

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Харківський національний університет імені В.Н.Каразіна

Факультет математики і інформатики

Кафедра теоретичної та прикладної інформатики

Кваліфікаційна робота

магістр

на тему «Дослідження продуктивності Log-rank тесту при використанні
різної ваги у клінічних дослідженнях»

Виконав: студент 2 курсу, групи МФ-61
спеціальність 122 «Комп'ютерні науки»
освітньо-наукова програма «Інформатика»

Лось Євген Олександрович

Керівник: Ватуля Г.Л., доктор технічних наук

Рецензент: Левчук І.П., кандидат фізико-
математичних наук

Харків – 2024 року

ЗМІСТ

ВСТУП	4
Мета роботи та обґрунтування актуальності теми	4
Огляд відомих результатів в області дослідження	4
Відомості про одержані результати та їх новизна	5
РОЗДІЛ 1. ПОСТАНОВКА ЗАДАЧІ ТА ОГЛЯД СУЧАСНОГО СТАНУ СПРАВ В ОБЛАСТІ КЛІНІЧНИХ ДОСЛІДЖЕНЬ	6
1.1. Постановка задачі	6
1.2. Розвинутий огляд сучасного стану справ в області дослідження	7
1.2.1. Клінічні дослідження та аналіз виживання	7
1.2.2. Клінічні дослідження зі збагаченням	9
1.2.3. Аналіз виживання у клінічних дослідженнях	9
1.2.4 Time-to-event аналіз	10
РОЗДІЛ 2. АНАЛІЗ МЕТОДІВ ДОСЛІДЖЕННЯ	12
2.1 Лог-ранг тест	12
2.2 Нестратифікований лог-ранг тест	13
2.3 Модифікований лог-ранг тест	13
2.4 Стратифіковані тести	15
РОЗДІЛ 3. АНАЛІЗ ПРОДУКТИВНОСТІ ЛОГ-РАНГ ТЕСТУ ПРИ ВИКОРИСТАННІ РІЗНИХ ВАГОВИХ ФУНКЦІЙ	16
3.1 Описання та обґрунтування алгоритмів та результатів дослідження ...	16
3.1.1. Симуляція даних	16
3.1.2 Розмір вибірки і кількість симуляцій	16
3.1.3. Розподіл даних	16

3.1.4. Пропорція підгруп	18
3.1.5. Пропорція цензорингу	19
3.1.6. Вигляд функції ризику для підгруп	21
3.1.7. Лог-ранг тест та вивід результатів	28
3.2 Програмні характеристики продукту	30
3.3 Приклади обчислювальних експериментів та аналіз результатів	32
ВИСНОВКИ	38
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	39
ДОДАТКИ	41
Додаток 1. Код програми на мові SAS	41

ВСТУП

Мета роботи та обґрунтування актуальності теми

У даній роботі досліджується продуктивність результатів, отриманих при застосуванні лог-ранг тесту з різними ваговими функціями до даних клінічних досліджень, які аналізують час до настання події.

Клінічні дослідження є однією з ключових галузей фармацевтичної індустрії. Аналіз виживання проводиться для багатьох клінічних досліджень, наприклад, повністю виправдовуючи свою назву у дослідженнях онкології. Одне з ключових завдань, які ставляться у цьому аналізі — чи є різниця між функціями виживання/ризик у групах пацієнтів, які приймають різні ліки, що свідчило б про більшу ефективність якихось із них. Це порівняння проводиться за саме за допомогою лог-ранг тесту, і саме базуючись на цих результатах робиться висновок про те, чи приносить застосування лікарського препарату користь. Таким чином, дані про продуктивність результатів цього тесту у специфічних умовах можуть мати вирішальне значення для вибору специфічних параметрів тесту, та, як наслідок, результату всього дослідження.

Огляд відомих результатів в області дослідження

Лог-ранг тест — це стандартний метод аналізу даних про виживання з багатою історією застосування в індустрії. Однак зазвичай він застосовується в припущенні, що усі групи пацієнтів в дослідженні мають однакові функції ризику. Останнім часом набирають популярність так звані клінічні дослідження за збагаченням (популяції пацієнтів), де збагачення покликане збільшити кількість пацієнтів, які добре реагують на досліджуваний препарат. Таке збагачення, цілком імовірно, може призвести до того, що збагачені групи пацієнтів отримають нові функції ризику або пропорції цензорованих даних. Автору невідомо про дослідження поведінки лог-ранг тесту при таких припущеннях, що є одним з факторів актуальності роботи.

Відомості про одержані результати та їх новизна

Було розроблено комп'ютерну програму (макрос) на мові програмування SAS, яка досі залишається стандартом у індустрії аналізу даних клінічних досліджень. Програма дозволяє симулювати дані про виживання у клінічному дослідженні, з можливістю налаштування розподілу даних, пропорції цензорованих пацієнтів, та вигляду функції ризику у підгрупах дослідної групи. Ці дані аналізуються лог-ранг тестом з можливістю вибору різних вагових функцій, що дає можливість оцінити їх відносну ефективність. Автору роботи невідомо про таку функціональність у стандартних програмних продуктах в індустрії, що обґрунтовує новизну результатів.

РОЗДІЛ 1. ПОСТАНОВКА ЗАДАЧІ ТА ОГЛЯД СУЧАСНОГО СТАНУ СПРАВ В ОБЛАСТІ КЛІНІЧНИХ ДОСЛІДЖЕНЬ

1.1. Постановка задачі

Клінічні дослідження — невід’ємна частина фармацевтичної індустрії, яка тестує безпечність та ефективність нових лікувальних препаратів або процедур [1]. Рандомізовані плацебо-контрольовані клінічні дослідження з подвійним засліпленням є золотим стандартом доказової медицини для вивчення ефекту лікування у загальній популяції [2].

Аналіз виживання є важливим інструментом у клінічних дослідженнях, спрямованих на вивчення часу до настання подій, таких як смерть, рецидив захворювання або відновлення здоров'я. Цей аналіз дозволяє визначити часові моменти, коли відбуваються події в ході спостереження, та оцінити ймовірність настання цих подій у певний час.

