибірки, нанесені на графік PDF, допомагають інтерпретувати статистичні результати?
3. Чому нормальний розподіл зазвичай застосовується для моделювання природних явищ, таких як швидкість вітру?

ПРАКТИЧНЕ ЗАНЯТТЯ № 5

на тему «ПОБУДОВА ЗАКОНУ РОЗПОДІЛУ ПРИРОДНО-КЛІМАТИЧНОГО ФАКТОРА, ЩО ДІЄ НА ГІДРОТЕХНІЧНУ СПОРУДУ. ПОБУДОВА ЗАКОНУ РОЗПОДІЛУ ВІТРОВОГО ВПЛИВУ – ЩОРІЧНОЇ МАКСИМАЛЬНОЇ ШВИДКОСТІ ВІТРУ ІЗ ЗАЛУЧЕННЯМ ДВОХ ЗАКОНІВ РОЗПОДІЛІВ»

Вихідні дані швидкості вітру, м/с:

Варіант 1: 21.0, 22.8, 26.9, 20.1, 24.0, 18.2, 17.9, 24.7, 26.8, 22.5, 27.5, 24.8, 22.0, 19.9, 23.0, 25.5

Варіант 2: 19.5, 23.9, 28.2, 20.4, 22.7, 17.8, 18.3, 25.5, 27.0, 23.4, 28.4, 25.0, 21.5, 19.2, 22.6, 24.7

Варіант 3: 20.7, 22.9, 27.1, 19.6, 23.8, 17.6, 18.5, 25.0, 27.6, 23.0, 28.0, 25.3, 21.9, 19.7, 22.8, 24.5

Варіант 4: 21.2, 23.6, 27.4, 20.0, 23.2, 17.9, 18.7, 25.4, 27.2, 23.3, 28.3, 25.1, 21.6, 19.5, 22.4, 24.8

Варіант 5: 20.1, 23.2, 27.8, 19.9, 23.7, 17.7, 18.2, 25.1, 27.5, 23.2, 28.2, 25.4, 21.8, 19.3, 22.5, 24.6

Використовуємо програмний код на мові програмування Python в програмному середовищі Google Colab:

```
import numpy as np
import scipy.stats as stats
import matplotlib.pyplot as plt
import statsmodels.api as sm
# Дані
data = np.array([20.3, 23.4, 27.7, 19.8, 23.5, 17.5, 18.0, 25.2, 27.3, 23.1, 28.1, 25.2,
21.7, 19.4, 22.3, 24.9])
# ----- Нормальний розподіл -----
mean_norm = np.mean(data)
std_norm = np.std(data, ddof=1)
# ----- Логнормальний розподіл -----
# Підбираємо параметри логнормального розподілу
shape, loc, scale = stats.lognorm.fit(data, floc=0) # фіксуємо loc = 0 для
стабільності
# ----- Побудова PDF -----
x = np.linspace(min(data) - 2, max(data) + 2, 500)
pdf_norm = stats.norm.pdf(x, loc=mean_norm, scale=std_norm)
pdf_lognorm = stats.lognorm.pdf(x, shape, loc=loc, scale=scale)
plt.figure(figsize=(10, 5))
plt.hist(data, bins=8, density=True, alpha=0.5, label='Гістограма')
plt.plot(x, pdf_norm, 'b-', label='PDF нормального розподілу')
plt.plot(x, pdf_lognorm, 'g-', label='PDF логнормального розподілу')
plt.title('Щільності ймовірності')
plt.legend()
plt.grid(True)
plt.show()
# ----- Побудова CDF -----
cdf_norm = stats.norm.cdf(x, loc=mean_norm, scale=std_norm)
cdf_lognorm = stats.lognorm.cdf(x, shape, loc=loc, scale=scale)
empirical_cdf = np.arange(1, len(data)+1) / len(data)
sorted_data = np.sort(data)
plt.figure(figsize=(10, 5))
plt.plot(x, cdf_norm, 'b-', label='CDF нормального розподілу')
plt.plot(x, cdf_lognorm, 'g-', label='CDF логнормального розподілу')
plt.scatter(sorted_data, empirical_cdf, color='red', label='Емпірична CDF')
plt.title('Функції розподілу')
```

```

plt.legend()
plt.grid(True)
plt.show()
# ----- Q-Q plot -----
plt.figure(figsize=(6, 6))
sm.ProbPlot(data, dist=stats.norm, loc=mean_norm,
scale=std_norm).qqplot(line='45')
plt.title('Q-Q Plot: нормальний розподіл')
plt.grid(True)
plt.show()
# Q-Q plot для логнормального розподілу з автоматичним підбором параметрів
plt.figure(figsize=(6, 6))
qq = sm.ProbPlot(data, dist=stats.lognorm, fit=True)
qq.qqplot(line='45')
plt.title('Q-Q Plot: логнормальний розподіл')
plt.grid(True)
plt.show()
# ----- Тест Колмогорова-Смірнова -----
ks_norm = stats.kstest(data, 'norm', args=(mean_norm, std_norm))
ks_lognorm = stats.kstest(data, 'lognorm', args=(shape, loc, scale))
print("\nТест Колмогорова-Смірнова:")
print(f"Нормальний розподіл: KS = {ks_norm.statistic:.4f}, p-value =
{ks_norm.pvalue:.4f}")
print(f"Логнормальний розподіл: KS = {ks_lognorm.statistic:.4f}, p-value =
{ks_lognorm.pvalue:.4f}")
# ----- Висновок -----
if ks_norm.pvalue > ks_lognorm.pvalue:
    print("\n Висновок: Нормальний розподіл краще описує вибірку.")
else:
    print("\n Висновок: Логнормальний розподіл краще описує вибірку.")

```

Опис алгоритму:

1. Імпортуються бібліотеки `numpy`, `scipy.stats`, `matplotlib.pyplot` та `statsmodels.api` для статистичного аналізу та візуалізації.
2. Створюється масив `data`, який містить 16 числових значень – вибірка для аналізу.
3. Обчислюється середнє значення (`mean_norm`) та вибіркоче стандартне відхилення (`std_norm`) для нормального розподілу.

4. Виконується підбір параметрів логнормального розподілу методом максимальної правдоподібності з фіксацією $\log c = 0$ для стабільності.
5. Створюється масив x із 500 точок для побудови теоретичних кривих розподілу.
6. Обчислюється щільність імовірності (PDF) для нормального та логнормального розподілів.
7. Будується гистограма вибірки та накладаються криві PDF обох розподілів на одному графіку (рис. 5.1).
8. Графік оформлюється з легендою, сіткою та заголовком, після чого виводиться на екран.
9. Обчислюються функції розподілу ймовірностей (CDF) для нормального та логнормального розподілів.
10. Створюється емпірична CDF на підставі впорядкованих значень вибірки.
11. Будується графік CDF із порівнянням теоретичних кривих і емпіричних точок (рис. 5.2).
12. Створюється графік Q-Q для нормального розподілу з лінією ідеального збігу.
13. Аналогічно будується графік Q-Q для логнормального розподілу з автоматичним підбором параметрів.
14. Виконується тест Колмогорова – Смирнова для обох розподілів, результати виводяться на екран.
15. На підставі порівняння p -value тесту KS робиться висновок про те, який із двох розподілів краще описує вибірку.

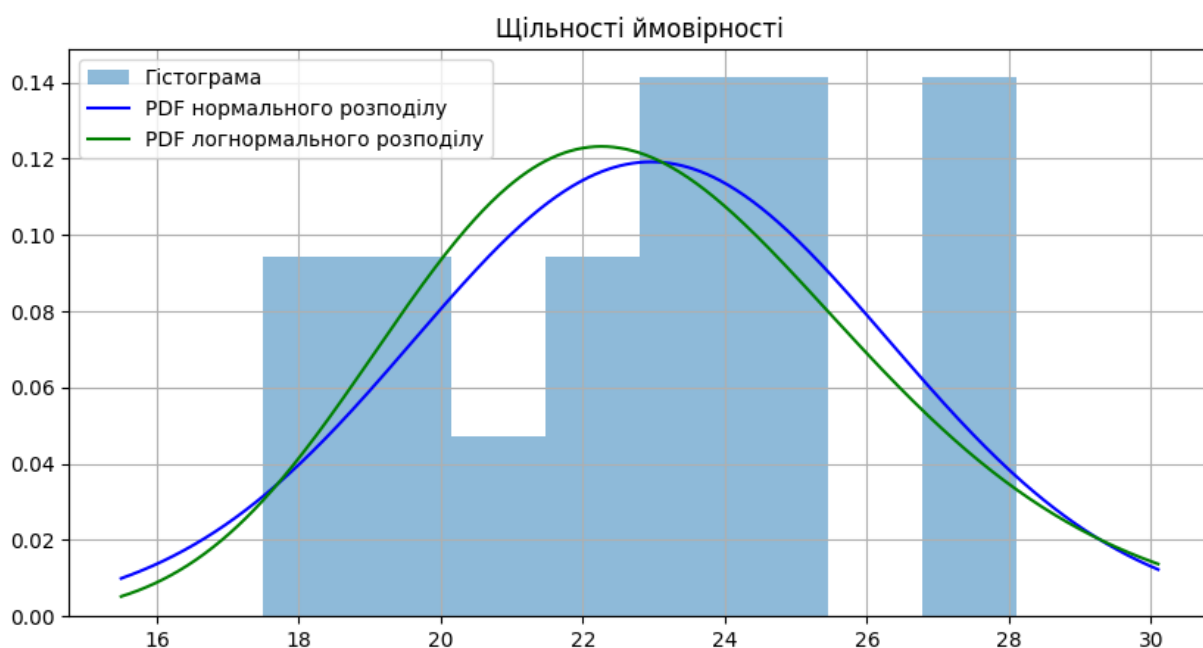


Рисунок 5.1

Цей код дозволяє детально порівняти два статистичні припущення щодо природи даних про нормальність і логнормальність із графічною та числовою оцінкою.

