**Міністерство освіти і науки України**

**Національний технічний університет України «КПІ» імені Ігоря Сікорського**

**Кафедра обчислювальної техніки ФІОТ**

**ЗВІТ**

**з лабораторної роботи №3**

**з навчальної дисципліни «Вступ до технології Data Science»**

**Тема:**

**ДОСЛІДЖЕННЯ РЕКУРЕНТНИХ АЛГОРИТМІВ ЗГЛАДЖУВАННЯ**

**Виконав:**

Студент X курсу кафедри ОТ ФІОТ,

Навчальної групи ІА-73

Петренко П.П.

**Перевірив:**

Професор кафедри ОТ ФІОТ

Писарчук О.О.

**Київ 2023**

**І. Мета:**

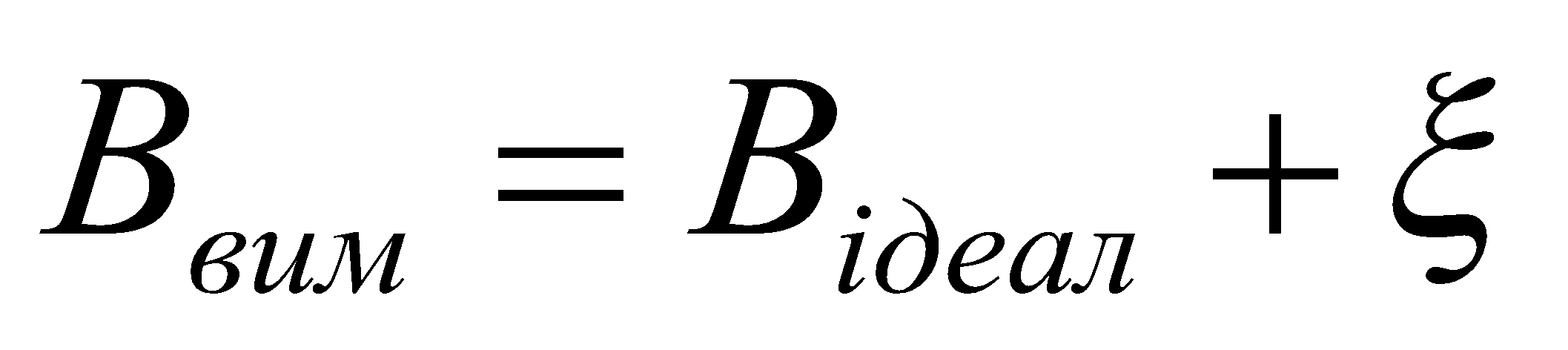
виявити дослідити та узагальнити особливості застосування методів первинної обробки експериментальних вибірок – виявлення аномальних вимірів та рекурентного згладжування з використанням спеціалізованих пакетів мови програмування Python.

**ІІ. Завдання:**

**І. Розробити універсальний скрипт мовою Python що реалізує наступні етапи моделювання та обробки експериментальних даних.**

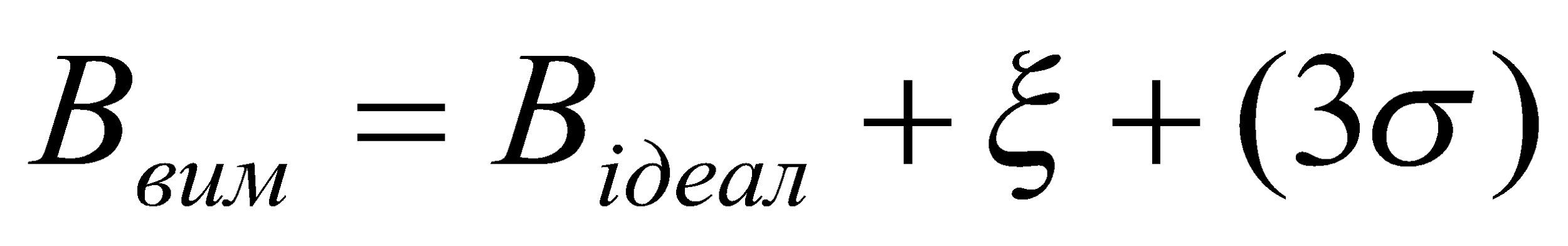
*1. Модель експериментальної вибірки з аномаліями відповідно до пунктів.*