Один з основних методів порівняння time-to-event даних - це лог-ранг тест (log-rank test). Цей непараметричний статистичний тест використовується для порівняння кривих виживання між двома або більше групами пацієнтів. Він оцінює, чи є статистично значима різниця між розподілами часу до настання події в різних групах. Лог-ранг тест розраховує спостережений та очікуваний рангові суми на основі виживальних даних кожної групи, і порівнює їх для визначення статистичної значущості різниці між кривими виживання.

Вагові функції в лог-ранг тесті відіграють ключову роль у визначенні значущості різниці між групами. Вони використовуються для надання більшого значення даним про пацієнтів з більш значущими часами виживання або з подіями, ніж даним про пацієнтів з менш значущими результатами.

Продуктивність лог-ранг тесту при використанні різних функцій ваги до даних, у яких групи пацієнтів мають різні відношення ризиків (hazard ratio) досліджена слабо. Така ситуація як раз можлива у клінічних дослідженнях зі збагаченням. Наприклад, пацієнти можуть бути розбиті на підгрупи, одна з яких

буде отримувати ефект поступово, в той час як інша – відразу отримати значний ефект, але потім його втратити (порівняно з контрольною групою).

Саме тому було вирішено розробити програму, яка б оцінювала ефективність цього тесту при різних відношеннях ризиків для популяцій.

1.2. Розвинутий огляд сучасного стану справ в області дослідження

1.2.1. Клінічні дослідження та аналіз виживання

Клінічні дослідження — це головний інструмент доказової медицини для оцінювання ефективності та безпеки нових засобів та методів лікування. Коректно спроектоване клінічне дослідження забезпечує наукову достовірність та вітдворюваність своїх результатів. Більшість країн вимагають для визнання препаратів/процедур лікувальними проведення відповідних клінічних досліджень, що робить їх невід’ємною частиною фармакологічної індустрії.

В залежності від типу досліджуваної речовини та стадії вивченості її лікувального ефекту, дослідники зазвичай спочатку проводять пілотні дослідження з невеликою кількістю волонтерів, і тільки потім переходять до великих порівняльних досліджень. Це дозволяє мінімізувати кількість останніх, адже, поміж іншого, мультицентрові міжнародні клінічні дослідження з сотнями пацієнтів можуть обходитись фармацевтичним компаніям у сотні ж мільйонів доларів. Тому зазвичай досліджувана речовина вивчається у клінічних дослідженнях різних фаз [3]:

1. Перша фаза — визначення безпеки препарату. В таких дослідженнях препарат дається невеликій групі здорових волонтерів (зазвичай до 20), і вивчається його фармакодинаміка, фармакокінетика, встановлюються границі безпечних доз та побічні ефекти.
2. Друга фаза — попереднє дослідження ефективності препарату, зазвичай в різних дозуваннях. В дослідженнях цієї фази його дають пацієнтам з цільовою хворобою, і зазвичай порівнюють

ефективність з групою плацебо. Такі дослідження зазвичай набирають до 100 пацієнтів з цільовою хворобою. Вони дозволяють оцінити потенційний лікувальний ефект препарату, але зазвичай не мають статистичної потужності для узагальнення цих результатів на загальну популяцію.

3. Третя фаза — остаточне дослідження безпеки та ефективності препарату для загальної популяції. Використовуючи оптимальні дозування, встановлені в попередніх фазах, дослідники набирають велику кількість пацієнтів (сотні/тисячі) для отримання статистично значущих результатів про ефективність та безпеку препарату. Саме на підставі цих результатів регулюючі організації виносять висновок про допуск лікарського засобу на ринок — для цього він повинен продемонструвати лікувальний ефект і якусь перевагу над уже існуючими засобами.
4. Четверта фаза — спостереження над препаратом під час його використання у загальній популяції. Це дозволяє отримати ще більше даних про ризики, переваги, використання препарату без нагляду лікарів, і т.д.

Найбільшими, і, як наслідок, найдорожчими дослідженнями є дослідження третьої фази. Вартість кожного з них може сягати десятків і сотень мільйонів доларів [4]. Саме тому фармацевтичні компанії шукають способи мінімізації їх вартості. Один з параметрів, від якого прямо вартість клінічного дослідження — це кількість пацієнтів (і країн), які приймають в ньому участь. Як наслідок, при дизайні дослідження біостатистики намагаються обрати мінімальну кількість учасників, яка, ймовірно, дасть результат достатньої статистичної потужності.

За даними FDA [4], тільки 10% усіх речовин, які доходять до стадії клінічних досліджень, отримують дозвіл розповсюдження в США.

1.2.2. Клінічні дослідження зі збагаченням

Клінічні дослідження зі збагаченням - це стратегія, яка спрямована на включення пацієнтів у дослідження з вибіркоvim або стратегічним підходом залежно від їхніх характеристик, таких як генетичні особливості, клінічний статус чи відгук на попередні лікування [6]. Наприклад, один із видів клінічних досліджень зі збагаченням - це дослідження зі збагаченням за прогностичними маркерами, де пацієнти відбираються на основі вмісту певних маркерів, що вказують на вірогідність певного ризику або переваги від конкретного лікування. Це дозволяє збільшити ефективність дослідження за допомогою зосередження на пацієнтах, які найбільш імовірно позитивно відреагують на лікування, тим самим зменшуючи кількість учасників і витрати [7].

Інший тип клінічних досліджень зі збагаченням - це дослідження зі збагаченням за реакцією на лікування. У цьому випадку пацієнти розподіляються на підгрупи в залежності від їхньої реакції на попередні курси лікування, дозволяючи вивчати ефективність лікування у специфічних підгрупах пацієнтів. Цей підхід сприяє розвитку персоналізованої медицини, де лікування може бути налаштоване на основі індивідуальних характеристик кожного пацієнта.

Клінічні дослідження зі збагаченням стають все більш популярними у світлі зростання інтересу до прецизійної медицини та індивідуального підходу до лікування [8].

1.2.3. Аналіз виживання у клінічних дослідженнях.

Аналіз виживання є важливим інструментом у клінічних дослідженнях, спрямованих на вивчення часу до настання подій, таких як смерть, рецидив захворювання або відновлення здоров'я. Цей аналіз дозволяє визначити часові моменти, коли відбуваються події в ході спостереження, та оцінити ймовірність настання цих подій у певний час [9]. Основною метою аналізу виживання є оцінка впливу різних факторів на тривалість часу до подій, що досліджуються. Для цього

використовуються методи, такі як криві виживання, криві ризику, моделі Сох та інші статистичні підходи, які дозволяють врахувати різноманітність факторів і взаємодію між ними.