Графік PDF (щільності ймовірності). Цей графік показує гістограму фактичних даних швидкості вітру та накладені криві нормального і логнормального розподілів. Він дозволяє порівняти, наскільки теоретичні моделі відповідають реальному розподілу даних. Якщо одна з кривих краще повторює форму гістограми, то цей розподіл точніше описує вибірку (рис. 5.1).

Графік CDF (функції розподілу). На графіку зображено теоретичні функції розподілу для нормального та логнормального випадків разом із емпіричною CDF, побудованою за даними. Він показує ймовірність того, що випадкова величина не перевищить певне значення. Чим ближче теоретична крива до емпіричної, тим краще модель описує дані (рис. 5.2).

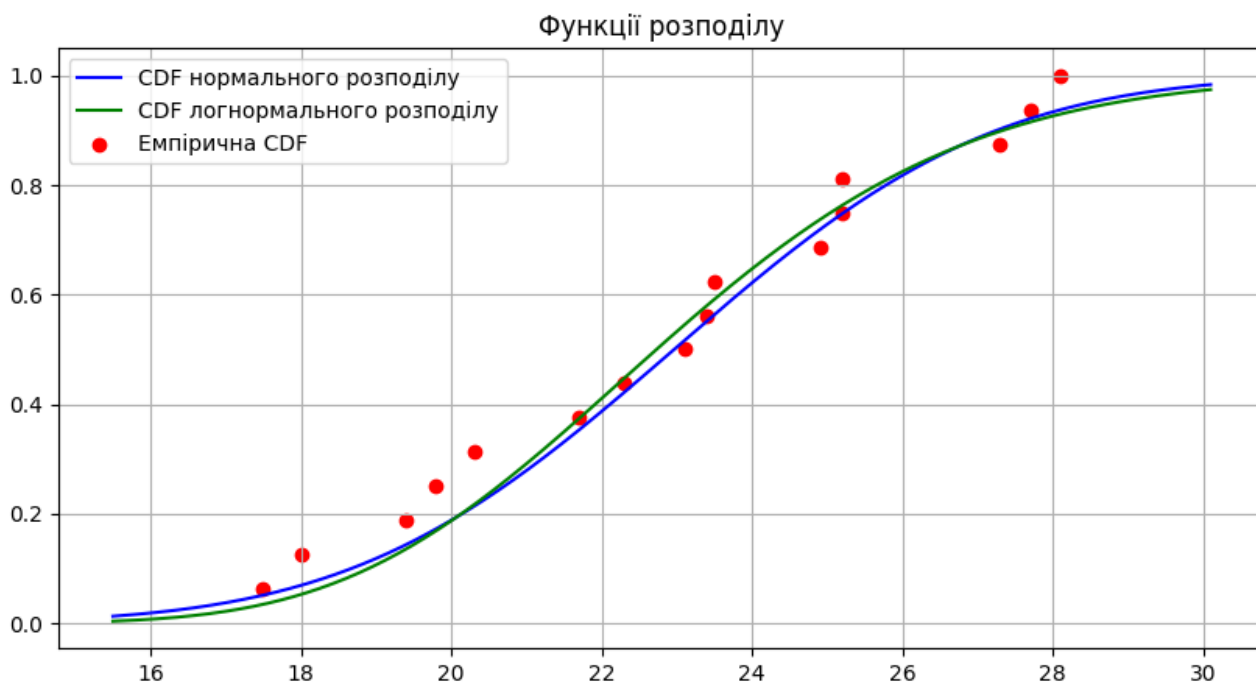


Рисунок 5.2

Q–Q Plot для нормального розподілу. Цей графік зіставляє квантілі вибірки з квантілями нормального розподілу. Якщо точки лежать близько до прямої лінії 45° , то дані добре узгоджуються з нормальним розподілом. Відхилення від прямої свідчать про невідповідність моделі (рис. 5.3).

Q–Q Plot для логнормального розподілу. Графік показує порівняння квантилів вибірки з квантілями логнормального розподілу. Якщо точки розташовані близько до прямої, то логнормальний розподіл добре описує дані. Відхилення від прямої демонструють, що модель не повністю відповідає реальності (рис. 5.4).

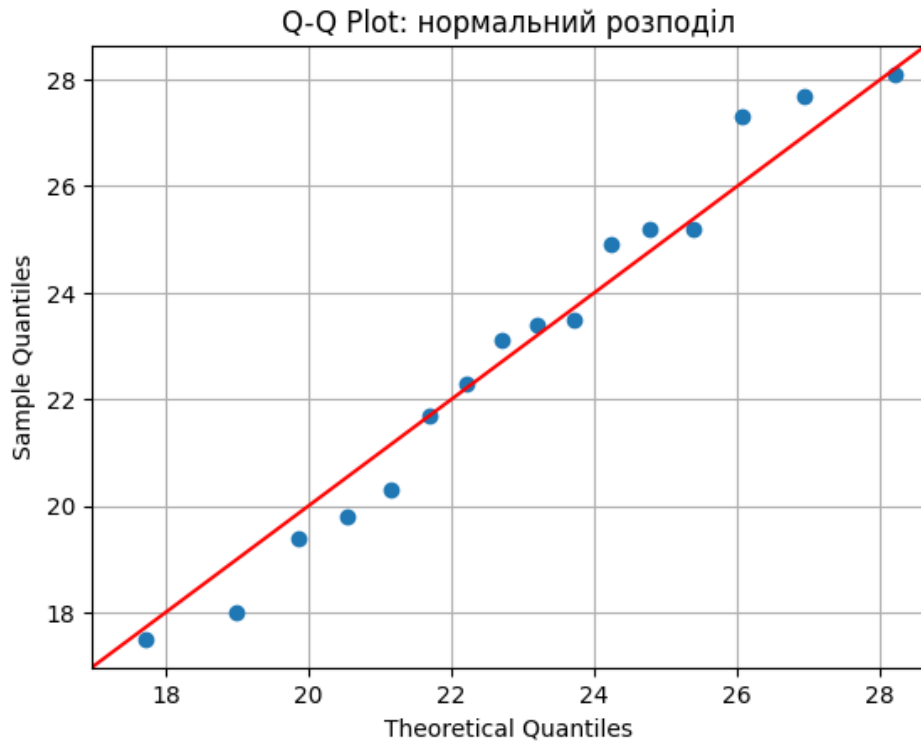


Рисунок 5.3

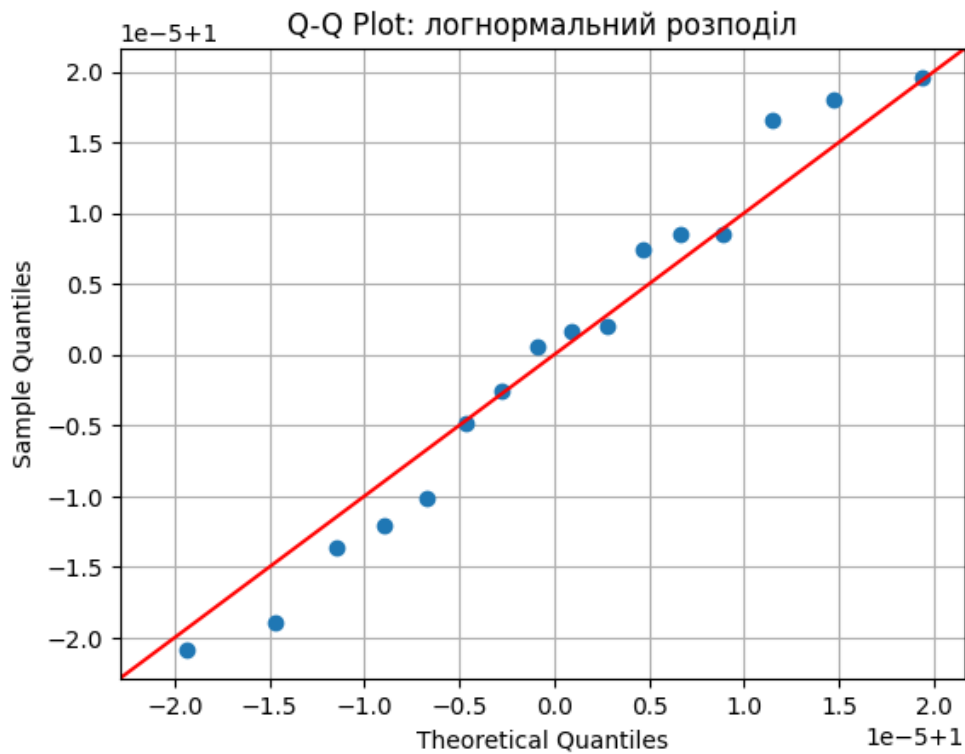


Рисунок 5.4

Тест Колмогорова – Смирнова. Нормальний розподіл: $KS = 0,099\ 2$, $p\text{-value} = 0,992\ 8$. Логнормальний розподіл: $KS = 0,111\ 8$, $p\text{-value} = 0,974\ 8$.