1.1. Розробити модель дискретних значень виміряних параметрів експериментальної вибірки з характеристиками: трендова модель має квадратичний закон зміни; вибірка має 100 вимірів; випадкова похибка вимірів розподілена за нормальним законом з нульовим середнім та змінним значенням середньоквадратичної похибки вимірювання; модель виміру – адитивна:

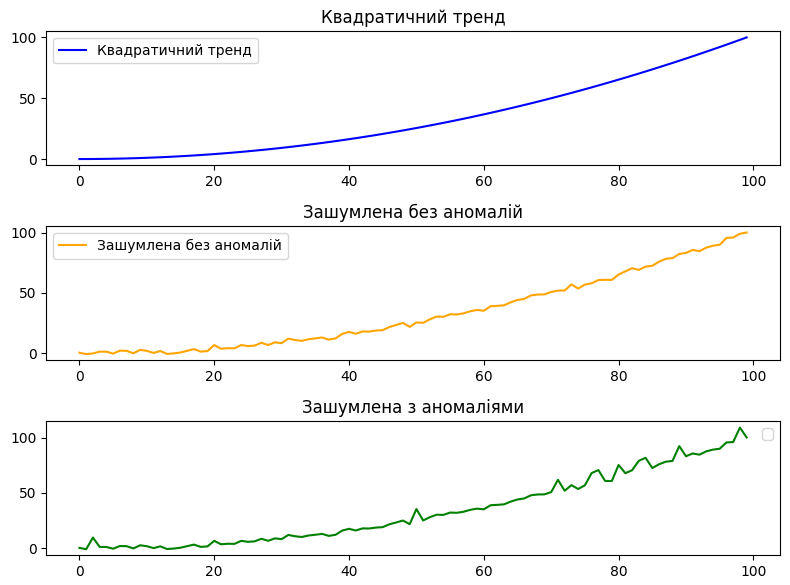
.

Для виконання даного завдання використати результати лабораторної роботи №1,2.

1.2. Модель генерації аномальних вимірів випадкової величини:

.

Аномальні виміри складають 10% від загальної кількості вимірів у експериментальної вибірки. Аномальні виміри рівномірно розташовані у межах дискретних значень експериментальної вибірки.

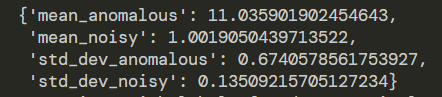


Риснок 1. Моделі: Квадратична, зашумлена, з аномаліями

*2. Виявлення аномальних вимірів та усунення їх впливу на результати обробки відповідно до підходів, заданих у таблиці Д1 додатку 1.*

***3. Здійснити згладжування експериментальної вибірки за відсутності аномальних вимірів відповідно до обраної форми рекурентного фільтру***

*4. З використанням методу Монте-Карло дослідити статистичні характеристики (математичне сподівання, середньоквадратичне відхилення, гістограма закону розподілу):* закону розподілу випадкової похибки вимірів; вхідної вибірки значень (зашумленої без аномальних вимірів); аномальної вибірки (зашумленої з аномальними вимірами); результатів рекурентного згладжування.

******

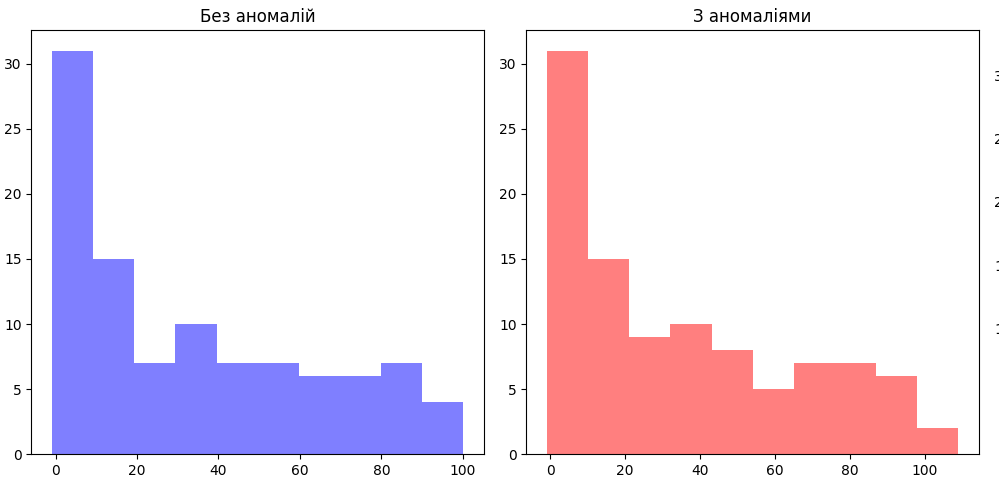
***Рисунок 2. Результати методу Монте-Карло для аналізу***

