

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ**  
**“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ”**  
**ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ**  
**КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ ЗАХИСТУ ІНФОРМАЦІЇ**

«До захисту допущено»

Завідувач кафедри

М. М. Савчук

“ \_\_\_\_\_ ” \_\_\_\_\_ 2015 р.  
(підпис) (ініціали, прізвище)

**Дипломна робота**

освітньо-кваліфікаційного рівня “бакалавр”

з напрямку підготовки 6.040301 «Прикладна математика»  
на тему «Розробка автоматизованого тестуючого комплексу,  
що враховує психологічні особливості студентів»

Виконав: студент 4 курсу групи ФІ-13

Кригін Валерій Михайлович

Керівник д-р ф.-м. наук, професор Дороговцев Андрій Анатолійович

Рецензент к. т. н., доцент Головенкін Володимир Павлович

\_\_\_\_\_  
(підпис)

\_\_\_\_\_  
(підпис)

\_\_\_\_\_  
(підпис)

Засвідчую, що у цій дипломній роботі  
немає запозичень з праць інших авторів  
без відповідних посилань.

Студент \_\_\_\_\_

## ЗМІСТ

Вступ . . . . .	3
1 Математичне підґрунтя . . . . .	5
1.1 Теоретичні відомості . . . . .	5
1.1.1 Метод головних компонент . . . . .	5
1.1.2 Побудова гістограми для критерію Пірсона . . . . .	9
1.1.3 Критерій узгодженості Пірсона $\chi^2$ . . . . .	10
1.1.4 Типи вищої нервової діяльності . . . . .	18
1.1.5 Теппінг-тест (Tapping rate) . . . . .	19
1.2 Моделювання результатів виконання тестових завдань . . . . .	21
1.2.1 Процес Пуассона . . . . .	21
1.2.2 Квадратична апроксимація методом найменших квадратів . . . . .	23
1.2.3 Аналіз квадратичної апроксимації . . . . .	27
Перелік посилань . . . . .	35
Додаток А . . . . .	36
Додаток Б . . . . .	38
Додаток В . . . . .	39

## ВСТУП

**Актуальність роботи** полягає в тому, що дистанційне навчання стає дедалі, а існуючі на даний момент системи тестування недостатньо гнучкі: вони аналізують лише відповіді на запитання, відносячи їх до вірних або невірних, а на цій базі роблять кінцевий висновок щодо знань студента. З одного боку, це підвищує їх об'єктивність, з іншого, навпаки: всі студенти різні і більшості з них потрібна допомога хоча б у вигляді порад. У викладачів може не вистачати часу на докладне знайомство з кожним студентом, але якщо буде програмно-технічний комплекс, що підкаже викладачам, до яких студентів який підхід краще мати, це безумовно підвищить якість навчання. Стрімкий розвиток комп'ютерної техніки й інформаційних технологій надає можливість визначати ритм складання тесту, а також індивідуальні особливості людини. Дані психологічних досліджень допоможуть правильно трактувати отримані значення, а добре вивчені та перевірені часом математичні методи надають великі можливості для систематизації та обробки результатів вимірювання.

*Об'єкт дослідження:* студенти, системи тестування.

*Предмет дослідження:* психологічні особливості студентів, моделі їх поведінки.

*Метою роботи є* збільшення об'єктивності тестування, а також покращення якості навчання за допомогою порад студентам і викладачам практичних занять.

Завдання наступні:

- 1) Вивчити математичні методи та розділи психології, що дозволять розв'язати поставлену задачу, пояснити та обґрунтувати отримані результати
- 2) Ознайомитися з правилами побудови тестових завдань для найбільш ефективної та об'єктивної процедури оцінки знань студентів
- 3) Розробити програмний комплекс тестування й обробки результатів

#### 4) Моделювання

# 1 МАТЕМАТИЧНЕ ПІДГРУНТЯ

## 1.1 Теоретичні відомості

### 1.1.1 Метод головних компонент

Метод головних компонент (Principal component analysis) — метод, що дозволяє зменшити розмірність досліджуваної вибірки з мінімальними втратами інформації. [1]

Маємо  $m$  об'єктів, з яких треба зняти по  $n$  певних властивостей. На вході в нас є виборки  $\vec{X}_k$ , кожна з яких відповідає сукупності властивостей  $k$ -го об'єкту

$$\vec{X}_k = \begin{bmatrix} x_k^1 \\ x_k^2 \\ \vdots \\ x_k^n \end{bmatrix}, \quad k = \overline{1, m}.$$

Згрупуємо всі вимірювання в одну матрицю  $X$

$$X = \begin{bmatrix} x_1^1 & x_2^1 & \dots & x_m^1 \\ x_1^2 & x_2^2 & \dots & x_m^2 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^n & x_2^n & \dots & x_m^n \end{bmatrix}.$$

Спочатку нам знадобиться знайти вибіркові середні значення для кожної

властивості

$$a_i = \frac{1}{m} \cdot \sum_{k=1}^m x_k^i, \quad i = \overline{1, n}.$$

Маємо вектор вибірових середніх значень

$$\vec{a} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}.$$

Центруємо отримані дані, що містяться в матриці  $X$ , віднявши від кожного стовбця вектор вибірових середніх  $\vec{a}$

$$\tilde{X} = \begin{bmatrix} \tilde{x}_1^1 & \tilde{x}_2^1 & \dots & \tilde{x}_m^1 \\ \tilde{x}_1^2 & \tilde{x}_2^2 & \dots & \tilde{x}_m^2 \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{x}_1^n & \tilde{x}_2^n & \dots & \tilde{x}_m^n \end{bmatrix} = \begin{bmatrix} x_1^1 - a_1 & x_2^1 - a_1 & \dots & x_m^1 - a_1 \\ x_1^2 - a_2 & x_2^2 - a_2 & \dots & x_m^2 - a_2 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^n - a_n & x_2^n - a_n & \dots & x_m^n - a_n \end{bmatrix}.$$

Обчислюємо вибірову коваріаційну матрицю властивостей. Вибіркову коваріацію  $i$  та  $j$  властивості рахуємо за формулою

$$\sigma_i^j = \frac{1}{m} \cdot \sum_{k=1}^m \tilde{x}_k^i \cdot \tilde{x}_k^j = \frac{1}{m} \cdot \sum_{k=1}^m \left[ (x_k^i - a_i) \cdot (x_k^j - a_j) \right], \quad i, j = \overline{1, n}.$$

Маємо вибірку коваріаційну матрицю

$$K = \begin{bmatrix} \sigma_1^1 & \sigma_2^1 & \dots & \sigma_n^1 \\ \sigma_1^2 & \sigma_2^2 & \dots & \sigma_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_1^n & \sigma_2^n & \dots & \sigma_n^n \end{bmatrix}.$$

Щоб отримувати лише потрібну інформацію, ми хочемо знайти таке ортогональне лінійне перетворення  $L$  вхідної матриці  $\tilde{X}$ , щоб отримати матрицю  $Y = L \cdot \tilde{X}$ , яка має діагональну вибірку коваріаційну матрицю  $K'$  з незростаючими зверху вниз значеннями. Діагональна вибірка коваріаційна матриця гарантує той факт, що отримані значення  $Y$  будуть некорельованими. Рангування значень діагональних елементів матриці  $K'$  за величиною дасть більш наочне уявлення про будову досліджуваних об'єктів, адже діагональні елементи — вибіркові дисперсії. Чим більше дисперсія, тим більше відповідна властивість змінюється від об'єкту до об'єкту, і тим більше корисної інформації вона нам надає.