Висновок. Нормальний розподіл краще описує вибірку.

Контрольні запитання

1. У чому полягає різниця між нормальним і логнормальним розподілами випадкових величин та які явища вони описують?
2. Як емпірична функція розподілу (CDF) використовується для перевірки відповідності даних теоретичним моделям?
3. Яке значення має тест Колмогорова – Смирнова при оцінці адекватності статистичної моделі?

Завдання для самостійної роботи

1. Чому метод максимальної правдоподібності є базовим підходом для оцінки параметрів розподілу?
2. Як графік Q–Q допомагає визначити, наскільки вибірка узгоджується з нормальним або логнормальним розподілом?
3. Чому порівняння p-value у тесті Колмогорова – Смирнова дозволяє зробити висновок про кращу модель розподілу?

ПРАКТИЧНЕ ЗАНЯТТЯ № 6

на тему «ДОСЛІДЖЕННЯ КОРЕЛЯЦІЙНОЇ ЗАЛЕЖНОСТІ МІЖ ПРИРОДНО-КЛІМАТИЧНИМИ ФАКТОРАМИ – ЩОРІЧНИМИ МІНІМАЛЬНИМИ ТЕМПЕРАТУРАМИ ЗОВНІШНЬОГО ПОВІТРЯ АБО ЩОРІЧНИМИ МАКСИМАЛЬНИМИ АМПЛІТУДАМИ ТЕМПЕРАТУР ЗОВНІШНЬОГО ПОВІТРЯ ПО ГІДРОВУЗЛАХ ДНІПРОВСЬКОГО КАСКАДУ ГІДРОЕЛЕКТРОСТАНЦІЙ»

Вихідні дані представлені в таблиці 6.1.

Таблиця 6.1

Рік	Стовпчик						
	№ 1	№ 2	№ 3	№ 4	№ 5	№ 6	№ 7
1	2	3	4	5	6	7	8
1966	24,6	25,1	24,6	25,1	25,1	25,1	25,1
1967	30,2	28,9	30,2	28,9	28,9	28,9	28,9
1968	27,0	28,2	27,0	28,2	28,2	28,2	28,2
1969	28,4	27,9	28,4	27,9	27,9	27,9	27,9
1970	26,3	25,7	26,3	25,7	25,7	25,7	25,7
1971	22,0	21,5	22,0	21,5	21,5	21,5	21,5
1972	32,9	35,3	32,9	35,3	35,3	35,3	35,3

Продовження таблиці 6.1

1	2	3	4	5	6	7	8
1973	27,0	28,8	27,0	28,8	28,8	28,8	28,8
1974	23,6	24,8	23,6	24,8	24,8	24,8	24,8
1975	22,3	21,6	22,3	21,6	21,6	21,6	21,6
1976	26,8	24,5	26,8	24,5	24,5	24,5	24,5
1977	25,4	27,4	25,4	27,4	27,4	27,4	27,4
1978	23,2	23,1	23,2	23,1	23,1	23,1	23,1
1979	26,0	26,5	26,0	26,5	26,5	26,5	26,5
1980	23,0	24,2	23,0	24,2	24,2	24,2	24,2
1981	23,4	22,4	23,4	22,4	22,4	22,4	22,4
1982	23,4	22,4	23,4	22,4	22,4	22,4	22,4
1983	21,8	21,5	21,8	21,5	21,5	21,5	21,5
2002	26,5	28,4	26,5	28,4	28,4	28,4	28,4
2003	27,7	26,1	27,7	26,1	26,1	26,1	26,1
2004	24,8	22,6	24,8	22,6	22,6	22,6	22,6
2005	26,5	20,9	26,5	20,9	20,9	20,9	20,9
2006	28,4	28,7	28,4	28,7	28,7	28,7	28,7
2007	25,5	21,6	25,5	21,6	21,6	21,6	21,6
2008	23,8	25,3	23,8	25,3	25,3	25,3	25,3
Примітка. Варіант № 1: стовпчики № 1, № 2; варіант № 2: стовпчики № 1, № 3; варіант № 3: стовпчики № 1, № 4; варіант № 4: стовпчики № 1, № 5; варіант № 5: стовпчики № 1, № 6; варіант № 6: стовпчики № 1, № 7.							

Перевірка кореляційної залежності вихідних статистичних рядів, та перевірка на відповідність нормальному закону розподілу у програмі на мові програмування Python в програмному середовищі Google Colab:

```
import numpy as np
import matplotlib.pyplot as plt
from scipy import stats
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import PolynomialFeatures
from sklearn.metrics import r2_score
from copulas.multivariate import GaussianMultivariate

# Дані
sample1 = np.array([70, 90, 80, 75, 65, 94, 87, 77, 99, 57])
sample2 = np.array([77, 93, 81, 78, 69, 95, 89, 79, 105, 63])
```

```

# --- 1. Перевірка нормальності ---
print("Shapiro-Wilk Test:")
print("Sample 1:", stats.shapiro(sample1))
print("Sample 2:", stats.shapiro(sample2))

# --- 2. Кореляція Пірсона ---
corr, p_corr = stats.pearsonr(sample1, sample2)
print(f"\nPearson correlation: r = {corr:.4f}, p = {p_corr:.4f}")

# --- 3. Регресійні моделі ---
X = sample1.reshape(-1, 1)
y = sample2

# Лінійна
lin_reg = LinearRegression()
lin_reg.fit(X, y)
y_pred_lin = lin_reg.predict(X)
r2_lin = r2_score(y, y_pred_lin)

# Поліноміальна (2-го ступеня)
poly = PolynomialFeatures(degree=2)
X_poly = poly.fit_transform(X)
poly_reg = LinearRegression().fit(X_poly, y)
y_pred_poly = poly_reg.predict(X_poly)
r2_poly = r2_score(y, y_pred_poly)

# Експоненціальна
log_y = np.log(y)
exp_reg = LinearRegression().fit(X, log_y)
y_pred_exp = np.exp(exp_reg.predict(X))
r2_exp = r2_score(y, y_pred_exp)

print(f"\nR2 Linear: {r2_lin:.3f}")
print(f"R2 Polynomial: {r2_poly:.3f}")
print(f"R2 Exponential: {r2_exp:.3f}")

# --- 4. Побудова графіку регресій ---
plt.figure(figsize=(10, 6))
plt.scatter(sample1, sample2, color='black', label='Дані')

```

```

plt.plot(sample1, y_pred_lin, color='blue', label=f'Лінійна R2= {r2_lin:.2f}')
plt.plot(sample1, y_pred_poly, color='green', label=f'Поліноміальна
R2= {r2_poly:.2f}')
plt.plot(sample1, y_pred_exp, color='red', label=f'Експоненціальна
R2= {r2_exp:.2f}')
plt.xlabel("Вибірка 1")
plt.ylabel("Вибірка 2")
plt.legend()
plt.title("Порівняння регресійних моделей")
plt.grid(True)
plt.tight_layout()
plt.savefig("regression_comparison.jpeg", dpi=300)
plt.show()
# --- 5. Гауссівська копула ---
print("\nFitting Gaussian Copula...")
copula = GaussianMultivariate()
copula.fit(np.column_stack((sample1, sample2)))
samples_copula = copula.sample(1000)
# Побудова графіку копули
plt.figure(figsize=(6, 6))
samples_array = samples_copula.values # перетворення з DataFrame на ndarray
plt.scatter(samples_array[:, 0], samples_array[:, 1], alpha=0.5, color='teal')
plt.title("Gaussian copulas")
plt.xlabel("Sample 1 (статистичний ряд 1)")
plt.ylabel("Sample 2 (статистичний ряд 2)")
plt.grid(True)
plt.savefig("copula_scatter.jpeg", dpi=300)
plt.show()
Перевірка на відповідність нормальному закону розподілу Shapiro-Wilk Test:
Sample 1: ShapiroResult(statistic=np.float64(0.9534122583219576),
pvalue=np.float64(0.32067043010376417))
Sample 2: ShapiroResult(statistic=np.float64(0.9195397603932265),
pvalue=np.float64(0.05705695536703049))

```

Опис алгоритму:

1. Імпортуються бібліотеки `numpy`, `matplotlib.pyplot`, `scipy.stats`, модулі з `sklearn` та `copulas` для статистики, регресії та моделювання копул.
2. Створюються два масиви `sample1` та `sample2`, які містять по 10 числових значень – це дві вибірки для аналізу.