Одним із ключових переваг аналізу виживання є можливість врахування цензорованих даних, коли інформація про час настання події недоступна або неповна. Це дозволяє врахувати ситуації, коли спостереження закінчується до настання події, або коли стан пацієнта не спостерігається до кінця дослідження. Аналіз виживання використовується в різних областях медицини, включаючи онкологію, кардіологію, епідеміологію та клінічні випробування лікарських засобів. Його результати допомагають приймати обґрунтовані рішення щодо лікування, розробки нових методів діагностики та терапії, а також управління пацієнтами з різними захворюваннями.

1.2.4 Time-to-event аналіз

У клінічних дослідженнях часто використовуються методи порівняння часу до настання подій (time-to-event data), таких як смерть, рецидив захворювання або відновлення здоров'я. Ці методи дозволяють оцінити ефективність лікування або вплив різних факторів на тривалість виживання пацієнтів [10].

Один з основних методів порівняння time-to-event даних - це лог-ранг тест. Цей непараметричний статистичний тест використовується для порівняння кривих виживання між двома або більше групами пацієнтів. Він оцінює, чи є статистично значима різниця між розподілами часу до настання події в різних групах. Лог-ранг тест розраховує спостережений та очікуваний рангові суми на основі виживальних даних кожної групи, і порівнює їх для визначення статистичної значущості різниці між кривими виживання.

Інший метод порівняння time-to-event даних - це використання моделей виживання, таких як модель Кокса або модель Вейбулла [11]. Ці моделі дозволяють врахувати вплив різних факторів на тривалість виживання та вивчити їхню взаємодію. Порівняння між групами може бути здійснене за допомогою

випробування взаємодії в моделях виживання, або шляхом порівняння коефіцієнтів ризику між групами.

РОЗДІЛ 2. АНАЛІЗ МЕТОДІВ ДОСЛІДЖЕННЯ

2.1 Лог-ранг тест

Нехай K позначає кількість груп, $S_k(t)$ — функцію виживання для k -ї групи, $k = 1, \dots, K$. Нульова і альтернативна гіпотези, які перевіряються виглядають як

$$H_0: h_1(t) = h_2(t) = \dots = h_k(t), t \leq \tau$$

$$H_1: \exists j, g \in 1, \dots, k; t \leq \tau: h_j(t) \neq h_g(t)$$

Нехай $(T_i, \delta_i, X_i), i = 1, \dots, n$ позначають незалежну вибірку цензорованих справа даних про виживання, де T_i — (можливо, цензорований справа) час, δ_i — індикатор цензорингу (наприклад, $\delta_i = 0$ якщо T_i цензоровано, і $\delta_i = 1$ якщо T_i — це час настання події), а $X_i = 1, \dots, K$ для K різних груп.

Нехай $t_1 < t_2 < \dots < t_D$ позначають різні моменти настання події у вибірці.

Для моменту часу $t_j, j = 1, \dots, D$ позначимо через W_j додатню вагову функцію, через $Y_{jk} = \sum_{i: T_i \geq t_j} I(X_i = k)$ — величину множини пацієнтів, що перебувають у зоні ризику, а через $d_{jk} = \sum_{i: T_i = t_j} \delta_i I(X_i = k)$ — кількість подій у k -й групі.

Позначимо також $Y_j = \sum_{k=1}^K Y_{jk}$, і позначимо $d_j = \sum_{k=1}^K d_{jk}$.

Можливі варіанти для вагової функції наведено у таблиці нижче:

Тест	$W(t_j)$
Лог-ранг (log-rank)	1
Вілкоксон (Wilcoxon)	Y_j
Тарона-Варе (Tarone-Ware)	$\sqrt{Y_j}$
Пето-Пето (Peto-Peto)	$S(t_j)$
Модифікований Пето-Пето	$\tilde{S}(t_j) \frac{Y_j}{Y_j + 1}$
Гаррінгтона-Флемінга (p, q) (Harrington-Fleming)	$(\hat{S}(t_j))^p (1 - \hat{S}(t_j))^q v_K, p \geq 0, q \geq 0$

Таблиця 1. Вагові функції для непараметричного лог-ранг тесту, які дають назву його варіантам.

В Таблиці 1 $\hat{S}(t)$ — це оцінка Каплана-Майєра в момент часу t для загальної вибірки, а $\tilde{S}(t)$ — це оцінка функції виживання, яка близька до $\hat{S}(t)$, і обчислюється за наступною формулою: $\tilde{S}(t) = \prod_{t_j \leq t} (1 - \frac{d_j}{Y_{j+1}})$.

2.2 Нестратифікований лог-ранг тест

Рангова статистика для перевірки H_0/H_1 має вигляд K -розмірного вектора $v = (v_1, v_2, \dots, v_K)'$ [12], де

$$v_k = \sum_{j=1}^D \left[W(t_j) (d_{jk} - Y_{jk} \frac{d_j}{Y_j}) \right].$$

Дисперсія і коваріація v_k та v_h будуть дорівнювати, відповідно,

$$V_{kk} = \sum_{j=1}^D \left[W^2(t_j) \frac{d_j(Y_j - d_j)Y_{jk}(Y_j - Y_k)}{Y_j^2(Y_j - 1)} \right], 1 \leq k \leq K$$

$$V_{kh} = - \sum_{j=1}^D \left[W^2(t_j) \frac{d_j(Y_j - d_j)Y_{jk}Y_{jh}}{Y_j^2(Y_j - 1)} \right], 1 \leq k \neq h \leq K.$$

Статистику v_K можна розглядати як зважену суму вибірових мінус очікуваних подій для k -ї групи для випадку, коли вірна нульова гіпотеза про те, що криві виживання ідентичні. Позначимо через V матрицю усіх коваріацій: $V = (V_{kh})$. Тоді загальна тестова статистика для гомогенності — це $v'V^-v$, де V^- — це генералізована зворотня до V . Ця статистику можна розглядати як таку, що має розподіл хі-квадрат з кількістю ступеней свободи, яка дорівнює рангу V при обчисленні рівней імовірностей.