***Математичного сподівання та середньоквадратичного відхилення***

***Зашумленої та зашумленої з аномаліями моделі***

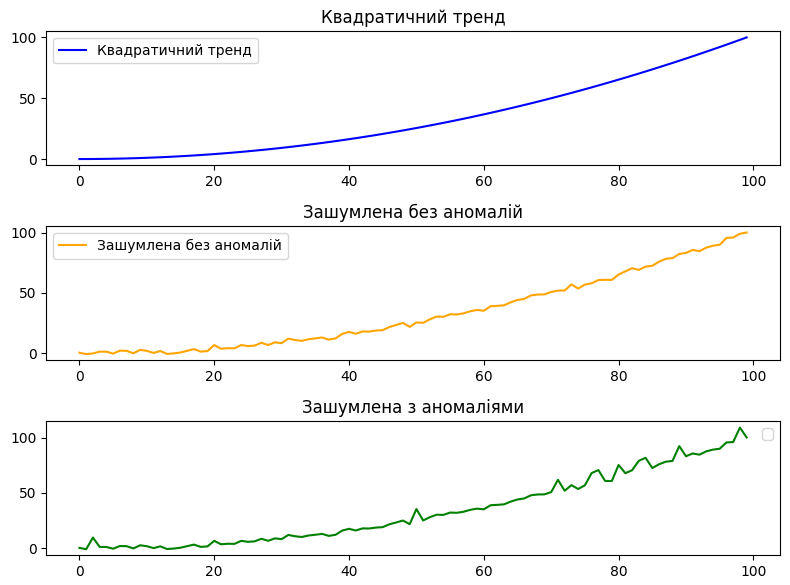
*5. Відобразити результати розрахунків:*

5.1. Статистичні характеристики (математичне сподівання, середньоквадратичне відхилення) закону розподілу випадкової похибки вимірів; вхідної вибірки значень (зашумленої без аномальних вимірів);а аномальної вибірки (зашумленої з аномальними вимірами); результатів рекурентного згладжування закону – у консолі та у формі таблички у звіті з лабораторної роботи.

****

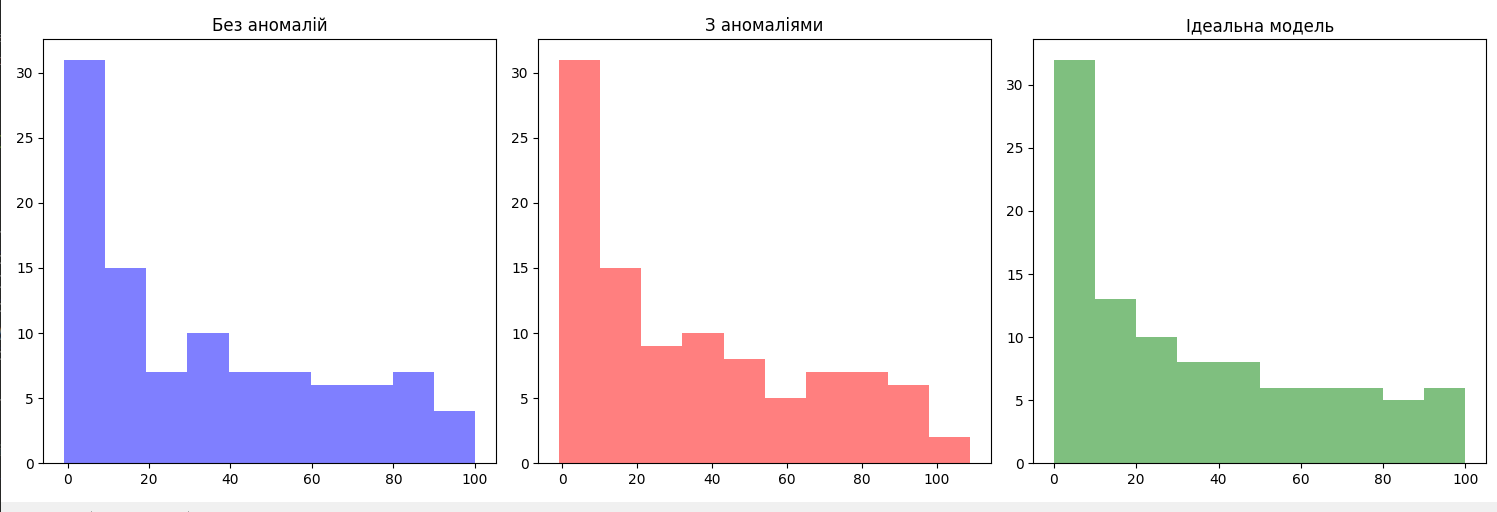
**Рисунок 3. Гістограма розподілу моделелей**