Вибіркова коваріаційна матриця  $K'$  для  $Y = L \cdot \tilde{X}$  має вигляд

$$K' = L \cdot K \cdot L^* = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_n \end{bmatrix}.$$

З лінійної алгебри відомо, що матриця  $L$  складається з координат власних ве-

кторів матриці  $K$ , а елементи  $\lambda_k$  — її власні числа, які існують і є невід’ємними через невід’ємну означеність матриці  $K$ . Вважаємо, що числа  $\lambda_1, \dots, \lambda_n$  впорядковані від більшого до меншого для зручності подальших дій. Позначимо власний вектор матриці  $K$ , що відповідає власному числу  $\lambda_k$ , як  $\vec{l}_k$ . Тоді

$$\vec{l}_k = [l_k^1, l_k^2, \dots, l_k^n], \quad k = \overline{1, n}.$$

Матриця  $L$  має вигляд

$$L = \begin{bmatrix} l_1^1 & l_1^2 & \dots & l_1^n \\ l_2^1 & l_2^2 & \dots & l_2^n \\ \vdots & \vdots & \ddots & \vdots \\ l_n^1 & l_n^2 & \dots & l_n^n \end{bmatrix}.$$

Треба зменшити розмірність простору досліджуваних параметрів системи з  $n$  до  $p < n$ , але при цьому втратити якомога менше відомостей про досліджувані об’єкти. Введемо міру інформації, що залишається при зменшенні кількості компонент, що розглядаються

$$I = \frac{\lambda_1 + \dots + \lambda_p}{\lambda_1 + \dots + \lambda_n}.$$

Будемо вважати, що діємо продуктивно, тому починаємо обирати з перших компонент, адже саме вони є найбільш інформативними. Також бачимо, що інформативність змінюється в межах від 0 (нічого не дізнаємось) до 1 (зберегли усю інформацію).



Надалі буде розглядатися матриця головних компонент  $Y$

$$Y = \begin{bmatrix} y_1^1 & y_2^1 & \dots & y_m^1 \\ y_1^2 & y_2^2 & \dots & y_m^2 \\ \vdots & \vdots & \ddots & \vdots \\ y_1^p & y_2^p & \dots & y_m^p \end{bmatrix}.$$

### 1.1.2 Побудова гістограми для критерію Пірсона

Для подальшого аналізу потрібно здобути щільність розподілу головних компонент. Оскільки маємо справу з вибіркою і вибірковими характеристиками, потрібно побудувати гістограму, адже це і є вибіркова характеристика, що відповідає щільності. [2]

Побудуємо  $j$ -й стовбець гістограми для виборки з  $k$ -ї строки матриці  $Y$

$$h_j^k = \frac{1}{m} \cdot \sum_{i=1}^m \mathbb{1}(y_i^k \in I_j^k), \quad j = \overline{1, N}, \quad k = \overline{1, p},$$

де  $I^k$  — набір напівінтервалів, що розбиває відрізок  $\left[ \min_{i=\overline{1, m}} y_i^k; \max_{i=\overline{1, m}} y_i^k \right]$  на  $N$  рівних частин. Для вибору  $N$  можна скористатися досить відомою формулою Стюрджеса (Sturges' formula) [3]

$$N = \lfloor \log_2 m \rfloor + 1.$$

Маємо матрицю гістограм

$$H = \begin{bmatrix} h_1^1 & h_2^1 & \dots & h_N^1 \\ h_1^2 & h_2^2 & \dots & h_N^2 \\ \vdots & \vdots & \ddots & \vdots \\ h_1^p & h_2^p & \dots & h_N^p \end{bmatrix}$$

і напівінтервалів, що відповідають кожному стовбчику кожної гістограми

$$I = \begin{bmatrix} I_1^1 & I_2^1 & \dots & I_N^1 \\ I_1^2 & I_2^2 & \dots & I_N^2 \\ \vdots & \vdots & \ddots & \vdots \\ I_1^p & I_2^p & \dots & I_N^p \end{bmatrix}.$$

### 1.1.3 Критерій узгодженості Пірсона $\chi^2$

Гістограма може використовуватися не тільки для графічної інтерпретації отриманих даних, але й для віднесення вибірки до якогось відомого розподілу. Відповідь на питання “Чи дійсно вибірка  $y_1^k, \dots, y_m^k$  має розподіл  $F^k$ ?” може надати критерій узгодженості Пірсона.

Розглянемо вектор

$$\eta^k = \left[ \frac{\nu_1^k - m \cdot \rho_1^k}{\sqrt{m \cdot \rho_1^k}}, \dots, \frac{\nu_N^k - m \cdot \rho_N^k}{\sqrt{m \cdot \rho_N^k}} \right].$$

Знайдемо його характеристичну функцію

$$\varphi_{\eta^k}(\lambda) = M e^{i \cdot (\lambda, \eta^k)}, \quad \lambda \in \mathbb{R}^N.$$

Для зручності перепозначимо індикатор

$$\mathfrak{I}_{i,j}^k = \mathbb{1}\{y_i^k \in I_j^k\}.$$

Подивимось, чому дорівнює скалярний добуток в експоненті

$$\begin{aligned} (\lambda, \eta^k) &= \sum_{j=1}^N \lambda_j \cdot \frac{\nu_j^k - m \cdot \rho_j^k}{\sqrt{m \cdot \rho_j^k}} = \sum_{j=1}^N \frac{\lambda_j}{\sqrt{m \cdot \rho_j^k}} \cdot \sum_{i=1}^m (\mathfrak{I}_{i,j}^k - \rho_j^k) = \\ &= \sum_{j=1}^N \sum_{i=1}^m \frac{\lambda_j}{\sqrt{m \cdot \rho_j^k}} \cdot (\mathfrak{I}_{i,j}^k - \rho_j^k) = \sum_{i=1}^m \sum_{j=1}^N \lambda_j \cdot \frac{\mathfrak{I}_{i,j}^k - \rho_j^k}{\sqrt{m \cdot \rho_j^k}}. \end{aligned}$$

Бачимо суму  $m$  незалежних однаково розподілених випадкових величин. Введемо позначення

$$\mathfrak{I}_j^k = \mathbb{1}\{y_1^k \in I_j^k\},$$

а також позначимо новий випадковий вектор

$$\zeta^k = \left[ \frac{\mathfrak{I}_1^k - \rho_1^k}{\sqrt{m \cdot \rho_1^k}}, \dots, \frac{\mathfrak{I}_N^k - \rho_N^k}{\sqrt{m \cdot \rho_N^k}} \right].$$

Скалярний добуток прийняв вигляд

$$(\lambda, \eta^k) = \sum_{i=1}^m \sum_{j=1}^N \lambda_j \cdot \zeta_j^k = \sum_{i=1}^m (\lambda, \zeta^k) = m \cdot (\lambda, \zeta^k).$$

За рахунок незалежності випадкових величин  $\zeta_j^k$  маємо

$$\varphi_{\eta^k}(\lambda) = M e^{i \cdot (\lambda, \eta^k)} = M e^{m \cdot i \cdot (\lambda, \zeta^k)} = \left( M e^{i \cdot (\lambda, \zeta^k)} \right)^m. \quad (1.1)$$

Розглянемо характеристичну функцію випадкового вектора  $\zeta^k$

$$\varphi_{\zeta^k}(\lambda) = M \left[ \exp \left\{ i \cdot \sum_{j=1}^N \lambda_j \cdot \zeta_j^k \right\} \right]. \quad (1.2)$$

Легко побачити, що

$$\begin{aligned} (\lambda, \zeta^k) &= \sum_{j=1}^N \lambda_j \cdot \zeta_j^k = \sum_{j=1}^N \lambda_j \cdot \frac{\mathfrak{I}_j^k - \rho_j^k}{\sqrt{m \cdot \rho_j^k}} = \sum_{j=1}^N \left( \frac{\lambda_j}{\sqrt{m \cdot \rho_j^k}} \cdot \mathfrak{I}_j^k - \frac{\sqrt{\rho_j^k} \cdot \lambda_j}{\sqrt{m}} \right) = \\ &= \sum_{j=1}^N \mathfrak{I}_j^k \cdot \left( \frac{\lambda_j}{\sqrt{m \cdot \rho_j^k}} - \sum_{l=1}^N \frac{\sqrt{\rho_l^k} \cdot \lambda_l}{\sqrt{m}} \right). \end{aligned}$$

Тобто характеристична функція (1.2) приймає вигляд

$$\varphi_{\zeta^k}(\lambda) = M \left[ \sum_{j=1}^N \mathfrak{I}_j^k \cdot \exp \left\{ \frac{i}{\sqrt{m}} \left( \frac{\lambda_j}{\sqrt{\rho_j^k}} - \sum_{l=1}^N \sqrt{\rho_l^k} \cdot \lambda_l \right) \right\} \right].$$