2.3 Модифікований лог-ранг тест

У процедурі PROC LIFETEST мови програмування SAS є можливість використовувати ваги для вибірових значень, вказані у твердженні WEIGHT [13]. Нехай $(T_i, \delta_i, X_i, \omega_i), i = 1, \dots, n$ позначають незалежну вибірку цензорованих

справа даних про виживання, де T_i — (можливо, цензорований справа) час, δ_i — індикатор цензорингу (наприклад, $\delta_i = 0$ якщо T_i цензоровано, і $\delta_i = 1$ якщо T_i — це час настання події), а $X_i = 1, \dots, K$ для K різних груп, а ω_i — ваги з твердження WEIGHT.

Нехай $t_1 < t_2 < \dots < t_D$ позначають різні моменти настання події у вибірці.

Для моменту часу $t_j, j = 1, \dots, D$ позначимо:

$$Y_{jk} = \sum_{i:T_i \geq t_j} I(X_i = k); Y_{jk}^\omega = \sum_{i:T_i \geq t_j} \omega_i I(X_i = k);$$

$$d_{jk} = \sum_{i:T_i = t_j} \delta_i I(X_i = k); d_{jk}^\omega = \sum_{i:T_i = t_j} \omega_i \delta_i I(X_i = k).$$

Нехай $d_j^\omega = \sum_{k=1}^K d_{jk}^\omega$ — зважена кількість подій, а $Y_j^\omega = \sum_{k=1}^K Y_{jk}^\omega$ — зважена кількість пацієнтів у зоні ризику у момент часу t_j . Тоді тестова статистика має вигляд

$$v_k = \sum_{j=1}^D \left[d_{jk}^\omega - Y_{jk}^\omega \frac{d_j^\omega}{Y_j^\omega} \right], k = 1, \dots, K,$$

а дисперсія та коваріація v_k та v_h , відповідно,

$$V_{kk} = \sum_{j=1}^D \left[\frac{d_j(Y_j - d_j)}{Y_j(Y_j - 1)} \sum_{i=1}^{Y_j} \left\{ \left(\frac{Y_{jk}^\omega}{Y_j^\omega} \right)^2 \omega_i^2 I(X_i \neq k) + \left(\frac{Y_j^\omega - Y_{jk}^\omega}{Y_j^\omega} \right)^2 \omega_i^2 I(X_i = k) \right\} \right],$$

для $1 \leq k \leq K$;

$$V_{kh} = \sum_{j=1}^D \left[\frac{d_j(Y_j - d_j)}{Y_j(Y_j - 1)} \sum_{i=1}^{Y_j} \left\{ \frac{Y_{jk}^\omega Y_{jh}^\omega}{(Y_j^\omega)^2} \omega_i^2 I(X_i \neq k, h) - \frac{(Y_j^\omega - Y_{jk}^\omega) Y_{jh}^{\omega^2}}{(Y_j^\omega)^2} \omega_i^2 I(X_i = k) \right. \right. \\ \left. \left. - \frac{(Y_j^\omega - Y_{jh}^\omega) Y_{jk}^{\omega^2}}{(Y_j^\omega)^2} \omega_i^2 I(X_i = h) \right\} \right], 1 \leq k \neq h \leq K.$$

Нехай $V = (V_{kh})$. При H_0 зважений K -вибірковий тест має ститистику, розподілену за хі-квадрат наступного вигляду: $\chi^2 = (v_1, \dots, v_K) V^{-1} (v_1, \dots, v_K)'$, з $K - 1$ ступенями свободи.

2.4 Стратифіковані тести

Розглянемо випадок, коли тест стратифікується на M рівнях для певного набору ознак (в SAS змінні для стратифікації можна передати ключовим словом STRATA). Для даних тільки по s -му стратуму ($s = 1, \dots, M$), нехай v_s позначає тестову статистику для цього стратуму, а V_s — матрицю коваріацій. Позначимо

$$v = \sum_{s=1}^M v_s$$
$$V = \sum_{s=1}^M V_s$$

Тоді глобальна тестова статистика будується як $\chi^2 = v'V^{-1}v$, і має у випадку справедливості нульової гіпотези розподіл хі-квадрат з таким самим числом ступенів свободи, як у індивідуального тесту для кожного стратуму.

РОЗДІЛ 3. АНАЛІЗ ПРОДУКТИВНОСТІ ЛОГ-РАНГ ТЕСТУ ПРИ ВИКОРИСТАННІ РІЗНИХ ВАГОВИХ ФУНКЦІЙ

3.1 Описання та обґрунтування алгоритмів та результатів дослідження

3.1.1. Симуляція даних

У першій, критично важливій частині програми ми розглянемо симуляцію наших даних про виживання. Параметри, які ми розглядатимемо — це розмір вибірки, кількість симуляцій, види розподілу даних, пропорції підгруп, пропорції цензорингу в кожній групі, та вигляд функції ризику. Кожен з цих факторів матиме значний вплив на симуляцію, і ми детально розглянемо SAS-код, який їх реалізує.

3.1.2 Розмір вибірки і кількість симуляцій

Розмір вибірки для клінічного дослідження — це ключовий параметр, який враховують при його дизайні біостатистики. Вони враховують багато факторів (таких як бюджет, швидкість набору пацієнтів, прогнозована статистична потужність результату, тощо), і проводять симуляції для того, щоб встановити конкретний розмір вибірки. В нашій програмі його можна вказати за допомогою параметра (макрозмінної) `study_n`. Ми припускаємо, що пропорція кількості пацієнтів групи, яка отримує досліджуваний препарат, та контрольної групи однакові. Ми також використовуємо параметр `simulation_n`, за допомогою якого користувач може вказати бажану кількість симуляцій.

3.1.3. Розподіл даних

Наша програма передбачає використання двох варіантів розподілу: експоненціального та Вейбула для симуляції даних про виживання (тобто результуючі дані будуть мати один з цих розподілів). Деякі дані про ці розподіли наведено у таблиці 2:

Розподіл	Експоненціальний	Вейбула
Функція ризику, $h(x)$	$\frac{1}{\sigma}$	$\frac{a}{b^a} x^{a-1}$
Функція виживання, $S(x)$	$e^{-\frac{x}{\sigma}}$	$e\left(\left(-\frac{x}{b}\right)^a\right)$
Функція щільності імовірності, $f(x)$	$\frac{1}{\sigma} e^{-\frac{x}{\sigma}}$	$\frac{a}{b^a} x^{a-1} e\left(\left(-\frac{x}{b}\right)^a\right)$
Прогнозоване виживання, $E(x)$	σ	$b\Gamma\left(1 + \frac{1}{a}\right)$

Таблиця 2. Характеристики експоненціального та розподілу Вейбула даних про виживання.