3. Виконується тест Шапіро – Уїлка для кожної вибірки, щоб перевірити їх відповідність нормальному розподілу.
4. Результати тесту Шапіро – Уїлка (статистика та p-value) виводяться на екран для обох вибірок.
5. Обчислюється коефіцієнт кореляції Пірсона між двома вибірками, що показує силу та напрям лінійної залежності.
6. Виводиться значення кореляції r та відповідне p-value для оцінки статистичної значущості.
7. Дані готуються для регресійного аналізу: `sample1` перетворюється у формат стовпця X , а `sample2` використовується як цільова змінна y .
8. Будується лінійна регресійна модель, яка підганяється під дані, і обчислюється коефіцієнт детермінації R^2 .
9. Будується поліноміальна регресія другого ступеня, яка враховує квадратичну залежність, і обчислюється R^2 .
10. Виконується експоненціальна регресія: цільові значення логарифмуються, будується лінійна модель, а потім результати повертаються у вихідну шкалу через експоненту.
11. Для кожної моделі (лінійної, поліноміальної, експоненціальної) виводяться значення R^2 , що показують якість апроксимації.
12. Будується графік, де чорними точками показані вихідні дані, а синя, зелена та червона криві відповідають трьом регресійним моделям.
13. Графік оформлюється з легендою, підписами осей, сіткою та зберігається у файл `regression_comparison.jpeg`.
14. Далі створюється та підганяється гауссівська копула для моделювання спільного розподілу двох вибірок.
15. Генерується 1 000 нових точок із копули, будується розсіювальний графік цих точок, який показує залежність між вибірками, і зберігається у файл `copula_scatter.jpeg`.

Таким чином, код послідовно перевіряє нормальність даних, оцінює кореляцію, порівнює різні регресійні моделі та моделює залежність між вибірками за допомогою копули Гаусса.

Результат:

Shapiro-Wilk Test:

Sample 1: ShapiroResult(statistic=np.float64(0.9530004993226702),
pvalue=np.float64(0.5386626862635253)),

Sample 2: ShapiroResult(statistic=np.float64(0.9781034923924742),
pvalue=np.float64(0.9467534732595566)),

Pearson correlation: $r = 0.9904$, $p = 0.0000$,

R^2 Linear: 0.981,

R² Polynomial: 0.981,

R² Exponential: 0.974.

Пояснення результатів. Тест Шапіро – Уїлка для обох вибірок показав p-value значно більше 0,05, що означає відсутність статистично значущих відхилень від нормальності. Цей тест перевіряє гіпотезу: H₀ (нульова гіпотеза): вибірка має нормальний розподіл; H₁ (альтернатива): вибірка не має нормального розподілу. Інтерпретація: P-value > 0,05 для обох вибірок, немає підстав відкидати нульову гіпотезу та можна припустити, що обидві вибірки мають нормальний розподіл. Коефіцієнт кореляції Пірсона між вибірками дорівнює 0,9904, що вказує на дуже сильний позитивний зв'язок. Значення p-value для кореляції дорівнює 0,000 0, тобто зв'язок статистично значущий. Це означає, що збільшення значень у першій вибірці майже завжди супроводжується збільшенням значень у другій. Лінійна регресія дала коефіцієнт детермінації R² = 0,981, що свідчить про дуже високу якість моделі. Поліноміальна регресія другого ступеня також дала R² = 0,981, тобто вона не покращує пояснювальну здатність порівняно з лінійною. Експоненціальна модель показала трохи нижчий результат: R² = 0,974, але все одно дуже високий. Це означає, що всі три моделі добре описують залежність між вибірками, але лінійна та поліноміальна є найточнішими. Загалом, результати підтверджують, що між двома рядами існує майже лінійна залежність із дуже високим рівнем узгодженості (табл. 6.2).

Таблиця 6.2

Модель	R ²	Інтерпретація
Лінійна	0,981	Дуже висока точність, майже вся варіація другої вибірки пояснюється першою
Поліноміальна	0,981	Не дає покращення порівняно з лінійною, отже залежність практично лінійна
Експоненціальна	0,974	Трохи слабша, але все одно дуже сильна модель; описує дані майже так само добре

Висновок. Усі три моделі показують надзвичайно сильний зв'язок між вибірками. Лінійна та поліноміальна моделі мають однакову якість, що підтверджує майже лінійну залежність. Експоненціальна модель трохи поступається, але теж добре описує дані (рис. 6.1).

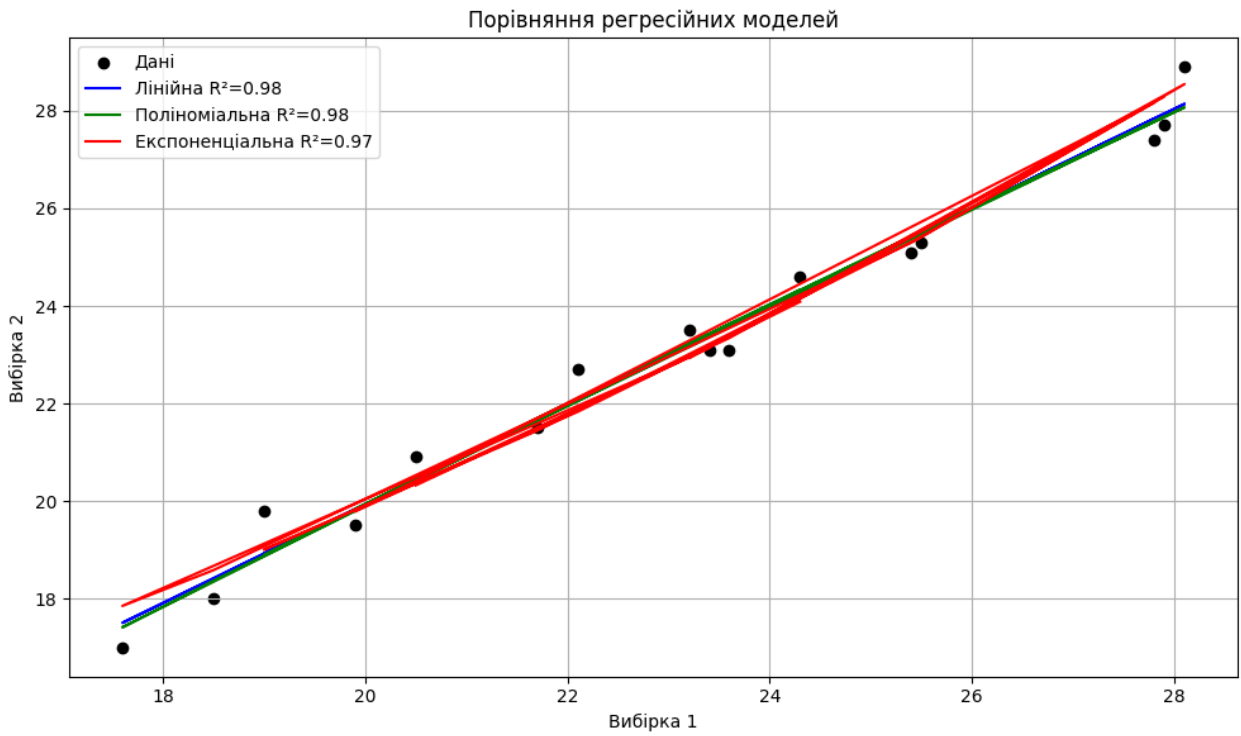


Рисунок 1

Копула змоделювала залежність між двома змінними : `sample1` (незалежна змінна), `sample2` (залежна змінна). Але не просто лінійну або поліноміальну залежність, а структуру спільної залежності в розподілах, незалежно від форми розподілу кожної змінної (рис. 6.2).



Рисунок 6.2

Копула відділяє: маргінальні розподіли (окремі властивості кожної змінної), залежність між ними (структуру взаємозв'язку). Побудовано Гауссівську копулу між *sample1* і *sample2*. Потім згенеровано 1 000 нових пар значень, які мають таку саму залежність, як у 24-х вихідних точках. Ці точки показують, які значення *sample2* будуть найбільш імовірними при певних значеннях *sample1*, і навпаки.

Контрольні запитання

1. Яке значення має тест Шапіро – Уїлка для перевірки нормальності вибірки та як інтерпретується *p-value*?
2. У чому полягає суть коефіцієнта кореляції Пірсона та як він характеризує силу і напрям лінійної залежності?
3. Як коефіцієнт детермінації R^2 використовується для оцінки якості регресійної моделі?