Перепозначимо вираз в круглих дужках

$$\mathfrak{z}^k = \frac{\lambda_j}{\sqrt{\rho_j^k}} - \sum_{l=1}^N \sqrt{\rho_l^k} \cdot \lambda_l.$$

Математичне очікування індикатора — ймовірність події, яку він перевіряє. Отже

$$\varphi_{\zeta^k}(\lambda) = \sum_{j=1}^N \rho_j^k \cdot \exp \left\{ \frac{i \cdot \mathfrak{z}^k}{\sqrt{m}} \right\}.$$

Якщо розмір вибірки  $m$  буде зростати, то характеристична функція  $\eta^k$  (1.1) буде поводитись наступним чином

$$\begin{aligned} \lim_{m \rightarrow \infty} \varphi_{\eta^k}(\lambda) &= \lim_{m \rightarrow \infty} \left( 1 + \sum_{k=1}^N p_k \cdot \left[ \exp \left\{ \frac{i \cdot \mathfrak{z}^k}{\sqrt{m}} \right\} - 1 \right] \cdot \frac{m}{m} \right)^m = \\ &= \lim_{m \rightarrow \infty} \exp \left\{ m \cdot \sum_{k=1}^N p_k \cdot \left[ \exp \left\{ \frac{i \cdot \mathfrak{z}^k}{\sqrt{m}} \right\} - 1 \right] \right\}. \end{aligned}$$

Для  $\exp \left\{ \frac{i \cdot \mathfrak{z}^k}{\sqrt{m}} \right\}$  використаємо співвідношення

$$e^\alpha - 1 \approx \alpha + \frac{\alpha^2}{2}, \quad \alpha \ll 1.$$

Маємо

$$\begin{aligned} \sum_{j=1}^N \rho_j^k \cdot \frac{i \cdot \mathfrak{z}^k}{\sqrt{m}} &= \sum_{j=1}^N \rho_j^k \cdot \frac{i}{\sqrt{m}} \cdot \left( \frac{\lambda_j}{\sqrt{\rho_j^k}} - \sum_{l=1}^N \sqrt{\rho_l^k} \cdot \lambda_l \right) = \\ &= \frac{i}{\sqrt{m}} \cdot \left( \sum_{j=1}^N \sqrt{\rho_j^k} \cdot \lambda_j - \sum_{l=1}^N \sqrt{\rho_l^k} \cdot \lambda_l \right) = 0, \end{aligned}$$

$$\begin{aligned}
\sum_{j=1}^N \rho_j^k \cdot \left( \frac{i \cdot \mathfrak{z}^k}{\sqrt{m}} \right)^2 &= - \sum_{j=1}^N \frac{\rho_j^k}{m} \cdot \left( \frac{\lambda_j}{\sqrt{\rho_j^k}} - \sum_{l=1}^N \sqrt{\rho_l^k} \cdot \lambda_l \right)^2 = \\
&= - \frac{1}{m} \cdot \left[ \sum_{j=1}^N \lambda_j^2 - \left( \sum_{l=1}^N \sqrt{\rho_l^k} \cdot \lambda_l \right)^2 \right],
\end{aligned}$$

тому

$$\begin{aligned}
\lim_{m \rightarrow \infty} \varphi_{\eta^k}(\lambda) &= \lim_{m \rightarrow \infty} \exp \left\{ - \frac{m}{m \cdot 2} \cdot \left[ \sum_{j=1}^N \lambda_j^2 - \left( \sum_{l=1}^N \sqrt{\rho_l^k} \cdot \lambda_l \right)^2 \right] \right\} = \\
&= \exp \left\{ - \frac{1}{2} \cdot \left[ \sum_{j=1}^N \lambda_j^2 - \left( \sum_{l=1}^N \sqrt{\rho_l^k} \cdot \lambda_l \right)^2 \right] \right\} = e^{-\frac{1}{2} \cdot (A^k \lambda, \lambda)}.
\end{aligned}$$

Матриця  $A^k$  побудована наступним чином

$$A^k = \left\| \delta_{ij} - \sqrt{\rho_i^k} \cdot \sqrt{\rho_j^k} \right\|_{i=1}^n.$$

Симетричність матриці очевидна, тому треба довести її невід'ємну визначеність, щоб стверджувати, що вона є коваріаційною. Для цього візьмемо вектор

$$e^k = \left[ \sqrt{\rho_1^k}, \dots, \sqrt{\rho_N^k} \right], \quad \|e^k\| = 1.$$

Тоді бачимо, що

$$(A^k \lambda, \lambda) = \|\lambda\|^2 - (\lambda, e^k)^2. \quad (1.3)$$

З нерівності Коші маємо

$$\|(\lambda, e^k)\| \leq \|\lambda\| \cdot \|e^k\| = \|\lambda\|.$$

Тобто матриця є дійсно невід’ємно визначеною і вектор  $\eta^k$  розподілений за нормальним законом з нульовим середнім і коваріаційною матрицею  $A^k$ .

Для подальших розрахунків розглянемо стандартний гаусівський вектор як суму випадкових нормально розподілених випадкових величин в стандартному базисі  $\mathbb{R}^N$ , який позначимо  $[e_1, \dots, e_N]$

$$\xi = \sum_{j=1}^N \xi_j \cdot e_j \sim N(\vec{0}, I).$$

Згадуємо, що ортогональні перетворення  $U$  зберігають відстані, а також справедливо наступне

$$U\xi \sim N(0, UIU^{-1}) \sim N(\vec{0}, I).$$

Також ортонормований базис залишається ортонормованим базисом після ортогонального перетворення  $U$ . Оберемо такий оператор  $U$ , щоб набір  $[e_1, \dots, e_N]$  під його дією перетворився на  $[f_1, \dots, f_N]$ , де

$$f_1 = e^k = \left[ \sqrt{\rho_1^k}, \dots, \sqrt{\rho_N^k} \right].$$

Тоді маємо вектор

$$U\xi = \hat{\xi} = \sum_{j=1}^N \hat{\xi}_j \cdot f_j \sim N(\vec{0}, I).$$

Подивимось, який розподіл має наступний вектор

$$\Upsilon = \sum_{j=2}^N \hat{\xi}_j \cdot f_j = \hat{\xi} - \hat{\xi}_1 \cdot e^k.$$

Для цього розглянемо квадратичну форму

$$M(\Upsilon, \lambda)^2 = \sum_{j=2}^N (\lambda, f_j)^2 = \sum_{j=1}^N (\lambda, f_j)^2 - (\lambda, f_1)^2 = \|\lambda\|^2 - (\lambda, e^k)^2 = (A^k \lambda, \lambda).$$

З рівності (1.3) бачимо, що випадкові вектори  $\eta^k$  та  $\Upsilon$  мають однаковий розподіл. Отже, розподіли їх норм теж співпадають. Оскільки сума  $N - 1$  квадратів незалежних стандартних гаусових випадкових величин має розподіл Пірсона з  $N - 1$  ступенями вільності

$$\|\Upsilon\|^2 = \sum_{j=2}^N \xi_j^2 \sim \chi_{N-1}^2.$$

Маємо

$$\|\eta^k\|^2 = \sum_{j=1}^N \frac{(\nu_j^k - m \cdot \rho_j^k)^2}{m \cdot \rho_j^k} = m \cdot \sum_{j=1}^N \frac{(h_j^k - \rho_j^k)^2}{\rho_j^k} \sim \chi_{N-1}^2.$$

Останнє співвідношення дає змогу перевіряти належність виборки  $y_1^k, \dots, y_m^k$  до розподілу  $F^k$ . Перевірка виглядає наступним чином.

Розглянемо випадкову величину

$$R^k = m \cdot \sum_{j=1}^N \frac{(h_j^k - \rho_j^k)^2}{\rho_j^k}. \quad (1.4)$$

Обираємо рівень значущості  $\alpha$  для функції розподілу  $\chi_{N-1}^2$  і шукаємо відповідне до кількості ступенів вільності  $r_\alpha$ . Рівень значущості — ймовірність помилки першого роду, тобто ймовірність того, що буде відкинута вірна гіпотезу

$$\mathbb{P}(\chi_{N-1}^2 \geq r_\alpha) = \alpha.$$



Якщо  $R^k \leq r_\alpha$ , то гіпотеза про те, що вибірка  $Y^k$  дійсно має розподіл  $F^k$ , приймається.