Як можна побачити у коді макроса нижче, спочатку користувач задає значення «EXPONENTIAL» або «WEIBULL» макрозмінній `data_distr`, щоб вказати розподіл для даних. Потім симульовані дані створюються за допомогою SAS-функції `rand(&data_distr.,parameter-1,...,parameter-k)`, в яку передаються параметри розподілу.

Значення цих параметрів відображають прогнозовані результуючі параметри ефективності лікування. Для експоненціального розподілу, макрозмінна `exp_h_hr`, значення якої становить $\frac{1}{\sigma}$, описує рівень ризику після початку лікування. Для контрольної групи цей ризик залишається сталим на протязі всього перебігу дослідження, тож для неї вся симуляція проходить з цим параметром. Однак для дослідної групи цей ризик змінюється з часом. Ми детально опишемо цей процес в наступних розділах, однак зараз зауважимо, що задане користувачем значення `exp_h_hr` в поточній програмі також використовується для симуляції початкових подій у дослідній групі.

Для розподілу Вейбула ми проводимо схожий процес: користувач задає значення макрозмінних `wb_a` та `wb_b`, які відповідають a та b з попередньої таблиці. Далі буде проведена симуляція даних для контрольної групи, а зміни у симуляції для дослідної групи будуть розглянуті пізніше.

В подальшому ми будемо називати контрольну групу Група 1, а дослідну групу — Група 2.

```

%**create randomized event time and censor status for specified distribution;
%if %sysfunc(upcase(&data_distr.)) = "EXPONENTIAL" %then %do;
    evnt_t = rand("EXPONENTIAL", 1/(&exp_h_hr.));
    cnsr_t = rand("EXPONENTIAL", 1/(&exp_hr_cnsr.));
%end;

%else %if %sysfunc(upcase(&data_distr.)) = "WEIBULL" %then %do;
    evnt_t = rand("WEIBULL", &wb_a., &wb_b.);
    cnsr_t = rand("WEIBULL", &wb_a., &wb_cnsr_time.); ;
%end;

```

3.1.4. Пропорція підгруп

Як ми вказували у попередніх розділах, наша програма описує випадок двох груп (контрольної та дослідної) та двох підгруп. Ми припускаємо, що різні пропорції підгруп можуть впливати на результуючу потужність лог-ранг тесту. Тому для зручності користувачів ми передбачили макрозмінну `sgrp1_prop`, через яку вони можуть передати значення пропорції, яку займає перша група. Відповідно, друга група автоматично буде використовувати пропорцію $1 - \text{sgrp1_prop}$. Таблиця нижче показує кілька варіантів симуляції дослідження з різними пропорціями:

Сценарій	Група	Підгрупа 1	Підгрупа 2
Сценарій 1	Контрольна (1)	0.5	0.5
	Дослідна (2)	0.5	0.5
Сценарій 2	Контрольна (1)	0.4	0.6
	Дослідна (2)	0.4	0.6
Сценарій 3	Контрольна (1)	0.3	0.7
	Дослідна (2)	0.3	0.7
Сценарій N	Контрольна (1)
	Дослідна (2)

Таблиця 3. Приклади симуляцій дослідження з різними пропорціями підгруп.

3.1.5. Пропорція цензорингу

Цензоринг відіграє надзвичайно важливу роль в аналізі виживання. Для спрощення симуляції та аналізу даних ми припускаємо, що цензоринг трапляється незалежно від настання подій. З іншого боку, пропорція цензованих даних має значний ефект в аналізі виживання, яким не можна нехтувати. Ми будемо вважати цю пропорцію ідентичною для обох груп.

Для контролю над цензорингом ми використовуємо макрозмінну `cnsr_prop`, в яку користувачі можуть передати значення від 0 до 1. Після того як користувач задав пропорцію, ми обчислюємо відповідні параметри розподілу згідно з нею та рівнем ризику, тобто $\frac{h_c(x)}{h_e(x)} = \frac{P_c}{P_e} = \frac{P_c}{1-P_c}$, де $h_c(x)$ та $h_e(x)$ — це рівні ризику (hazard rate) для цензованих даних та даних з подіями, а P_c та P_e — пропорції цензованих даних та подій.

Як видно у фрагменті коду наведеному нижче, оскільки рівень ризику для експоненційного розподілу є сталим, рівень ризику для цензованих даних обчислюється просто як $\frac{P_c}{1-P_c} h_e(x)$. Користувач може задати значення P_c та $h_e(x)$ за допомогою макрозмінних `cnsr_prop` та `exp_h_hr`. Далі програма створює макрозмінну `exp_hr_cnsr`, в яку зберігається обчислений рівень ризику для цензованої симуляції у експоненціальному розподілі.

Для розподілу Вейбула рівень ризику становить $\frac{a}{b^a} x^{a-1}$, і в припущенні що a фіксоване, для конкретної події та цензованих даних ми отримуємо $b_c = b_e \left(\frac{1-P_c}{P_c} \right)^{\frac{1}{a}}$, де b_c та b_e — параметри b розподілу Вейбула для цензованих даних та даних з подіями. Далі ми використовуємо таким чином обчислене b_c та фіксоване a для симуляції цензованих даних. В коді для цього використовується макрозмінна `wb_cnsr_time`. Ключова логіка цих кроків показана нижче:

```
%*Derive macro variables based on distribution parameters for the simulation of
censoring time;
%let exp_hr_cnsr = (&cnsr_prop./(1 - &cnsr_prop.))*&exp_h_hr.;
%let wb_cnsr_time = &wb_b.*((1-&cnsr_prop.)/&cnsr_prop.)**(1/(&wb_a.));
```

```

%**create randomized event time and censor status for specified distribution;
%if %sysfunc(upcase(&data_distr.)) = "EXPONENTIAL" %then %do;
    evnt_t = rand("EXPONENTIAL", 1/(&exp_h_hr.));
    cnsr_t = rand("EXPONENTIAL", 1/(&exp_hr_cnsr.));
%end;

%else %if %sysfunc(upcase(&data_distr.)) = "WEIBULL" %then %do;
    evnt_t = rand("WEIBULL", &wb_a., &wb_b.);
    cnsr_t = rand("WEIBULL", &wb_a., &wb_cnsr_time.); ;
%end;

```

Треба відзначити, що фактична пропорція цензорованих даних, отримана в результаті таких обчислень може бути не точно рівною значенню, вказаному користувачем, оскільки вона залежить від розміру вибірки. У таблиці нижче наведено приклад таких обчислень, який ілюструє, що фактична пропорція наближається до теоретичної коли об'єм вибірки росте.