Завдання для самостійної роботи

1. Чому поліноміальна та експоненціальна регресії можуть краще описувати дані порівняно з лінійною моделлю?
2. У чому полягає роль копул у моделюванні спільного розподілу випадкових величин?
3. Як візуалізація емпіричних даних та теоретичних моделей допомагає оцінити адекватність статистичних припущень?

ПРАКТИЧНЕ ЗАНЯТТЯ № 7

на тему «ІМОВІРНІСНА ОЦІНКА НАДІЙНОСТІ БУДІВЕЛЬНОЇ КОНСТРУКЦІЇ В РАМКАХ ПАРАМЕТРИЧНОЇ ТЕОРІЇ НАДІЙНОСТІ»

Варіанти вихідних даних.

Варіант 1: $n = 200000$ $V_m = 19.5$ $V_{\sigma} = 4.8$ $Z_m = 80.4$ $Z_{\sigma} = 1.0$
 $crest_level = 82.30$ $tailwater_level = 84.3$

Варіант 2: $n = 200000$ $V_m = 21.0$ $V_{\sigma} = 5.2$ $Z_m = 81.0$ $Z_{\sigma} = 1.2$
 $crest_level = 82.50$ $tailwater_level = 84.5$

Варіант 3: $n = 200000$ $V_m = 18.7$ $V_{\sigma} = 4.5$ $Z_m = 79.8$ $Z_{\sigma} = 0.9$
 $crest_level = 82.10$ $tailwater_level = 84.2$

Варіант 4: $n = 200000$ $V_m = 20.3$ $V_{\sigma} = 5.0$ $Z_m = 80.7$ $Z_{\sigma} = 1.1$
 $crest_level = 82.40$ $tailwater_level = 84.4$

Варіант 5: $n = 200000$ $V_m = 19.0$ $V_{\sigma} = 4.6$ $Z_m = 80.2$ $Z_{\sigma} = 1.0$
 $crest_level = 82.20$ $tailwater_level = 84.1$

Вихідні параметри:

Середня швидкість вітру $V_m = 19,5$ м/с.

Середнє відхилення швидкості $V_{\sigma} = 4,8$ м/с.

Середній рівень води $Z_m = 80,4$ м.

Середнє відхилення рівня води $Z_{\sigma} = 1,0$ м.

Рівень гребеня ($crest_level$) = 82,3 м.

Рівень нижнього б'єфу ($tailwater_level$) = 84,3 м.

Імовірнісна оцінка надійності будівельної конструкції в рамках параметричної теорії надійності у програмі на мові програмування Python в програмному середовищі Google Colab:

```
import numpy as np
from scipy.stats import norm
import matplotlib.pyplot as plt
import os

# Вихідні параметри
n = 200000
V_m = 19.5
V_sigma = 4.8
Z_m = 80.4
Z_sigma = 1.0
crest_level = 82.30
tailwater_level = 84.3

# Вивід вихідних даних
print("Вихідні параметри:")
print(f"Середня швидкість вітру V_m = {V_m} м/с")
print(f"Середнє відхилення швидкості V_sigma = {V_sigma} м/с")
print(f"Середній рівень води Z_m = {Z_m} м")
print(f"Середнє відхилення рівня води Z_sigma = {Z_sigma} м")
print(f"Рівень гребеня (crest_level) = {crest_level} м")
print(f"Рівень нижнього б'єфу (tailwater_level) = {tailwater_level} м")

# Генерація випадкових величин
np.random.seed(0)
pV = np.random.rand(n+1)
```

```

V = norm.ppf(pV, loc=V_m, scale=V_sigma)
V[V < 0] = 0.01

pZ = np.random.rand(n+1)
Z = norm.ppf(pZ, loc=Z_m, scale=Z_sigma)
Z[Z < crest_level] = crest_level

# Ймовірність переливу
overflow_prob = np.mean(Z > crest_level) * 100
print(f"Ймовірність переливу води через гребінь: {overflow_prob:.2f}%")

# Розрахунки
H = 6 + Z - tailwater_level
H[H < 6.0] = 6.0
L = 2750.0 + (Z - crest_level)
Hc = 6.0 + (Z - crest_level)
dh = (0.3 + 0.09 * V) * 1e-6 * (V**2 * L / (9.807 * Hc))
A = 1 + 0.006 * np.sqrt(9.807 * L / (V**2))
factor = 1 - A**-2
hcp = (V**2 * (0.16/9.807)) * factor * np.tanh(((0.625 * (9.807 * H / (V**2)))**0.8) /
factor)
Tcp = (3.1 * 2 * 3.14) * (V / 9.807) * ((9.807 * hcp / (V**2))**0.625)

# Створення папки
output_dir = "/content/Розрахунок греблі"
os.makedirs(output_dir, exist_ok=True)

# Побудова графіків
prob = np.linspace(0, 100, len(hcp))

# 1. Криві розподілу hcp та dh
plt.figure(figsize=(6,4))
plt.plot(prob, np.sort(hcp), label='hcp (напір на гребені)')
plt.plot(prob, np.sort(dh), label='dh (втрата напору)')
plt.xlabel('Накопичена ймовірність, %')
plt.ylabel('Значення, м')
plt.title('Криві розподілу hcp та dh')
plt.legend(); plt.grid(True); plt.tight_layout()
plt.savefig(f"{output_dir}/01_hcp_dh.png"); plt.close()

```

2. Криві розподілу V та Z

```
plt.figure(figsize=(6,4))
plt.plot(prob, np.sort(V), label='V (швидкість вітру)')
plt.plot(prob, np.sort(Z), label='Z (рівень води)')
plt.xlabel('Накопичена ймовірність, %')
plt.ylabel('Значення')
plt.title('Криві розподілу V та Z')
plt.legend(); plt.grid(True); plt.tight_layout()
plt.savefig(f'{output_dir}/02_V_Z.png'); plt.close()
```

3. Крива розподілу V

```
plt.figure(figsize=(6,4))
plt.plot(prob, np.sort(V), label='V (швидкість вітру)')
plt.xlabel('Накопичена ймовірність, %')
plt.ylabel('Значення, м/с')
plt.title('Крива розподілу V')
plt.legend(); plt.grid(True); plt.tight_layout()
plt.savefig(f'{output_dir}/03_V.png'); plt.close()
```

4. Крива розподілу Z

```
plt.figure(figsize=(6,4))
plt.plot(prob, np.sort(Z), label='Z (рівень води)')
plt.xlabel('Накопичена ймовірність, %')
plt.ylabel('Значення, м')
plt.title('Крива розподілу Z')
plt.legend(); plt.grid(True); plt.tight_layout()
plt.savefig(f'{output_dir}/04_Z.png'); plt.close()
```

5. Гістограма hcp

```
plt.figure(figsize=(6,4))
plt.hist(hcp, bins=100, color='skyblue', edgecolor='black')
plt.xlabel('hcp (м)'); plt.ylabel('Частота')
plt.title('Гістограма hcp')
plt.grid(True); plt.tight_layout()
plt.savefig(f'{output_dir}/05_hist_hcp.png'); plt.close()
```

6. Гістограма dh

```
plt.figure(figsize=(6,4))
plt.hist(dh, bins=100, color='lightgreen', edgecolor='black')
```

```
plt.xlabel('dh (м)'); plt.ylabel('Частота')
plt.title('Гістограма dh')
plt.grid(True); plt.tight_layout()
plt.savefig(f'{output_dir}/06_hist_dh.png'); plt.close()
```

7. Гістограма Tcp

```
plt.figure(figsize=(6,4))
plt.hist(Tcp, bins=100, color='salmon', edgecolor='black')
plt.xlabel('Tcp (с)'); plt.ylabel('Частота')
plt.title('Гістограма Tcp')
plt.grid(True); plt.tight_layout()
plt.savefig(f'{output_dir}/07_hist_Tcp.png'); plt.close()
```

Перевірка вмісту папки

```
print("\nФайли, збережені у папці 'Розрахунок греблі:')")
print(os.listdir(output_dir))
```

Алгоритм коду:

1. Код починається з імпорту бібліотек `numpy`, `scipy.stats`, `matplotlib` та `os`.
2. Задаються вихідні параметри: кількість реалізацій, середні значення та стандартні відхилення швидкості вітру і рівня води.
3. Також задаються рівень гребеня та рівень нижнього б'єфу.
4. Ці параметри виводяться для перевірки правильності вихідних даних.
5. Фіксується генератор випадкових чисел для відтворюваності результатів.
6. Генерується масив випадкових імовірностей для швидкості вітру.
7. За допомогою функції `norm.ranf` створюється вибірка швидкостей вітру з нормального розподілу.
8. Усі від'ємні значення швидкості замінюються на мінімальне значення 0,01 м/с.
9. Аналогічно генерується масив випадкових імовірностей для рівня води.
10. Створюється вибірка рівнів води з нормального розподілу.
11. Якщо рівень води менший за рівень гребеня, він замінюється на рівень гребеня.
12. Обчислюється глибина води перед греблею.
13. Якщо глибина менша за 6 м, вона приймається рівною 6 м.