Розглянемо той випадок, коли ймовірність  $\rho_i^k$  відгадана невірно. Повернемося до формули (1.4)

$$R^k = \sum_{j=1}^N \frac{(\nu_j^k - m \cdot \rho_j^k)^2}{m \cdot \rho_j^k}.$$

Всі члени суми є невід'ємними. Якщо хоча б один елемент буде завеликим, то великою буде вся сума. Маємо випадкову величину  $\eta$

$$\eta = \nu_i^k - m \cdot \rho_i^k = \sum_{j=1}^m (\xi_j - \rho_i^k), \quad \mathbb{1}(y_j^k \in I_i^k) = \xi_j.$$

Якщо  $\rho_i^k$  вгадано невірно, то воно не дорівнює математичному очікуванню індикатора. Додамо та віднімемо справжнє математичне очікування

$$\eta = \sum_{j=1}^m (\xi_j - M \xi_1 + M \xi_1 - \rho_i^k) = \sum_{j=1}^m (\xi_j - M \xi_1) + \sum_{j=1}^m (M \xi_1 - \rho_i^k).$$

Останній доданок є просто різницею, яку помножено на  $m$

$$\eta = \sum_{j=1}^m (\xi_j - M \xi_1) + m \cdot (M \xi_1 - \rho_i^k).$$

Поділимо на  $\sqrt{m}$ , щоб скористатися центральною граничною теоремою

$$\frac{\eta}{\sqrt{m}} = \frac{1}{\sqrt{m}} \cdot \sum_{j=1}^m (\xi_j - M \xi_1) + \frac{1}{\sqrt{m}} \cdot m \cdot (M \xi_1 - \rho_i^k).$$

Перший доданок асимптотично має розподіл  $N(0, \sigma^2)$ , де  $\sigma^2$  — дисперсія випад-

кової величини  $\xi_1$  для достатньо великих  $m$ . Отже вся сума зростає пропорційно до  $\sqrt{m}$

$$\frac{\eta}{\sqrt{m}} = \frac{1}{\sqrt{m}} \cdot \sum_{j=1}^m (\xi_j - M \xi_1) + \sqrt{m} \cdot (M \xi_1 - \rho_i^k) \sim \sqrt{m} \cdot (M \xi_1 - \rho_i^k).$$

Тобто зараз  $R^k$  буде зростати пропорційно до величини  $m$ , і буде великим у порівнянні з  $r_\alpha$ , що призведе до відхилення невірної гіпотези.

#### 1.1.4 Типи вищої нервової діяльності

Для визначення того, які показники вимірювати і яким чином, скористуємось відомою класифікацією типів вищої нервової діяльності.

Згідно з Павловим[4] типи вищої нервової діяльності характеризуються трьома показниками: сила нервової системи (сильна або слабка), врівноваженість (врівноважена або неврівноважена) та рухливість (рухлива або інертна). Павлов розглядає 4 комбінації цих показників з 8 можливих:

- 1) Слабка
- 2) Сильна та неврівноважена
- 3) Сильна, врівноважена та інертна
- 4) Сильна, врівноважена та рухлива

Далі ці класи (комбінації) будуть називатися відповідно слабкий, неврівноважений, інертний та рухливий.

### 1.1.5 Теппінг-тест (Tapping rate)

Існують відомі залежності між типом вищої нервової діяльності та зміною максимального темпу рухів кистю руки з часом. Протягом 30 секунд людина намагається притримуватися максимально можливого для себе темпу. Показники темпу фіксуються через кожні 5 секунд, а далі по 6 отриманим точкам будується крива темпа руху. [5]

Для тесту можна використовувати ручку (олівець) і папір, або телеграф. Сучасні технології дозволяють проводити тест за допомогою клавіатури комп'ютера або екрану планшета.

З олівцем і папіром тест поводиться наступним чином:

- 1) На папері креслиться 6 квадратів
- 2) Людина починає ставити якомога більше точок в першому квадраті впродовж перших 5 секунд
- 3) Коли проходить 5 секунд, потрібно перейти до наступного квадрату і ставити точки там
- 4) Процедура повторюється до тих пір, доки не пройде 30 секунд — в кінці буде заповнено всі 6 квадратів

Далі підраховується кількість точок в кожному квадраті та малюється ламана, де горизонтальна вісь відповідає номеру часового проміжку (номеру квадрата), а вертикальна відповідає кількості точок в квадраті.

Трактуються отримані дані наступним чином:

- 1) Спадна ламана відповідає слабкому типу (рис. 1.1д). Вона спадає після перших 5 секунд тесту і не повертається до початкового рівня
- 2) Ламана, що спочатку зростає, а після 10-15 секунд спадає нижче початкового рівня (проміжна між рівною та опуклою) відповідає неврівноваженому типу (рис. 1.1б).

- 3) Опукла вниз ламана відповідає інертному типу (рис. 1.1г). Вона спочатку спадає, а на 25-30 секундах може зрости до початкового темпу
- 4) Опукла вгору ламана відповідає рухливому типу (рис. 1.1а). Це така лама-на, що зростає в перші 10-15 секунд тесту, а після 25-30 секунд повертає-ться або падає нижче початкового рівня
- 5) Також темп може залишатися приблизно на одному рівні протягом усього тесту, що є оптимальним для складання іспитів (рис. 1.1в).

Достовірним відхиленням достатньо вважати різницю в два і більше натиснення між двома сусідніми п'ятисекундними проміжками. [5]

Оскільки цей тест заснований на вимірюванні витривалості нервової систе-ми людини за умови максимального навантаження та перевіряє темп реагування (натиснення) на подразнювачі (внутрішній подразнювач — команда собі “треба тиснути”), було вирішено використовувати відомі вигляди кривих (рис. 1.1) при моделюванні результатів виконання завдань однакової складності.

Потрібно зауважити, що швидкість розв'язування задач може змінюватися з досвідом. Тобто, якщо студент зі слабкою нервовою системою буде тренувати-ся виконувати завдання, то його показники з часом перейдуть на якісно новий рівень. Щодо студентів з сильною нервовою системою: швидкість не завжди означає якість виконання завдань.

Завдання системи — класифікувати студентів в автоматичному режимі, з метою подальшого надання порад викладачам практичних занять щодо підвище-ння продуктивності роботи кожного студента. Це полегшує знаходження інди-відуального підходу. Наприклад, тим, хто надто швидко втомлюється, потрібно розв'язувати якомога більше базових завдань, що не є складними, але розв'язок яких повинен бути на рівні рефлексів. Студентам, які поспішають, буде корисно ретельно коментувати у письмовій формі хід своїх думок, щоб вгамуватися та підвищити свою уважність.