Розподіл	Розмір вибірки	Вказана пропорція цензорованих даних	Фактична пропорція цензорованих даних
Експоненційний	100	0.2	0.1976
Експоненційний	200	0.2	0.1982
Експоненційний	500	0.2	0.2014
Експоненційний	800	0.2	0.1992
Вейбула	100	0.2	0.2060
Вейбула	200	0.2	0.2028
Вейбула	500	0.2	0.2016
Вейбула	800	0.2	0.2016

Таблиця 4. Зведена статистика по вказаній і фактичній пропорції цензорованих даних про виживання з різними об'ємами вибірок, при 1000 симуляціях.

Далі, спираючись на симульований час настання подій та цензорингу, наша програма створює дані з часом виживання та статусом цензорингу. Як видно з фрагменту коду, наведеного нижче, якщо час цензорингу менше за час настання

події, таке спостереження вважається цензорованим, і час виживання дорівнює часу цензорингу. Інакше час виживання дорівнює часу до настання події, і цензорингу не відбувається. Слід зауважити, що статус цензорингу і час виживання обчислені таким чином застосовуються для контрольної групи, але є лише базовими значеннями для досліджуваної групи. Час виживання і статус цензорингу для останньої буде змінюватись далі згідно з вказаним користувачами типом зміни функції ризику, що буде описано в наступному розділі.

```
if cnsr_t lt evnt_t then do;  
    cnsr = 1;  
    surv_t = cnsr_t;  
end;  
else do;  
    cnsr = 0;  
    surv_t = evnt_t;  
end ;
```

3.1.6. Вигляд функції ризику для підгруп

Це найважливіша частина нашої програми, яка забезпечує виконання припущення про збагачення наших підгруп. Ми передбачили чотири можливих вигляди для функції ризику: спадна, спадна, вігнута, та опукла.

Спадна функція ризику для дослідної групи схожа на функцію ризику для контрольної групи, але трохи менше за неї, і вона ще зменшується з часом. Зростаюча функція ризику для дослідної групи суттєво менша за ризик для контрольної групи на початку дослідження, і зростає з часом; вона все ще менша за функцію ризику контрольної групи навіть у кінці дослідження. Вігнута функція ризику схожа на таку для першої групи у середній частині часового проміжку, і помітно менше на початку і в кінці дослідження. Опукла функція означає, що ризик для другої групи буде менше в середині дослідження, і приблизно однаковий на його початку і в кінці.

Ми вважаємо, що в цій роботі було дуже важливо передбачити випадок, коли рівень ризику для двох підгруп другої групи неоднаково змінюється з часом. Користувач може вказати типи змін для двох підгруп за допомогою двох

макрозмінних — subgrp1_type та subgrp2_type. Код нижче демонструє як макрос обробляє такі вхідні дані.

```
%if %sysfunc(lowcase(&sgrp1_hf.)) = "DECR" %then %do;
...
%end;
%else %if %sysfunc(lowcase(&sgrp1_hf.)) = "INCR" %then %do;
...
%end;
%else %if %sysfunc(lowcase(&sgrp1_hf.)) = "CONC" %then %do;
...
%end;
%else %if %sysfunc(lowcase(&sgrp1_hf.)) = "CONV" %then %do;
...
%end;
```

Загалом передбачено шість можливих комбінацій для функцій ризику двох підгруп дослідної групи:

Комбінація	Підгрупа 1	Підгрупа 2
1	Зростаюча	Спадна
2	Зростаюча	Увігнута
3	Зростаюча	Опукла
4	Спадна	Увігнута
5	Спадна	Опукла
6	Увігнута	Опукла

Таблиця 5. Можливі комбінації для функцій ризику підгруп дослідної групи.

Наша ціль досягається за допомогою наведеного нижче алгоритму.

Етап 1: отримати час подій для 20-го, 40-го, 60-го та 80-го перцентілей для кожної підгрупи другої групи для кожної симуляції. Ключовий фрагмент коду цього етапу:

```
proc means data = sim_ds p20 p40 p60 p80;
  where arm = 2;
  class sim_i subgroup;
  var evnt_t;
run;
```

Етап 2. Тепер кожна підгрупа другої групи розбита на п'ять частин, обмежених порохованими вище перцентілями та мінімумом і максимумом. Кожне

симульоване спостереження з часом виживання, як цензорованим, так і ні, буде включено до однієї з таких частин.

min	p20	p40	p60	p80	max
Частина 1	Частина 2	Частина 3	Частина 4	Частина 5	

Ілюстрація 1. Розбивка підгруп дослідної групи на частини.

Етап 3. Для спадної функції ризику нам потрібно було досягти плавної зміни з першої частини до п'ятої.

Етап 3.1. Для експоненціальної функції розподілу зі спадною функцією ризику: згідно з Таблицею 2, наша функція ризику — це константа, обернена до очікуваної тривалості виживання. В нашій програмі ми надаємо користувачу два параметри — `exp_h_hr` та `exp_l_hr`, за допомогою яких можна задати найбільше та найменше значення ризику.

Для кожної частини даних у підгрупі зі спадною функцією ризику оригінальний та модифікований час до настання події пов'язані наступним відношенням:

$$\frac{E_u}{E_s} = \frac{h_s}{h_u} = \frac{h_s}{h_s(h_s - h_e) \frac{i}{5}} \quad (1)$$

де E_u та E_s — це модифікований та оригінальний час настання події; h_s , h_u та h_e — рівні ризику на початковому моменті часу, оновлені для кожного моменту, та на останньому моменті часу; $i = 1, 2, 3, 4, 5$ для кожної частини підгрупи. З цього рівняння ми можемо виразити

$$E_u = E_s \frac{h_s}{h_s(h_s - h_e) \frac{i}{5}} \quad (2)$$

Ми будемо оновлювати час події для кожної частини на рівні пацієнтів, тобто за цією формулою для кожного спостереження.