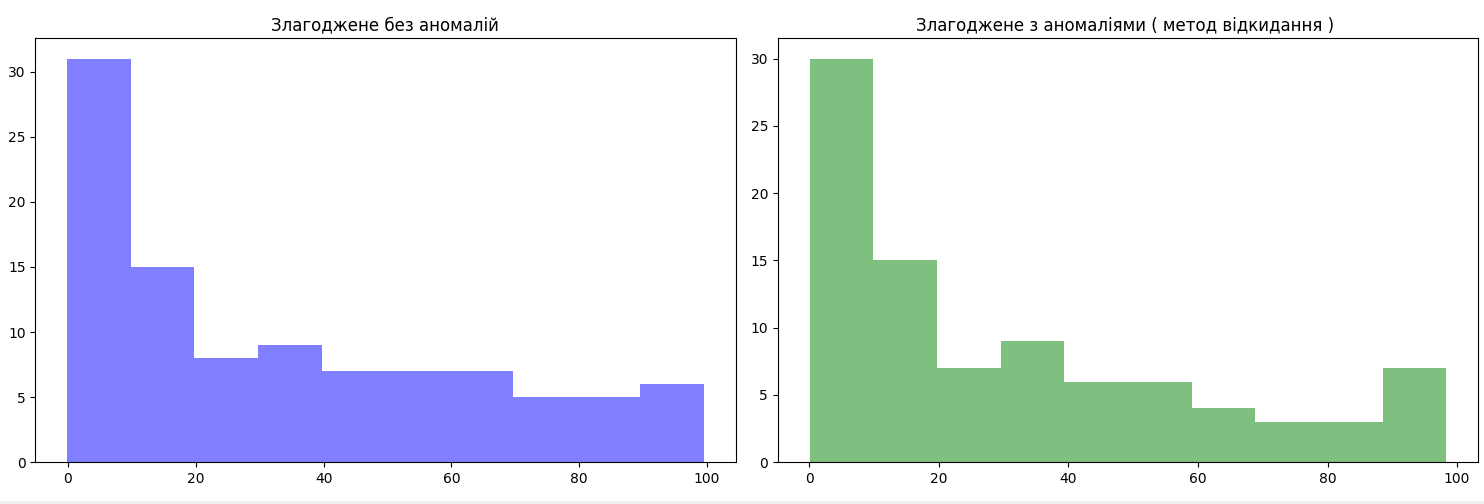
5.2. Графіки (в одному графічному вікні): квадратичного тренду; зашумленої без аномальних вимірів вибірки; зашумленої з аномальними вимірами вибірки; результатів рекурентного згладжування.