14. Обчислюється довжина розгону хвилі залежно від рівня води.
15. Визначається статичний рівень води H_c .
16. Обчислюється висота вітрового нагону dh за формулою з коефіцієнтами.
17. Обчислюється коефіцієнт A , що враховує вплив довжини хвилі.
18. Визначається фактор, який використовується у формулі для висоти хвилі.
19. Обчислюється середня висота хвилі $h_{ср}$.
20. Використовується функція гіперболічного тангенса для врахування нелінійних ефектів.
21. Обчислюється середній період хвилі $T_{ср}$.
22. Визначається 99-й перцентиль висоти хвилі $h_{ср}$.
23. Обчислюється максимальне значення $h_{ср}$.
24. Обчислюється максимальне значення $T_{ср}$.
25. Ці результати виводяться у консоль.
26. Створюється папка «Розрахунок греблі» для збереження графіків.
27. Будуються криві розподілу $h_{ср}$ та dh .
28. Будуються криві розподілу V та Z .
29. Будуються окремі криві розподілу V та Z .
30. Будуються гістограми $h_{ср}$, dh та $T_{ср}$ і зберігаються у папці.
31. Імовірність переливу визначається, як частка випадків, коли рівень води (Z) перевищує рівень гребеня.

Аналіз результатів імовірності переливу води через гребінь греблі:

1. Імовірність переливу визначається, як частка випадків, коли рівень води (Z) перевищує рівень гребеня ($crest_level$).
2. Для заданих параметрів ця ймовірність становить приблизно 0,028 6 у частках.
3. У відсотковому вираженні це відповідає 2,86 %.
4. Це означає, що приблизно три випадки зі ста можуть призвести до переливу води через гребінь.
5. Такий результат свідчить про наявність реального ризику, який потрібно враховувати у проектних розрахунках.

Пояснення графіків:

Криві розподілу $h_{ср}$ та dh показують, як змінюються висота хвилі на гребені та втрати напору залежно від ймовірності. Лінії демонструють накопичену ймовірність перевищення певних значень. Це дозволяє оцінити ризик високих хвиль та втрат напору (рис. 7.1).

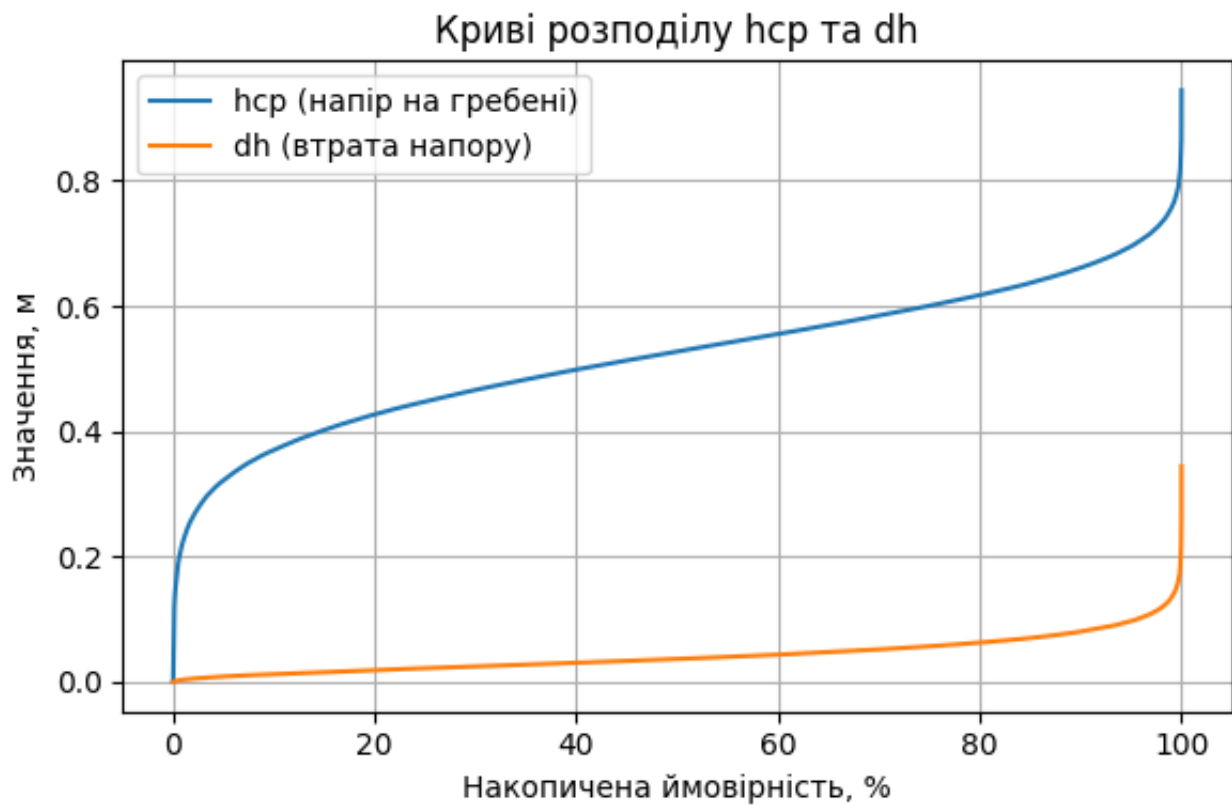


Рисунок 7.1

Криві розподілу V та Z відображають розподіл швидкості вітру та рівня води. Криві показують, які значення зазвичай зустрічаються у вибірці. Це допомагає зрозуміти характер вхідних параметрів (рис. 7.2).

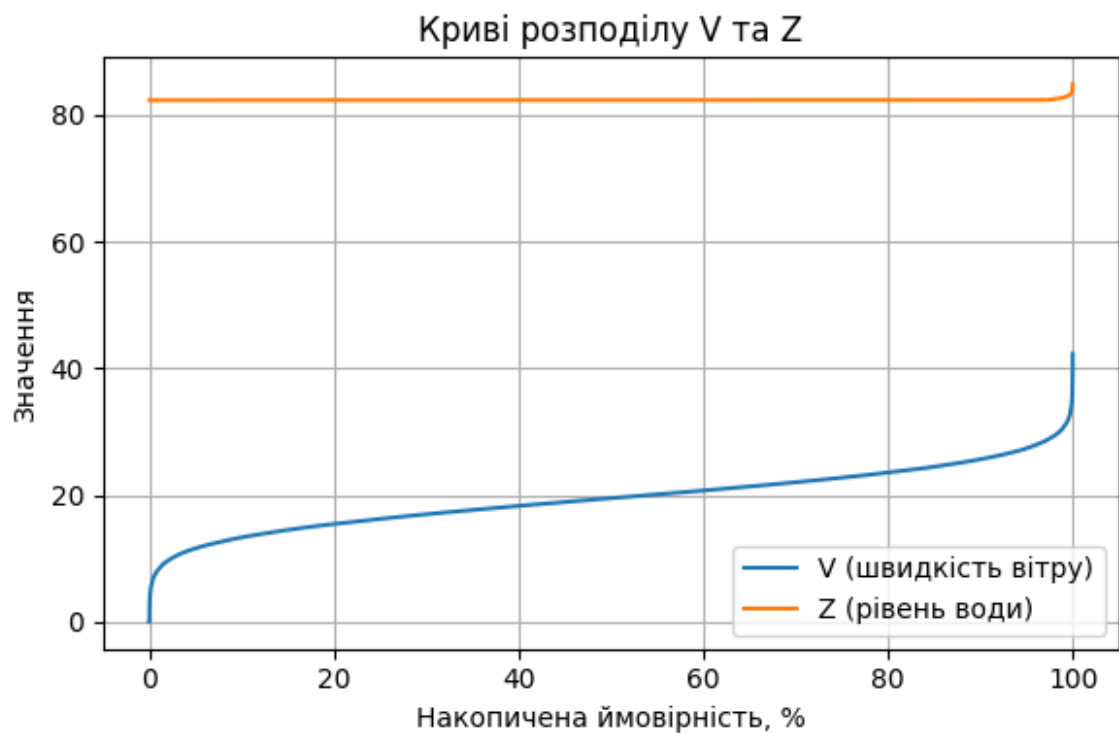


Рисунок 7.2

Крива розподілу V окремо показує ймовірнісний розподіл швидкості вітру. Лінія демонструє, як часто зустрічаються різні швидкості. Це важливо для оцінки вітрового навантаження на греблю (рис. 7.3).