Для тесту було обрано саме 30-секундний проміжок часу, адже спочатку

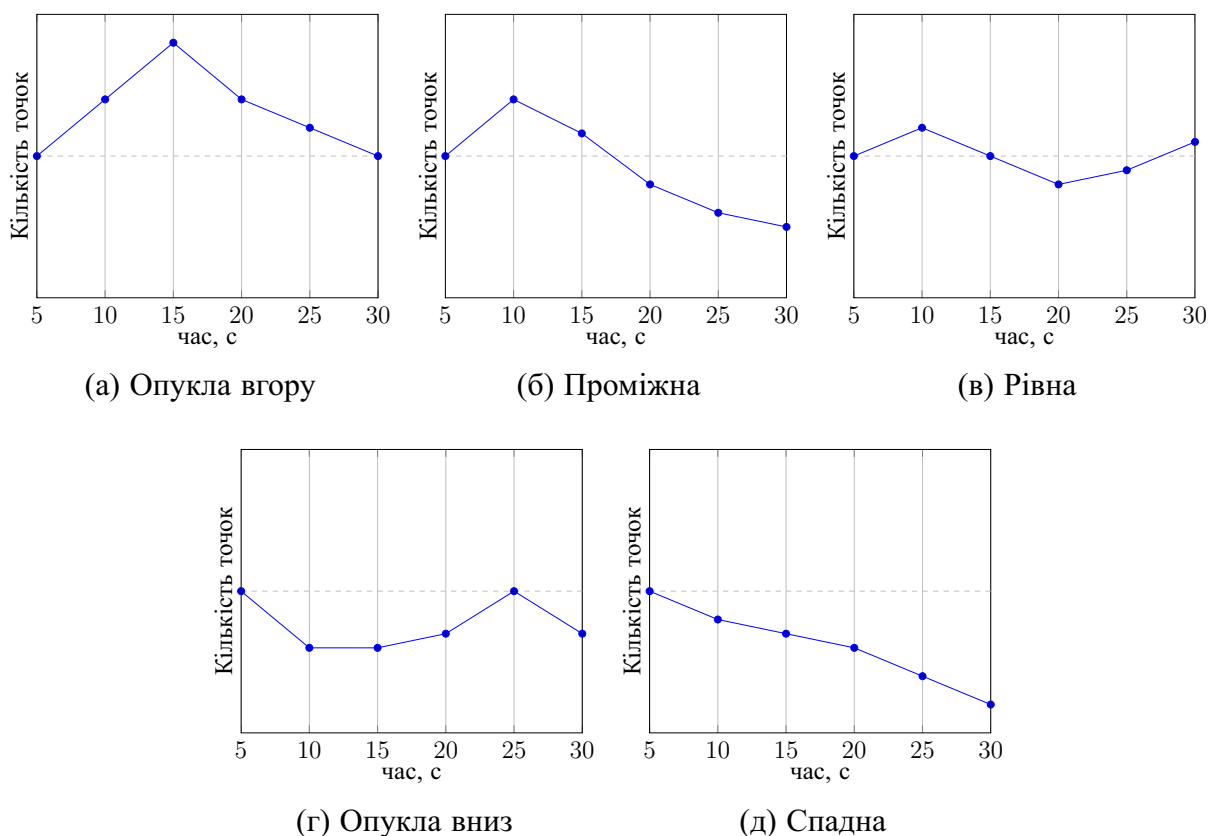


Рисунок 1.1 — Загальний вигляд залежностей кількості поставлених точок від часу. Пунктирна лінія — кількість точок в перші 5 секунд

виміри виконувалися протягом однієї хвилини і було виявлено, що найважливіша інформація отримується протягом перших 20-25 секунд, а далі лише марно втрачається час та сили тестованого. [5]

## 1.2 Моделювання результатів виконання тестових завдань

### 1.2.1 Процес Пуассона

Щоб моделювати темп виконання студентами контрольних задач, використаємо відомі дані з психології щодо ритму виконання теппінг-тесту для людей з різними типами вищої нервової діяльності. Оскільки постукування є потоком однорідних випадкових подій, змоделюємо їх як реалізації процесу Пуассона.

Нехай  $\{N(t) | t \geq 0\}$  — процес Пуассона з інтенсивністю  $\lambda(t)$ . Маємо 7 контрольних точок на часовій вісі  $t_0 = 0, t_1 = 5, \dots, t_6 = 30$ . Введемо відрізок часу  $T$ , за який проводиться вимірювання, та проміжки  $T_i$ , на які його розбито в експерименті

$$T = (0, 30], \quad T_i = (t_{i-1}, t_i], \quad i = \overline{1, 6}.$$

Нехай  $\nu_i$  визначається наступним чином

$$\nu_i = N(t_i) - N(t_{i-1}), \quad i = \overline{1, 6},$$

тоді  $\nu_i$  — випадкова величина, що має розподіл Пуассона з параметром  $\lambda_i$ , де [6]

$$\lambda_i = m(T_i) = \int_{t_{i-1}}^{t_i} \lambda(\tau) d\tau, \quad i = \overline{1, 6}.$$

Функція  $m$  — міра інтенсивності. [7]

Проміжки  $T_i$  утворюють розбиття множини  $T$ , а отже не перетинаються. Це означає, що випадкові величини  $\nu_i$  незалежні між собою, і є можливість згенерувати процес натискання за допомогою 6 незалежних випадкових величин, що мають розподіл Пуассона.

Інтенсивність (параметр  $\lambda$ ) є математичним очікуванням випадкової величини, що має розподіл Пуассона. Маючи результати реальних експериментів, можна обчислити середні значення кількості постукувань для кожного часового проміжку  $T_i$ , а отримані результати використовувати в якості параметрів  $\lambda_i$ .

Візьмемо значення відомого експерименту з підручника [5], які зображено на рис. 1.2. Оскільки результат один, то його значення будуть середніми з вибір-

ки, що складається з одного вектора — їх і використаємо в якості параметрів  $\lambda_i$ . На рис. 1.3а зображено різні реалізації послідовності випадкових величин, що розподілені за законом Пуассона з параметрами

$$[\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_6] = [43, 40, 38, 37, 38, 35]. \quad (1.5)$$

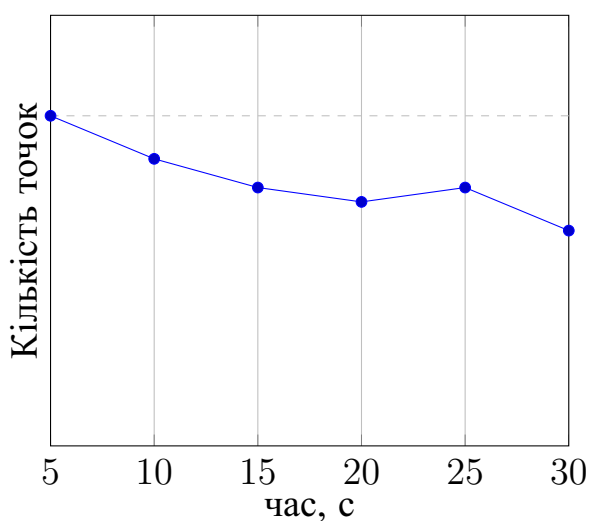


Рисунок 1.2 — Спадна ламана, побудована на основі даних з [5]

### 1.2.2 Квадратична апроксимація методом найменших квадратів

Види результуючих ламаних (опукла вгору, опукла вниз, рівна, спадна) можна представити квадратичними функціями виду

$$y(t) = a \cdot t^2 + b \cdot t + c.$$

На рис. 1.3б зображено реалізації послідовності випадкових величин, що розподілені за законом Пуассона, та криві, що є результатом апроксимації цих реалізацій.