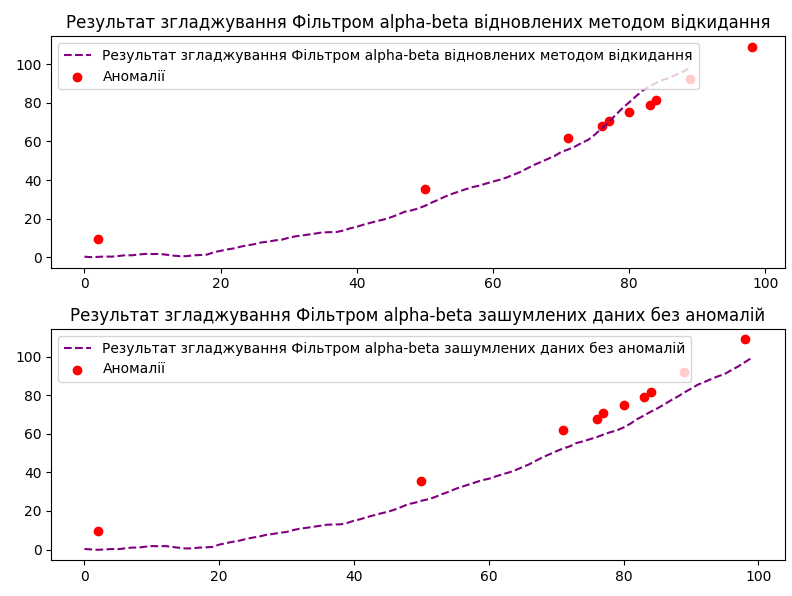
**Рисунок 4. Графіки зашумленої та зашумленої з аномаліями моделі.**

5.3. Гістограми (в одному графічному вікні) похибок: зашумленої без аномальних вимірів вибірки; зашумленої з аномальними вимірами вибірки; результатів рекурентного згладжування.

****

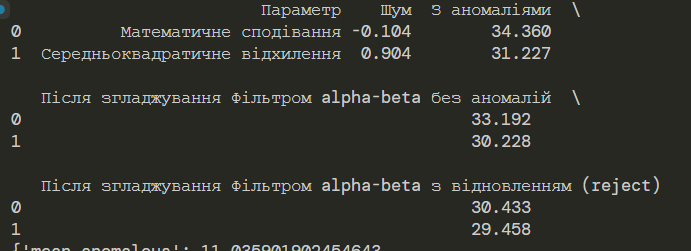
****

**Рисунок 5. Гістограми зашумленої, зашумленої з аномаліями, злагодженої з аномаліями та злагодженої без аномалій моделями**

****

**Рисунок 6. Графіки злагодження Фільтром alpha-betaзашумленої та зашумленої з аномаліями**

**Рисунок 7. Статистичні дані**

****

**ІІІ. Оформити звіт з лабораторної роботи.**

**ІІ. Провести дослідження зміни статистичних характеристик сформованих моделей при зміні стохастичних характеристик помилки експериментальних даних.**

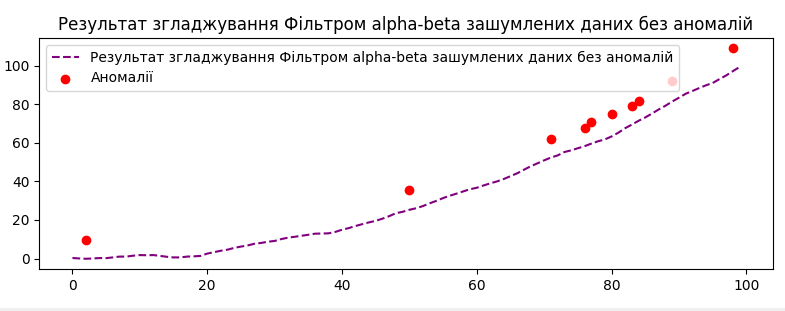


Рисунок 9. Демонстрація зглагоджування Фільтром alpha-beta

Таким чином, Фільтром alpha-beta демонструє свою ефективність у згладжуванні даних часових рядів і пом'якшенні впливу аномалій, що призводить до покращення оцінок основної динаміки системи. Аналіз методом Монте-Карло дає кількісне уявлення про ефективність фільтра за наявності аномалій.