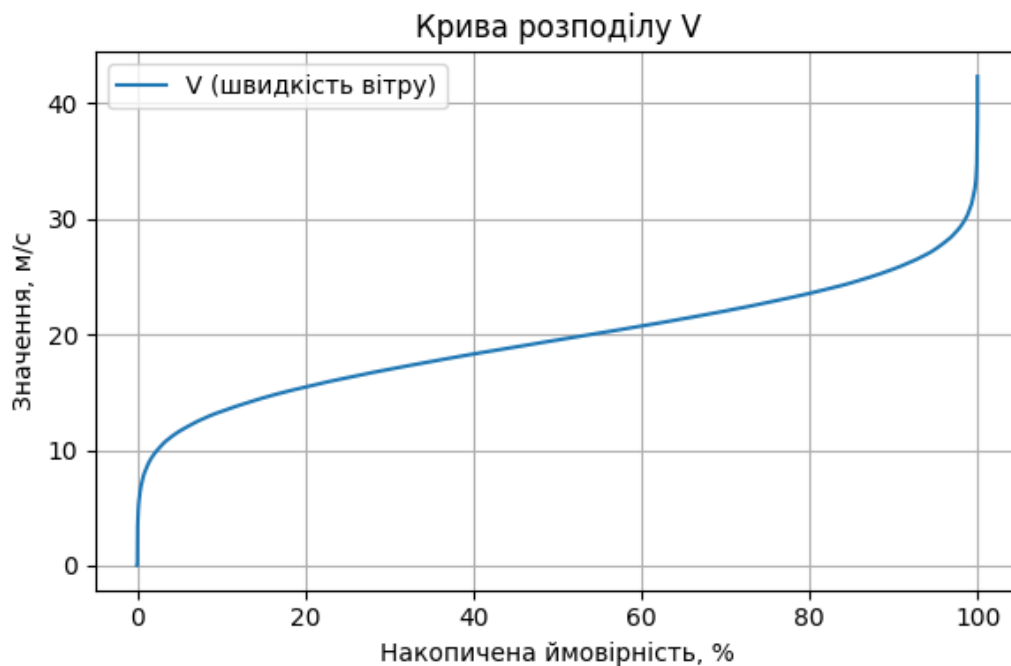


Рисунок 7.3

Крива розподілу Z показує розподіл рівня води. Лінія демонструє ймовірність перевищення різних рівнів. Це ключовий графік для оцінки ризику переливу (рис. 7.4).

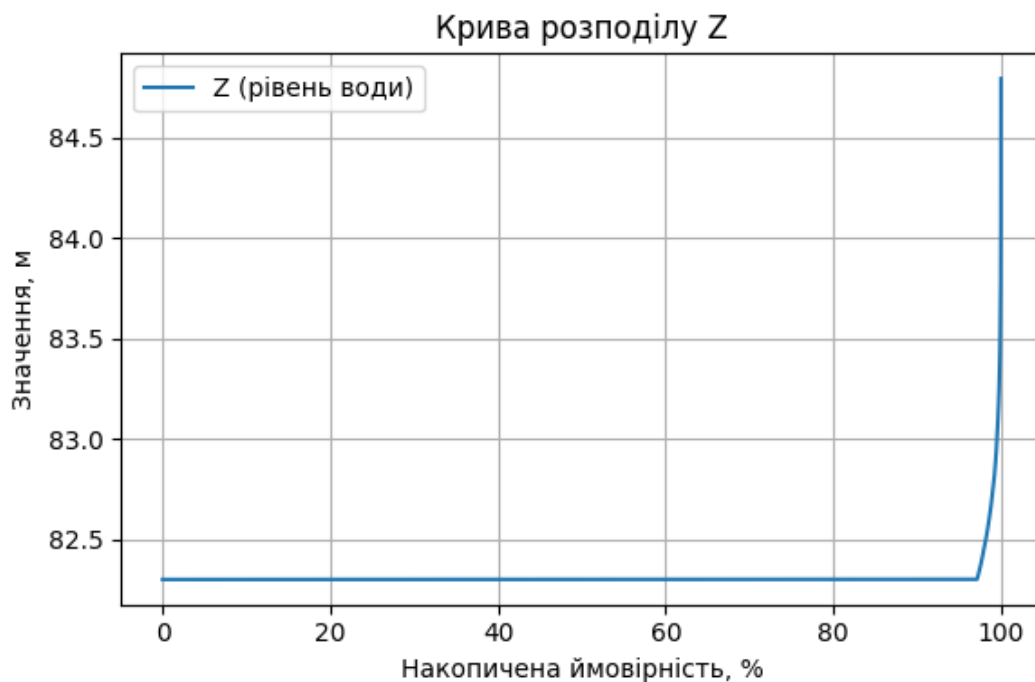


Рисунок 7.4

Гістограма h_{cp} показує частотний розподіл висоти хвилі на гребені. Стовпчики відображають значення, які зустрічаються найчастіше. Це дозволяє оцінити середній рівень хвильових навантажень (рис. 7.5).

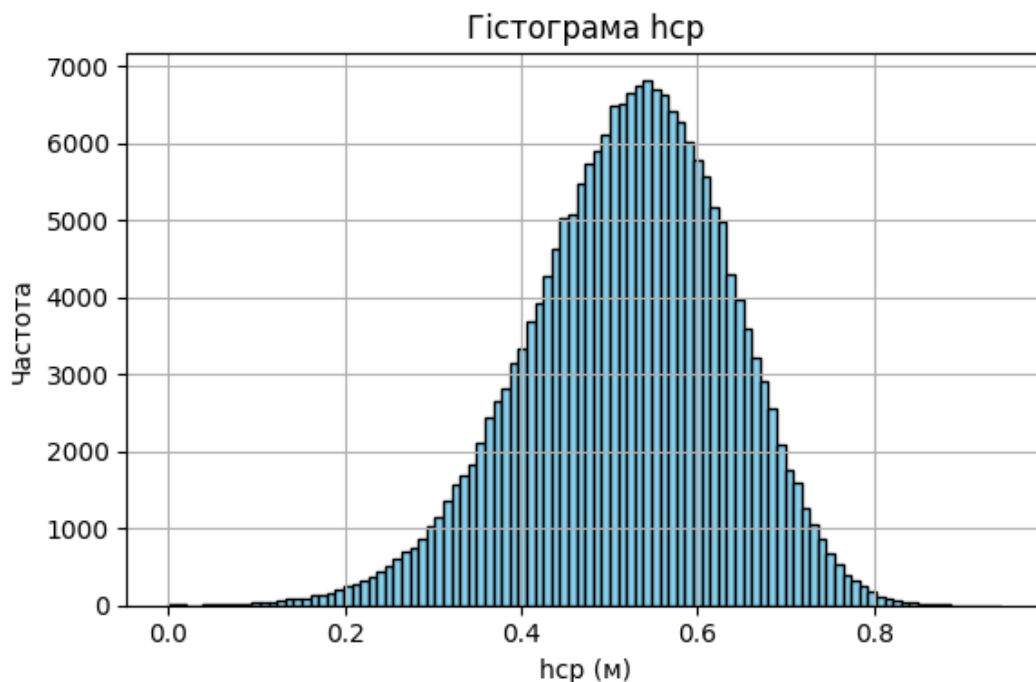


Рисунок 7.5

Гістограма dh показує частотний розподіл втрат напору. Стовпчики демонструють, які втрати є найбільш імовірними. Це важливо для оцінки ефективності роботи греблі (рис. 7.6).

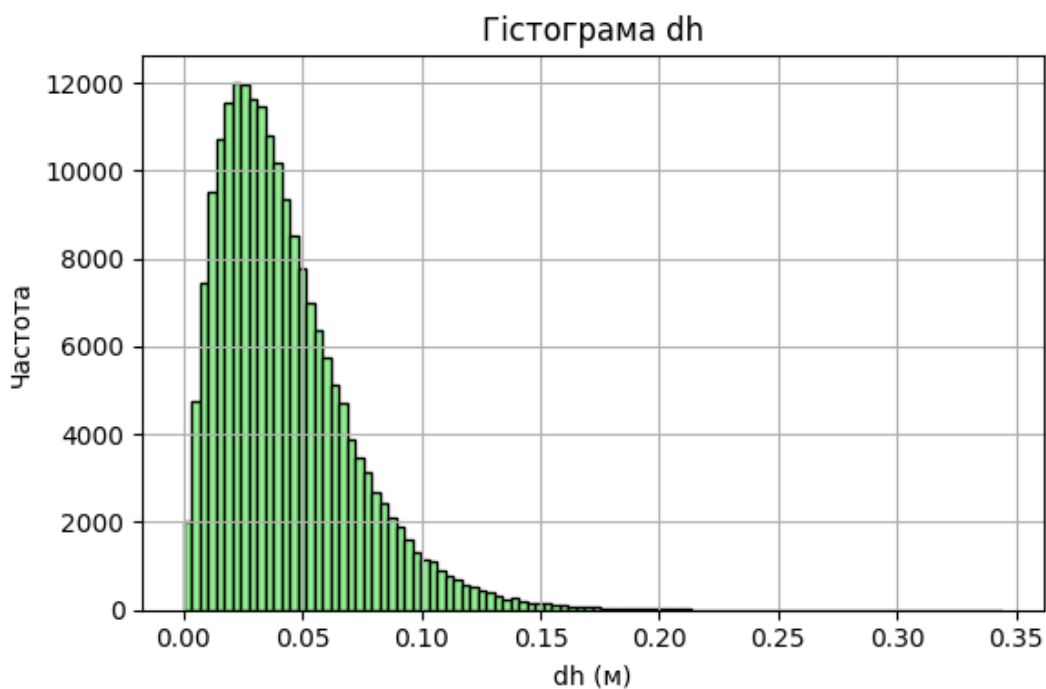


Рисунок 7.6

Гістограма T_{cp} показує частотний розподіл середнього періоду хвилі. Стовпчики демонструють, які періоди хвиль зустрічаються найчастіше. Це допомагає зрозуміти динаміку хвильових процесів перед греблею (рис. 7.7).