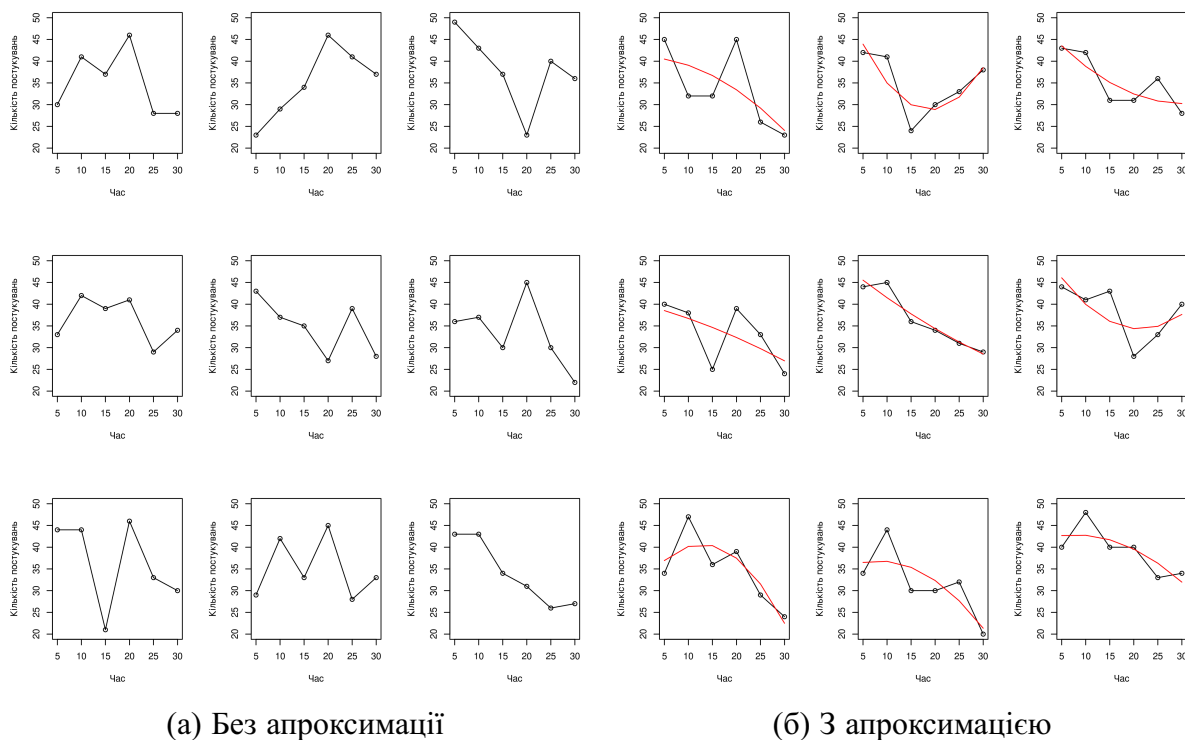


Рисунок 1.3 — Приклади результатів моделювання

Щоб представити результати у вигляді квадратичних функцій, використаємо метод найменших квадратів. Нехай  $\{y_i \mid i = \overline{1, 6}\}$  — результати поточного теплінг-тесту. Суть методу полягає в знаходженні таких коефіцієнтів  $a$ ,  $b$  і  $c$ , з якими наступна величина приймає мінімальне значення

$$R^2(a, b, c) = \sum_{i=1}^6 (a \cdot t_i^2 + b \cdot t_i + c - y_i)^2.$$

Для цього треба взяти часткові похідні по кожному коефіцієнту, прирівняти ці



похідні до нуля та розв'язати отриману систему

$$\begin{cases} \sum_{i=1}^6 2 \cdot (a \cdot t_i^2 + b \cdot t_i + c - y_i) &= 0 \\ \sum_{i=1}^6 2 \cdot t_i \cdot (a \cdot t_i^2 + b \cdot t_i + c - y_i) &= 0 \\ \sum_{i=1}^6 2 \cdot t_i^2 \cdot (a \cdot t_i^2 + b \cdot t_i + c - y_i) &= 0. \end{cases}$$

Цю систему можна переписати в матричному вигляді наступним чином

$$\begin{bmatrix} \sum_{i=1}^6 t_i^2 & \sum_{i=1}^6 t_i & \sum_{i=1}^6 1 \\ \sum_{i=1}^6 t_i^3 & \sum_{i=1}^6 t_i^2 & \sum_{i=1}^6 t_i \\ \sum_{i=1}^6 t_i^4 & \sum_{i=1}^6 t_i^3 & \sum_{i=1}^6 t_i^2 \end{bmatrix} \cdot \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^6 y_i \\ \sum_{i=1}^6 y_i \cdot t_i \\ \sum_{i=1}^6 y_i \cdot t_i^2 \end{bmatrix}.$$

Введемо позначення:

$$\begin{aligned} \sum_{i=1}^6 1 &= N, \\ \sum_{i=1}^6 t_i^k &= X^k, \quad k = \overline{1, 4}, \\ \sum_{i=1}^6 y_i \cdot t_i^k &= YX^k, \quad k = \overline{0, 2}. \end{aligned}$$

Рівняння прийняло наступний вигляд

$$\begin{bmatrix} X^2 & X & N \\ X^3 & X^2 & X \\ X^4 & X^3 & X^2 \end{bmatrix} \cdot \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} Y \\ YX \\ YX^2 \end{bmatrix}.$$

Використаємо метод Крамера для розв'язання системи лінійних рівнянь. Визначник  $\Delta$

$$\Delta = X^2 \cdot (X^2 \cdot X^2 - X \cdot X^3) - X \cdot (X^3 \cdot X^2 - X \cdot X^4) + N \cdot (X^3 \cdot X^3 - X^2 \cdot X^4).$$

Визначник  $\Delta_a$

$$\Delta_a = Y \cdot (X^2 \cdot X^2 - X \cdot X^3) - YX \cdot (X \cdot X^2 - N \cdot X^3) + YX^2 \cdot (X \cdot X - N \cdot X^2).$$

Визначник  $\Delta_b$

$$\Delta_b = -Y \cdot (X^3 \cdot X^2 - X \cdot X^4) + YX \cdot (X^2 \cdot X^2 - N \cdot X^4) - YX^2 \cdot (X^2 \cdot X - N \cdot X^3).$$

Визначник  $\Delta_c$

$$\Delta_c = Y \cdot (X^3 \cdot X^3 - X^2 \cdot X^4) - YX \cdot (X^2 \cdot X^3 - X \cdot X^4) + YX^2 \cdot (X^2 \cdot X^2 - X \cdot X^3).$$

Відомо, що розв'язками є наступні вирази

$$a = \frac{\Delta_a}{\Delta}, \quad b = \frac{\Delta_b}{\Delta}, \quad c = \frac{\Delta_c}{\Delta}.$$

Бачимо, що параметри — лінійні комбінації  $y_i$ , а коефіцієнти при  $y_i$  — раціональні

доби. Тобто можна записати розв'язки наступним чином

$$a = \sum_{i=1}^6 y_i \cdot \frac{a_i}{\Delta}, \quad b = \sum_{i=1}^6 y_i \cdot \frac{b_i}{\Delta}, \quad c = \sum_{i=1}^6 y_i \cdot \frac{c_i}{\Delta}.$$

### 1.2.3 Аналіз квадратичної апроксимації

#### 1.2.3.1 Задача

Запишемо умови, що накладаються на  $a$ ,  $b$  і  $c$ , за яких парабола

$$y(t) = a \cdot t^2 + b \cdot t + c$$

відповідає раніше визначеним типам вищої нервової діяльності. Введемо такі позначення:  $t_{max}$  і  $t_{min}$  — точки, в яких парабола досягає локального максимуму та мінімуму на відрізку  $[t_1, t_6]$  відповідно ( $t_1$  і  $t_6$  є точками першого та останнього спостереження). Оскільки маємо справу з випадковими величинами, потрібно ввести похибку  $\varepsilon$ , яка вважається допустимою. Для теппінг-тесту зазвичай беруть  $\varepsilon = 2$ . [5]

Глобального екстремуму функція  $y$  досягає в точці

$$t_g = -\frac{b}{2 \cdot a}.$$

### 1.2.3.2 Змішаний тип

Крива, що відповідає змішаному типу, весь час тримається приблизно на початковому рівні

$$\begin{cases} y(t_{max}) - y(t_1) \leq \varepsilon, \\ y(t_1) - y(t_{min}) \leq \varepsilon. \end{cases}$$

Для першої нерівності можливі три випадки:

- 1) якщо локальний максимум знаходиться в точці  $t_1$ , нерівність приймає вид  $0 \leq \varepsilon$ , що виконується завжди;
- 2) якщо локальний максимум знаходиться в точці  $t_6$ , на  $b$  накладається обмеження

$$b \leq \frac{\varepsilon}{t_6 - t_1} - a \cdot (t_6 + t_1);$$

- 3) коли глобальний максимум досягається на проміжку  $(t_1; t_6)$

$$\begin{cases} t_1 < -\frac{b}{2 \cdot a} < t_6 \\ a < 0 \end{cases} \Rightarrow t_{max} = -\frac{b}{2 \cdot a},$$

обмеження приймає більш складний вид

$$-\frac{b^2}{4 \cdot a} - b \cdot t_1 - a \cdot t_1^2 \leq \varepsilon.$$

Для другої нерівності можливі ті ж самі випадки, але з іншими обмеженнями

- 1) якщо локальний мінімум знаходиться в точці  $t_1$ , нерівність приймає вид  $0 \leq \varepsilon$ , що виконується завжди;
- 2) якщо локальний мінімум знаходиться в точці  $t_6$ , на  $b$  накладається обмеже-

ння

$$b \geq \frac{\varepsilon}{t_1 - t_6} - a \cdot (t_1 + t_6);$$

3) коли глобальний мінімум досягається на проміжку  $(t_1; t_6)$

$$\begin{cases} t_1 < -\frac{b}{2 \cdot a} < t_6 \\ a < 0 \end{cases} \Rightarrow t_{max} = -\frac{b}{2 \cdot a},$$