За такою ж логікою ми проводимо оновлення часу цензорингу, використовуючи аналогічні рівні ризику для цензорованих даних. І далі ми отримуємо остаточні час виживання і статус цензорингу для кожної підгрупи у

досліджуваній групі спираючись на оновлений час події та цензорингу, за тією ж логікою, що і у попередній секції для контрольної групи.

Етап 3.2. Для розподілу Вейбула зі спадною функцією ризику: згідно з Таблицею 2, рівень ризику становить $\frac{a}{b^a} x^{a-1}$, тобто є функцією часу виживання для кожного пацієнта. Позначимо оригінальний рівень ризику через h_s , найнижчий через h_e , а їх відношення через $HRR = \frac{h_e}{h_s}$, до якого користувач має доступ за допомогою параметра `wb_hr`. Позначимо також оновлені рівні ризику через h_u . Щоб отримати спадну функцію ризику, візьмемо

$$\frac{h_u}{h_s} = 1 - (1 - HRR) \frac{i}{5} = \frac{\frac{a}{b^a} E_u^{a-1}}{\frac{a}{b^a} E_s^{a-1}} = \left(\frac{E_u}{E_s} \right)^{a-1} \quad (3)$$

де E_s та E_u — оригінальний та модифікований час до настання події, $i = 1, 2, 3, 4, 5$. З цього виразу можна отримати

$$E_u = E_s \left(1 - (1 - HRR) \frac{i}{5} \right)^{\frac{1}{a-1}} \quad (4)$$

Останню формулу можна використовувати, щоб підрахувати модифікований час до настання події для кожного спостереження.

Використовуючи ті ж міркування, що і на етапі 3.1 можна отримати модифіковані цензоровані дані, та фінальний час виживання та статус цензорингу.

Етап 4. Для зростаючої функції ризику, ми почнемо з найменшого значення для частини 1, та будемо поступово її збільшувати для подальших частин.

Етап 4.1: обчислення рівня ризику для зростаючої функції у випадку експоненційного розподілу. Процес обчислення аналогічний тому, який ми проводили на етапі 3.1, однак для зростаючої функції ми замінимо формулу 1 на

$$\frac{E_u}{E_s} = \frac{h_s}{h_u} = \frac{h_s}{h_s - (h_s - h_e) \frac{(6-i)}{5}}$$

З цієї формули отримуємо

$$E_u = E_s \frac{h_s}{h_s - (h_s - h_e) \frac{(6-i)}{5}}$$

Аналогічно, ми обчислюємо значення цього виразу для кожного пацієнта щоб оновити час до настання події, і використовуємо таку саму логіку для визначення того, чи треба цензурувати цей оновлений час, та врешті-решт фіналізуємо цей час та статус цензорингу згідно з логікою, яку ми використовували для групи контролю.

Етап 4.2. Для зростаючої функції ризику при розподілі Вейбула: ми проводимо обчислення аналогічні тим, які ми проводили на Етапі 3.2, але заміняємо формулу 3 на таку, з якої можна отримати зростаючу функцію:

$$\frac{h_u}{h_s} = 1 - (1 - HRR) \frac{(6 - i)}{5} = \left(\frac{E_u}{E_s} \right)^{a-1}$$

з якої можна отримати

$$E_u = E_s \left(1 - (1 - HRR) \frac{(6 - i)}{5} \right)^{\frac{1}{a-1}}$$

Користуючись цією формулою, як і раніше, ми можемо оновити час до настання події для кожного пацієнта, і використати нашу логіку для цензорингу даних щоб отримати фінальний час та статус цензорингу.

Етап 5. Для функції ризику увігнутого типу ми прагнемо отримати значення, яку будуть менші за ризики для контрольної групи на початку та в кінці дослідження, та схожі на них в середині нашого часового проміжку.

Етап 5.1. Функція ризику увігнутого типу для даних, розподілених експоненціально — аналогічно етапу 3.1, ми оновлюємо формулу 1 так, щоб отримати функцію увігнутого типу:

$$\frac{E_u}{E_s} = \frac{h_s}{h_u} = \frac{h_s}{h_s - (h_s - h_e) \frac{|3 - i|}{2}}$$

звідки отримуємо

$$E_u = E_s \frac{h_s}{h_s - (h_s - h_e) \frac{|3 - i|}{2}}$$

Як і у попередніх етапах, ми використаємо цю формулу для кожного спостереження, отримавши модифікований час до настання подій, та ту саму логіку для отримання нового часу цензорингу та фінальних даних.

Етап 5.2. Функція ризику увігнутого типу для розподілу Вейбула: як і на етапі 3.2, ми обчислюємо формулу:

$$\frac{h_u}{h_s} = 1 - (1 - HRR) \frac{|3 - i|}{2} = \left(\frac{E_u}{E_s} \right)^{a-1}$$

звідки отримуємо

$$E_u = E_s \left(1 - (1 - HRR) \frac{|3 - i|}{2} \right)^{\frac{1}{a-1}}$$

та перераховуємо за допомогою неї час до настання події та статус цензорингу для другої групи.

Етап 6. Функція ризику опуклого типу: вона подібна до функції ризику для контрольної групи на початку і в кінці нашого часового проміжку, та менша всередині нього.

Етап 6.1: функція ризику опуклого типу для експоненційного розподілу. Як і на етапі 3.1, ми обчислюємо аналог формули 1, але для опуклої функції:

$$\frac{E_u}{E_s} = \frac{h_s}{h_u} = \frac{h_s}{h_s - (h_s - h_e) \frac{2 - |3 - i|}{2}}$$

звідки отримуємо

$$E_u = E_s \frac{h_s}{h_s - (h_s - h_e) \frac{2 - |3 - i|}{2}}$$

Аналогічно, ми знову обчислюємо значення цієї формули для кожного пацієнта щоб встановити модифікований час до настання події та статус цензорингу для групи 2.

Етап 6.2: функція розподілу опуклого типу для розподілу Вейбула. Як і на етапі 3.2, ми виводимо аналог формули 3:

$$\frac{h_u}{h_s} = 1 - (1 - HRR) \frac{3 - |3 - i|}{2} = \left(\frac{E_u}{E_s} \right)^{a-1}$$

звідки отримуємо

$$E_u = E_s \left(1 - (1 - HRR) \frac{3 - |3 - i|}{2} \right)^{\frac{1}{a-1}}$$

Аналогічно до попередніх пунктів, ми використовуємо цю формулу для оновлення часу до настання подій та статусу цензорингу для групи 2.