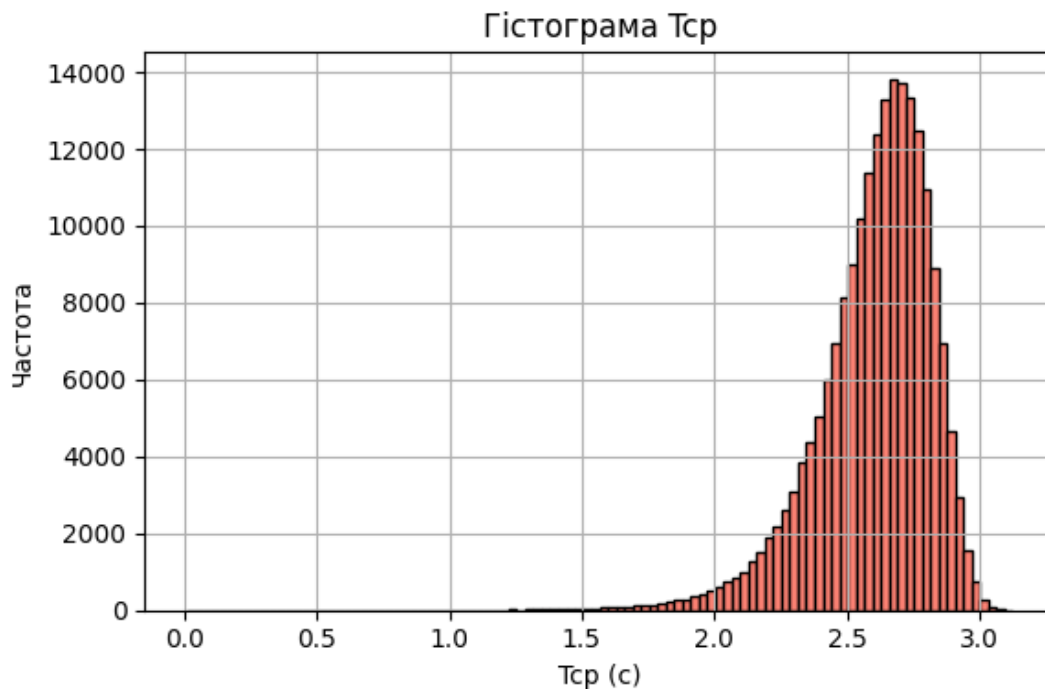


Рисунок 7.7

Контрольні запитання

1. Як нормальний розподіл використовується для моделювання випадкових величин, таких як швидкість вітру та рівень води?
2. Чому при статистичному аналізі важливо враховувати перцентилі (наприклад, 99-й) і що вони показують?
3. Як визначається імовірність переливу греблі в термінах частки реалізацій, що перевищують критичний рівень?

Завдання для самостійної роботи

1. У чому полягає роль граничних умов (мінімальна швидкість вітру, мінімальна глибина води) у статистичному моделюванні?
2. Як функція розподілу ймовірностей (CDF) допомагає оцінити ризик перевищення рівня гребеня?
3. Чому використання статистичних характеристик (середнє, стандартне відхилення) є ключовим для оцінки надійності гідротехнічних споруд?

СПИСОК ПОСИЛАНЬ

1. Дистанційний курс «Надійність систем у цивільному, промисловому та гідротехнічному будівництві».

URL:<https://dl.kname.edu.ua/course/view.php?id=4120>, (дата звернення: 05.02.2026).

2. Кічаєва О. В. Надійність основ та фундаментів : конспект лекцій для студентів очної та заочної форм навчання другого (магістерського) рівня вищої освіти за спеціальністю 192 – Будівництво та цивільна інженерія. Харків : ХНУМГ ім. О. М. Бекетова, 2021. 60 с. URL: https://eprints.kname.edu.ua/57640/1/2020%20%D0%9B%20%D0%BF%D0%B5%D1%87%20%D0%9D%D0%B0%D0%B4%D1%96%D0%B9%D0%BD%D1%96%D1%81%D1%82%D1%8C_%D0%9A%D0%B8%D1%87%D0%B0%D0%B5%D0%B2%D0%B0.pdf, (дата звернення: 05.02.2026).

3. Beacher G. B., Cristian J. T. Reliability and statistics in geotechnical Engineering. New York : John Wiley, 2003. 619 p.

4. Ross S. M. Introduction to probability and statistics for engineers and scientists. New York : Academic press, 2020. 674 p.

5. Smith D. J. Reliability, maintainability and risk: practical methods for engineers. Waltham : Butterworth-Heinemann, 2021. 456 p.

6. Montgomery D. C., Runger G. C. Applied statistics and probability for engineers. Hoboken : John Wiley & Sons, 2020. 710 p.

7. Keith T. Z. Multiple regression and beyond: An introduction to multiple regression and structural equation modelling. 3-rd ed. New York : Taylor & Francis, 2019. 639 p.

8. ДБН В.1.1–12:2014 Будівництво у сейсмічних районах України. [Чинний від 2014–10–01]. Київ : Мінрегіон України, ДП «Укрархбудінформ», 2014. 110 с. (Державні будівельні норми України).

9. ДБН В.1.2–2:2006 Навантаження і впливи. Норми проектування. [Чинний від 2007-01-01]. Київ : Видавництво «Сталь», 2006. 60 с. (Державні будівельні норми України).

10. ДБН В.1.2-14-2018 Загальні принципи забезпечення надійності та конструктивної безпеки будівель і споруд. [Чинний від 2019-01-01]. Київ : ДП «Укрархбудінформ», 2018. 30 с. (Державні будівельні норми України).

11. ДБН В.2.4-3:2010 Гідротехнічні споруди. Основні положення. [Чинний від 2011-01-01]. Київ : ДП «Укрархбудінформ», 2010. 37 с. (Державні будівельні норми України).

12. ДСТУ Б. В. 1.1-27: 2010 Будівельна кліматологія. [Чинний від 2011-11-01]. Київ : Державне підприємство «Укрархбудінформ», 2011. 124 с. (Національний стандарт України).
13. JCSS Probabilistic model code. Part 1. Basis of design. JCSS working materials. JCSS. 2000. 62 p.
14. JCSS Probabilistic model code. Part 2. Load models. JCSS working materials. JCSS. 2001. 73 p.
15. JCSS Probabilistic model code. Part 3. Material properties resistance models. Memorandum. JCSS. 2000. 41 p.
16. Safety of Structures. An independent technical expert review of partial factors for actions and load combinations in EN 1990 «Basis of Structural Design»: BRE Client Report № 210297 [Electronic resource]. Building Research Establishment. 2003. URL: https://www.concreteeurope.eu/images/ECP_Documents/SafetyOfStructures_compress.pdf, (Last accessed: 02.02.2026).

Електронне навчальне видання

Методичні рекомендації
до проведення практичних занять
та організації самостійної роботи
з навчальної дисципліни

**«НАДІЙНІСТЬ СИСТЕМ У ЦИВІЛЬНОМУ, ПРОМИСЛОВОМУ ТА
ГІДРОТЕХНІЧНОМУ БУДІВНИЦТВІ»**

*(для здобувачів третього (освітньо-наукового) рівня вищої освіти денної
форми навчання зі спеціальності 192 – Будівництво та цивільна інженерія*

Укладач **МОЗГОВИЙ** Андрій Олексійович

Відповідальний за випуск *В. А. Александрович*

Редактор *О. А. Норик*

Комп'ютерне верстання *А. О. Мозговий*

План 2026, поз. 12М

Підп. до друку 24.03.2026. Формат 60 × 84/16.

Ум. друк. арк. 3,8.

Видавець і виготовлювач:

Харківський національний університет
міського господарства імені О. М. Бекетова,

вул. Черноглазівська, 17, Харків, 61002.

Електронна адреса: office@kname.edu.ua

Свідоцтво суб'єкта видавничої справи:

ДК № 8386 від 14.07.2025.