обмеження приймає вид

$$\frac{3 \cdot b^2}{4 \cdot a} + b \cdot t_1 + a \cdot t_1^2 \leq \varepsilon.$$

### 1.2.3.3 Слабкий тип

Крива, що відповідає слабкому типу, спадає після моменту  $t_1$  і не повертається до початкового положення

$$\begin{cases} t_{max} = t_1, \\ y(t_{max}) - y(t_{min}) > \varepsilon. \end{cases}$$

Якщо розглянути другу нерівність з урахуванням першої

$$y(t_1) - y(t_{min}) > \varepsilon,$$

в термінах  $a$  і  $b$  маємо схожі до обмежень для змішаного типу умови

1) якщо локальний максимум знаходиться в точці  $t_1$ , нерівність приймає вид

$0 > \varepsilon$ , що ніколи не виконується;

- 2) якщо локальний мінімум знаходиться в точці  $t_6$ , на  $b$  накладається обмеження

$$b < \frac{\varepsilon}{t_1 - t_6} - a \cdot (t_1 + t_6);$$

- 3) коли глобальний мінімум досягається на проміжку  $(t_1; t_6)$

$$\begin{cases} t_1 < -\frac{b}{2 \cdot a} < t_6 \\ a < 0 \end{cases} \Rightarrow t_{max} = -\frac{b}{2 \cdot a},$$

маємо обмеження

$$-\frac{b^2}{4 \cdot a} - b \cdot t_1 - a \cdot t_1^2 > \varepsilon$$

#### 1.2.3.4 Неврівноважений тип

Крива, що відповідає неуврівноваженому типу, спочатку зростає, потім не пізніше моменту часу  $t_3$  починає спадати нижче початкового рівня

$$\begin{cases} t_2 \leq t_{max} \leq t_3 < t_{min}, \\ y(t_{max}) - y(t_1) > \varepsilon, \\ y(t_1) - y(t_{min}) > \varepsilon. \end{cases}$$

Оскільки такі умови можливі лише у випадку коли  $t_{max}$  є точкою глобального максимуму, а екстремум може бути лише один  $t_{max} = -\frac{b}{2 \cdot a}$ , система при-

ймає вид

$$\left\{ \begin{array}{l} t_2 \leq -\frac{b}{2 \cdot a} \leq t_3, \\ a < 0, \\ t_{min} = t_6, \\ y\left(-\frac{b}{2 \cdot a}\right) - y(t_1) > \varepsilon, \\ y(t_1) - y(t_6) > \varepsilon. \end{array} \right.$$

Обмеження

$$\left\{ \begin{array}{l} a < 0, \\ -2 \cdot a \cdot t_2 \leq b \leq -2 \cdot a \cdot t_3, \\ b > \frac{\varepsilon}{t_6 - t_1} + a \cdot (t_6 + t_1), \\ -\frac{b^2}{4 \cdot a} - b \cdot t_1 - a \cdot t_1^2 > \varepsilon. \end{array} \right.$$

### 1.2.3.5 Рухливий тип

Крива, що відповідає рухливому типу, спочатку зростає, не пізніше моменту  $t_3$  починає спадати, а до моменту  $t_6$  повертається до початкового рівня або спадає нижче нього

$$\left\{ \begin{array}{l} t_3 \leq t_{max}, \\ y(t_{max}) - y(t_1) > \varepsilon, \\ y(t_6) \leq y(t_1). \end{array} \right.$$

Можливий лише той випадок, коли максимум є водночас локальним і гло-

бальним

$$\begin{cases} a < 0, \\ b \leq -2 \cdot a \cdot t_3, \\ b \leq \frac{\varepsilon}{t_6 - t_1} - a \cdot (t_6 + t_1), \\ -\frac{b^2}{4 \cdot a} - b \cdot t_1 - a \cdot t_1^2 \leq \varepsilon, \end{cases}$$

адже коли  $t_{max} = t_6$ , маємо суперечність

$$\varepsilon < y(t_6) - y(t_1) \leq 0.$$

### 1.2.3.6 Інертний тип

Крива, що відповідає інертному типу, спочатку спадає, а під кінець тесту може зрости до початкового рівня

$$\begin{cases} t_{min} > t_1, \\ y(t_{max}) - y(t_1) > \varepsilon. \end{cases}$$

Можливі два випадки

- 1) мінімум є водночас локальним і глобальним, тоді максимум знаходиться в кінці відрізка

$$\begin{cases} a > 0, \\ b \geq -2 \cdot a \cdot t_6, \\ -2 \cdot a \cdot t_1 > b > \frac{\varepsilon}{t_6 - t_1} - a \cdot (t_6 + t_1), \end{cases}$$



бо інакше (якщо  $t_{max} = t_1$ ) нерівність стає суперечливою

$$\varepsilon < 0;$$

- 2) мінімум знаходиться в кінці відрізка, тоді локальним максимум може до-  
рівнювати лише екстремуму

$$\begin{cases} t_{min} = t_6, \\ -\frac{b^2}{4 \cdot a} - b \cdot t_1 - a \cdot t_1^2 > \varepsilon. \end{cases}$$

### 1.2.3.7 Результати класифікації

Було згенеровано 6 вибірок по 100 реалізацій процесу Пуассона з інтенсивністю, яку було вказано у формулі (1.5). Ці реалізації відповідають проходженню теплінг-тесту шістьма групами по сто студентів. Результати класифікації можна побачити на рис. 1.4.

Опис:

- Найбільш розповсюджений тип — слабкий, що зустрівся у 60% випадків;
- 11% студентів відносяться до неврівноваженого типу;
- 7% — студенти рухливого типу;
- Студенти інертного типу зустрілися майже у 13% випадків;
- 2% — студенти зі змішаним типом вищої нервової діяльності;
- Не вдалося віднести до того чи іншого типу близько 7% студентів.