**ІІІ. Довести адекватність сформованих моделей та працездатність розробленого скріпта.**

**import numpy as np**

**import matplotlib.pyplot as plt**

**import pandas as pd**

***def* alpha\_beta\_filter(*data*, *alpha*, *beta*):**

**estimation, velocity = *data*[0], 0**

**estimations = []**

**for measurement in *data*:**

**estimation = estimation + velocity**

**residual = measurement - estimation**

**estimation = estimation + *alpha* \* residual**

**velocity = velocity + *beta* \* residual / 1**

**estimations.append(estimation)**

**return estimations**

***def* run(*measurements*):**

**alpha, beta = 0.2, 0.1**

**filtered\_data = alpha\_beta\_filter(*measurements*, alpha, beta)**

**return np.array(filtered\_data)**

***def* detect\_anomalies\_least\_squares(*data*, *threshold\_factor*=2.0):**

**"""**

**Виявлення аномалій методом найменших квадратів.**

**Params:**

**- data: numpy array, часовий ряд вимірювань**

**- threshold\_factor: множник порогового значення**

**Returns:**

**- anomalies: список індексів аномальних вимірювань**

**"""**

**n = len(*data*)**

**X = np.column\_stack((np.arange(1, n + 1), np.ones(n)))**

**y = *data*.reshape(-1, 1)**

**beta, residuals, \_, \_ = np.linalg.lstsq(X, y, *rcond*=None)**

**residuals = y - np.dot(X, beta)**

**std\_residuals = np.std(residuals)**

**anomalies = np.where(np.abs(residuals) > *threshold\_factor* \* std\_residuals)[0]**

**return anomalies**

***def* reject\_anomalies(*data*, *anomalies*):**

**"""**

**Метод усунення впливу аномальних вимірів (відкидання).**

**Parameters:**

**- data: numpy array, вхідні виміри**

**- anomalies: numpy array, індекси аномальних вимірів**

**Returns:**

**- cleaned\_data: numpy array, виміри після усунення аномалій**

**"""**

**cleaned\_data = np.delete(*data*, *anomalies*)**

**return cleaned\_data**

***def* monte\_carlo\_analysis(*noisy\_data*, *anomalies*, *iterations*=1000):**

**""" anomalies - indicies of anomalies """**

**errors\_noisy = []**

**errors\_anomalous = []**

**for \_ in range(*iterations*):**

**noisy\_data\_iteration = *noisy\_data* + np.random.normal(0, 1, len(*noisy\_data*))**

**errors\_noisy.append(np.mean((noisy\_data\_iteration - *noisy\_data*) \*\* 2))**

**anomalous\_data = np.copy(noisy\_data\_iteration)**

**anomalous\_data[*anomalies*] += 10**

**errors\_anomalous.append(np.mean((anomalous\_data - *noisy\_data*) \*\* 2))**

**return {**

**'mean\_noisy': np.mean(errors\_noisy),**

**'std\_dev\_noisy': np.std(errors\_noisy),**

**'mean\_anomalous': np.mean(errors\_anomalous),**

**'std\_dev\_anomalous': np.std(errors\_anomalous),**

**}**

**np.random.seed(42)**

**# CONSTANTS**

**MESAURMENTS = 100**

**PERCENTAGE\_ANOMALIES = 0.1**

**# MODELS**

**""" Квадратичний тренд """**

**model\_ideal = np.linspace(0, 10, MESAURMENTS) \*\* 2**

**""" Шум нормального розпоілу """**

**noise = np.random.normal(*loc*=0.0, *scale*=1.0, *size*=MESAURMENTS)**

**""" Зашумлені виміри без аномалій """**

**measurements = model\_ideal + noise + 3 \* np.std(noise) \* (np.random.rand(MESAURMENTS) - 0.5)**

**""" Аномалії """**

**anomalies = np.random.choice(MESAURMENTS, int(MESAURMENTS \* PERCENTAGE\_ANOMALIES), *replace*=False)**

**""" Зашумлені виміри з аномаліями """**

**measurments\_with\_anomalies = measurements.copy()**

**measurments\_with\_anomalies[anomalies] += 10**

**""" Відновлені від аномалій виміри шляхом відкидання """**

**restored\_measurments\_rejected = reject\_anomalies(measurements, anomalies)**

**""" Згладжені виміри відновлених вимірів """**

**smoothed\_restored\_reject\_measurments = run(restored\_measurments\_rejected)**

**""" Злагоджені виміри вимірів без аномалій """**

**smoothed\_without\_anomalies = run(measurements)**

**statistics\_table = pd.DataFrame({**

**'Параметр': ['Математичне сподівання', 'Середньоквадратичне відхилення'],**

**'Шум': [np.mean(noise), np.std(noise)],**

**'З аномаліями': [np.mean(measurments\_with\_anomalies), np.std(measurments\_with\_anomalies)],**

**'Після згладжування Фільтром alpha-beta без аномалій': [np.mean(smoothed\_without\_anomalies), np.std(smoothed\_without\_anomalies)],**

**'Після згладжування Фільтром alpha-beta з відновленням (reject)': [np.mean(smoothed\_restored\_reject\_measurments), np.std(smoothed\_restored\_reject\_measurments)],**