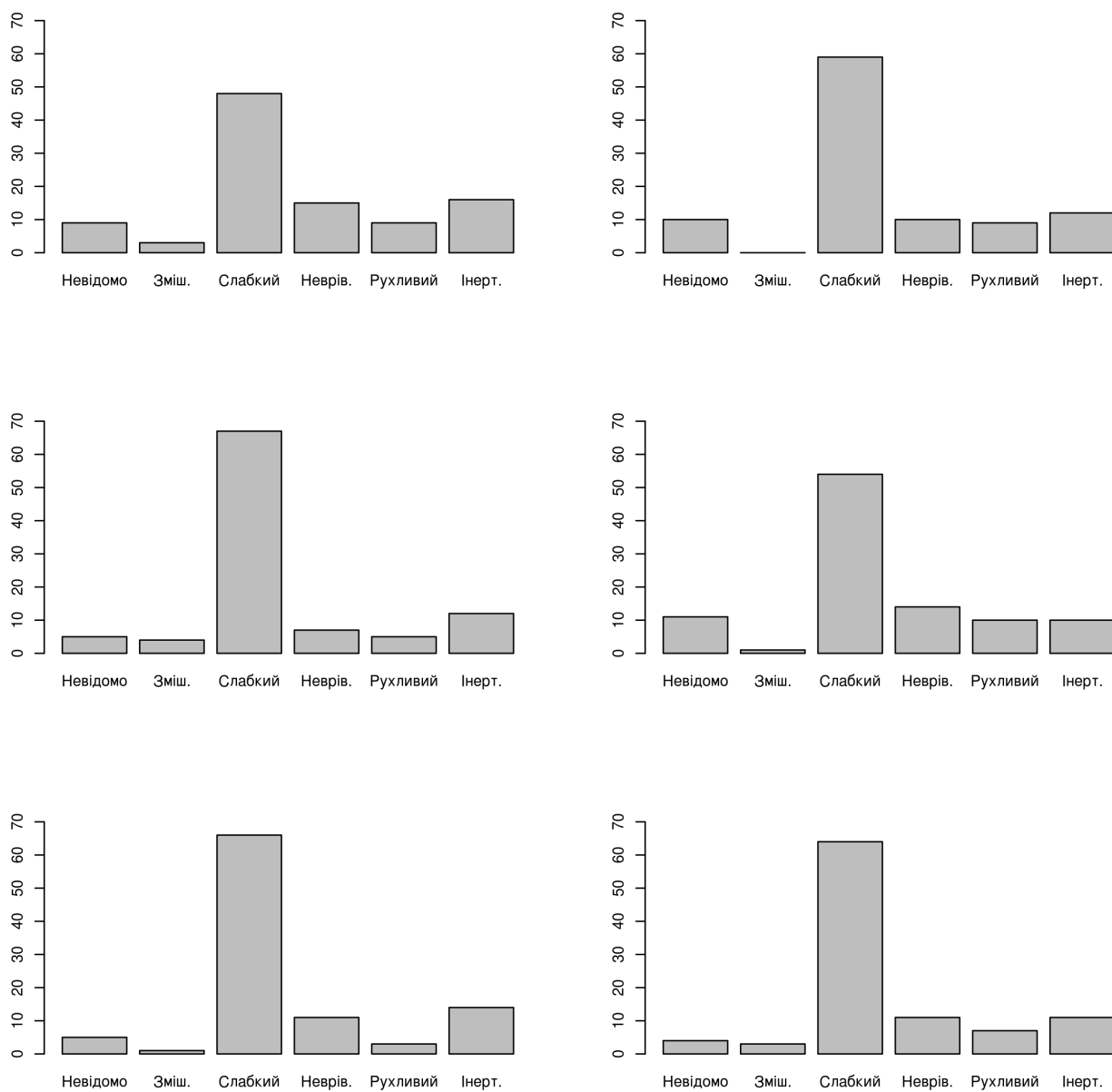


Рисунок 1.4 — Результати класифікації

## ПЕРЕЛІК ПОСИЛАНЬ

1. Айвазян, С.А. Прикладная статистика: Классификация и снижение размерности : Справочное издание / С.А. Айвазян. — М.: Финансы и статистика, 1989. — 606 с.
2. Боровков, А. А. Математическая статистика / А. А. Боровков // Учебники для ВУЗов. Специальная литература. — СПб.: Лань, 2010. — 705 с.
3. Sturges, Herbert A. The Choice of a Class Interval / Herbert A. Sturges // *J-J-AM-STAT-ASSOC.* — 1926. — March. — Vol. 21, no. 153. — Pp. 65–66.
4. Павлов, И.П. Двадцатилетний опыт объективного изучения высшей деятельности (поведения) животных / И.П. Павлов. — М.: Рипол Классик, 1973. — 659 с.
5. Ильин, Е.П. Дифференциальная психофизиология / Е.П. Ильин // Серия Учебник нового века. — СПб.: Питер, 2001. — 464 с.
6. Булинский, А.В. Теория случайных процессов / А.В. Булинский, А.Н. Ширяев // Теория вероятностей. Математическая статистика : ТВМС. — М.: Физматлит, 2003. — 399 с.
7. Kingman, J.F.C. Poisson Processes / J.F.C. Kingman // Oxford studies in probability. — Oxford: Clarendon Press, 1992. — 112 pp.

## ДОДАТОК А

### Лістинг коду програми, яка моделює результати теппінг-тесту

```

1  library(NHPoisson)
2  source("generate_sequence.r")
3  source("analyze_approximation.r")
4  wideScreen()
5  # Read command line arguments
6  args <- commandArgs(TRUE)
7  display_approximation <- FALSE
8  if (length(args) > 0 && args[1] == "approximation") {
9      display_approximation <- TRUE
10 }
11 # Init cells for charts
12 n <- 3
13 m <- 3
14 png('output.png', width=18, height=22, units="cm", res=300)
15 par(mfrow=c(n,m))
16 chart.xlab <- "Час"
17 chart.ylab <- "Кількість постукувань"
18 chart.ylim <- c(20,50)
19 # Parameters
20 sample.time <- seq(5, 30, 5)
21 sample.size = length(sample.time)
22 sample.prediction_degree = 2
23 sample.poly = poly(sample.time, sample.prediction_degree, raw=TRUE)
24 #lambda.default = c(40, 38, 36, 34, 32, 30, 0)
25 lambda.default = rep(40, sample.size+1)
26 # Fill every cell
27 for (i in 1:(n*m)) {
28     # Generate non-homogeneous Poisson process trajectory
29     sample.current <- generate_trajectory(lambda.default)
30     # Plot
31     plot(sample.time, sample.current, xlab=chart.xlab, ylab=chart.ylab,
32         ylim=chart.ylim)
33     lines(sample.time, sample.current)
34     if (display_approximation) {
35         # Calculate approximation
36         sample.model <- lm(sample.current ~ sample.poly)
37         sample.prediction <- predict(sample.model)
38         extrema <- find_extrema(sample.model, sample.time)
39         # Draw approximation
40         lines(sample.time, sample.prediction, col='red')
41         # Draw start rhythm
42         abline(h=sample.model$coefficients[3]*sample.time[1]^2 +
43             sample.model$coefficients[2]*sample.time[1] +
44             sample.model$coefficients[1], col='green', lty=2, lwd=1)
45         # Draw mininum and maximum

```

```
46      abline(v=extrema[['min']], col='gray', lty=3, lwd=1)
47      abline(v=extrema[['max']], col='blue', lty=3, lwd=1)
48      # Display group
49      print(get_group(sample.model, sample.time,
50                    extrema[['min']], extrema[['max']]))
51    }
52  }
53  dev.off()
```

## ДОДАТОК Б

### Лістинг коду модуля, що генерує процес Пуассона

```
1 generate_trajectory <- function(lambda.default) {  
2   # Generate non-homogeneous Poisson process  
3   sample_trajectory <- simNHP.fun(lambda=lambda.default)  
4   # Calculate number of events in every sample.time interval  
5   # (we have sample.size of them)  
6   events_on_step <- function(x) length(sample_trajectory[  
7                                     sample_trajectory$posNH==x])  
8   sample.current <- unlist((Map(events_on_step, 1:(sample.size))))  
9   return(sample.current)  
10 }
```

## ДОДАТОК В

Лістинг коду програми, яка класифікує моделі студентів за типами вищої нервової діяльності

```

1  find_extrema <- function(model, time) {
2    a = model$coefficients[3]
3    b = model$coefficients[2]
4    c = model$coefficients[1]
5
6    t.start      = head(time, n=1)
7    t.finish     = tail(time, n=1)
8
9    y <- function(t) a*(t^2) + b*t + c
10   first_derivative <- function(t) 2*a*t + b
11   second_derivative <- function(t) 2*a
12
13   global_extremum <- - b / (2*a)
14
15   initial_extremum = global_extremum
16   if (initial_extremum < t.start) {
17     initial_extremum <- t.start;
18   }
19   else if (initial_extremum > t.finish) {
20     initial_extremum <- t.finish;
21   }
22
23   t.min <- Reduce(function(x, n) ifelse(y(x) < y(n), x, n),
24                   c(t.start, t.finish), initial_extremum);
25   t.max <- Reduce(function(x, n) ifelse(y(x) > y(n), x, n),
26                   c(t.start, t.finish), initial_extremum);
27   list(min=t.min, max=t.max)
28 }
29
30 get_group <- function(model, time, t_min, t_max) {
31   a = model$coefficients[3]
32   b = model$coefficients[2]
33   c = model$coefficients[1]
34   EPSILON = 2
35   y <- function(t) a*(t^2) + b*t + c
36
37   if (y(t_max) - y(time[1]) <= EPSILON && y(time[1]) - y(t_min) <= EPSILON) {
38     return(0)
39   }
40   else if(abs(t_max - time[1]) < 1E-5 && y(t_max) - y(t_min) > EPSILON) {
41     return(1)
42   }
43   else if(time[2] <= t_max && t_max <= time[3] && time[3] < t_min

```

```

44         && y(t_max) - y(time[1]) > EPSILON && y(time[1]) - y(t_min) > EPSILON) {
45             return(2)
46         }
47     else if(time[3] <= t_max && y(t_max) - y(time[1]) > EPSILON
48         && y(time[6]) <= y(time[1])) {
49         return(3)
50     }
51     else if(t_min > time[1] && y(t_max) - y(time[1]) < EPSILON) {
52         return(4)
53     }
54     return(-1)
55 }

```