**})**

**with pd.option\_context('display.max\_rows', None,**

**'display.max\_columns', None,**

**'display.precision', 3,**

**):**

**print(statistics\_table)**

**import pprint**

**pprint.pprint(**

**monte\_carlo\_analysis(measurements, anomalies)**

**)**

**""" Графіки даних """**

**""" Ідеальна та зашумлені моделі """**

**plt.figure(*figsize*=(8, 6))**

**plt.subplot(3, 1, 1)**

**plt.plot(model\_ideal, *label*='Квадратичний тренд', *color*='blue')**

**plt.legend()**

**plt.title('Квадратичний тренд')**

**plt.subplot(3, 1, 2)**

**plt.plot(measurements, *label*='Зашумлена без аномалій', *color*='orange')**

**plt.legend()**

**plt.title('Зашумлена без аномалій')**

**plt.subplot(3, 1, 3)**

**plt.legend()**

**plt.title('Зашумлена з аномаліями')**

**plt.plot(measurments\_with\_anomalies, *label*='Зашумлена з аномаліями', *color*='green')**

**plt.tight\_layout()**

**plt.show()**

**""" Злагоджені моделі """**

**plt.figure(*figsize*=(8, 6))**

**plt.subplot(2, 1, 1)**

**plt.plot(smoothed\_restored\_reject\_measurments, *label*='Результат згладжування Фільтром alpha-beta відновлених методом відкидання', *linestyle*='--', *color*='purple')**

**plt.legend()**

**plt.title('Результат згладжування Фільтром alpha-beta відновлених методом відкидання')**

**valid\_anomalies = anomalies[anomalies < len(measurments\_with\_anomalies)]**

**plt.scatter(valid\_anomalies, measurments\_with\_anomalies[valid\_anomalies], *color*='red', *label*='Аномалії')**

**plt.legend()**

**plt.subplot(2, 1, 2)**

**plt.plot(smoothed\_without\_anomalies, *label*='Результат згладжування Фільтром alpha-beta зашумлених даних без аномалій', *linestyle*='--', *color*='purple')**

**plt.legend()**

**plt.title('Результат згладжування Фільтром alpha-beta зашумлених даних без аномалій')**

**valid\_anomalies = anomalies[anomalies < len(measurments\_with\_anomalies)]**

**plt.scatter(valid\_anomalies, measurments\_with\_anomalies[valid\_anomalies], *color*='red', *label*='Аномалії')**

**plt.legend()**

**# Додаткові налаштування**

**plt.tight\_layout()**

**plt.show()**

**""" Гістограми розподілу """**

**fig, axs = plt.subplots(1, 3, *figsize*=(15, 5))**

**axs[0].hist(measurements, *alpha*=0.5, *color*='blue', *label*='Без аномалій')**

**axs[0].set\_title('Без аномалій')**

**axs[1].hist(measurments\_with\_anomalies, *alpha*=0.5, *color*='red', *label*='З аномаліями')**

**axs[1].set\_title('З аномаліями')**

**axs[2].hist(model\_ideal, *alpha*=0.5, *color*='green', *label*='Another Data')**

**axs[2].set\_title('Ідеальна модель')**

**plt.tight\_layout()**

**plt.show()**

**""" Гістограми розподілу злагодження Фільтром alpha-beta """**

**fig, axs = plt.subplots(1, 2, *figsize*=(15, 5))**

**axs[0].hist(smoothed\_without\_anomalies, *alpha*=0.5, *color*='blue', *label*='Злагоджене без аномалій')**

**axs[0].set\_title('Злагоджене без аномалій')**

**axs[1].hist(smoothed\_restored\_reject\_measurments, *alpha*=0.5, *color*='green', *label*='Злагоджене з аномаліями ( метод відкидання )')**

**axs[1].set\_title('Злагоджене з аномаліями ( метод відкидання ) ')**

**plt.tight\_layout()**

**plt.show()**

**Розроблений код повинен бути раціональним та відповідати вимогам до чистого коду.**

**III. Висновок:**

виявив дослідив та узагальнив особливості застосування методів первинної обробки експериментальних вибірок – виявлення аномальних вимірів та рекурентного згладжування з використанням спеціалізованих пакетів мови програмування Python.

Виконав: студент Петренко П.П.