Наведений нижче SAS код демонструє імплементацію ключових моментів цього алгоритму для різних функцій ризику.

```
%if %sysfunc(upcase(&data_distr.)) = "EXPONENTIAL" %then %do;
  %if %sysfunc(lowcase(&sgrp1_hf.)) = "DECR" %then %do;
    evnt_t_2 = evnt_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-&exp_l_hr.) *
    time_perc_arm2/5);
    cnsr_t_2 = cnsr_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-&exp_l_hr.) *
    time_perc_arm2/5);
  %end;

  %else %if %sysfunc(lowcase(&sgrp1_hf.)) = "INCR" %then %do;
    evnt_t_2 = evnt_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
    &exp_l_hr.)*(6-time_perc_arm2)/5);
    cnsr_t_2 = cnsr_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
    &exp_l_hr.)*(6-time_perc_arm2)/5);
  %end;

  %else %if %sysfunc(lowcase(&sgrp1_hf.)) = "CONC" %then %do;
    evnt_t_2 = evnt_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
    &exp_l_hr.)*abs(3-time_perc_arm2)/2);
    cnsr_t_2 = cnsr_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
    &exp_l_hr.)*abs(3-time_perc_arm2)/2);
  %end;

  %else %if %sysfunc(lowcase(&sgrp1_hf.)) = "CONV" %then %do;
    evnt_t_2 = evnt_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
    &exp_l_hr.)*abs(2-abs(3-time_perc_arm2))/2);
    cnsr_t_2 = cnsr_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
    &exp_l_hr.)*abs(2-abs(3-time_perc_arm2))/2);
  %end;
%end;

%else %if %sysfunc(upcase(&data_distr.)) = "WEIBULL" %then %do;
  %if %sysfunc(lowcase(&sgrp1_hf.)) = "DECR" %then %do;
    evnt_t_2 = evnt_t * (1-(1-&wb_hr.)*time_perc_arm2/5)**(1/(&wb_a.-1));
    cnsr_t_2 = cnsr_t * (1-(1-&wb_hr.)*time_perc_arm2/5)**(1/(&wb_a.-1));
```

```

%end;

%else %if %sysfunc(lowcase(&sgrp1_hf.)) = "INCR" %then %do;
    evnt_t_2 = evnt_t * (1-(1-&wb_hr.)*(6-time_perc_arm2)/5)**(1/(&wb_a.-1));
    cnsr_t_2 = cnsr_t * (1-(1-&wb_hr.)*(6-time_perc_arm2)/5)**(1/(&wb_a.-1));

%end;

%else %if %sysfunc(lowcase(&sgrp1_hf.)) = "CONC" %then %do;
    evnt_t_2 = evnt_t * (1-(1-&wb_hr.)*abs(time_perc_arm2-3)/2)**(1/(&wb_a.-1)) ;
    cnsr_t_2 = cnsr_t * (1-(1-&wb_hr.)*abs(time_perc_arm2-3)/2)**(1/(&wb_a.-1));
%end;

%else %if %sysfunc(lowcase(&sgrp1_hf.)) = "CONV" %then %do;
    evnt_t_2 = evnt_t * (1-(1-&wb_hr.)*abs(2-abs(time_perc_arm2-3))/2)**(1/(&wb_a.-1)) ;
    cnsr_t_2 = cnsr_t * (1-(1-&wb_hr.)*abs(2-abs(time_perc_arm2-3))/2)**(1/(&wb_a.-1)) ;
%end;
%end;

```

3.1.7. Лог-ранг тест та вивід результатів

У другій частині нашого макросу ми застосовуємо лог-ранг тест до кожного симульованого датасету. Ця імплементація є стандартною для аналізу даних виживання у SAS, оскільки типові вагові функції з Таблиці 1 реалізовані у стандартній процедурі `proc lifetest`. Користувач може передати в неї одне або декілька ключових слів, які визначають вагові функції за допомогою макрозмінної `weight`. Таблиця 6 наводить допустимі значення цих параметрів `proc lifetest` у SAS.

Ми також припускаємо що значення одиниці індикатору цензорингу свідчить про цензоровану подію, а нуль — про подію, яка відбулася. Це стандартний підхід у аналізі виживання.

```

ods output HomTests = logrank_res_pre;
proc lifetest data = save_loc.&sim_ds.;

```

```

by sim_i;
time surv_t * cnsr (1);
strata /test =&weight. group = arm;
run;

data logrank_pwr;
set logrank_res_pre;
if ProbChiSq le 0.05 then diff_found = 1;
else if ProbChiSq gt 0.05 then diff_found = 0;
run;

```

Вагова функція	Опція SAS
Усі вагові функції	ALL; при цьому Флемінга-Гаррінгтона буде вказана як FLEMING (1, 0)
Лог-ранг	LOGRANK
Гехана (Gehan)	WILCOXON
Тарона-Варе	TARONE
Пето-Пето	PETO
Модифікована Пето-Пето	MODPETO
Флемінга-Гаррінгтона	FLEMING (NUM1, NUM2)

Таблиця 6. Ключові слова у твердженні WEIGHT в SAS, які відповідають ваговим функціям з Таблиці 1.

Далі результати цього кроку передаються у `proc freq` для підрахунку емпіричної потужності тесту. `Proc Sgplot` створює гістограму з результатами цих підрахунків. Користувач може задати ім'я вихідного файлу за допомогою макрозмінної `report_name`, та шлях до нього за допомогою макрозмінної `path`. Ключові операції у цій частині наведено нижче:

```

ods output CrossTabFreqs = res_ds_pre;
proc freq data = logrank_pwr;
table Test * diff_found/nocol nopercnt;
run;
options orientation=portrait;
ods pdf file = "&save_loc./&plot_file..pdf" startpage = never;
title "Empirical Power of Log-Rank Test with Various Weight Functions";
Proc print data = save_loc.&power_ds (drop = Frequency diff_found) noobs label;
run;
ods graphics/ noborder;

```

```
proc sgplot data = save_loc.&power_ds;
    vbar Test/response = pct barwidth = 0.25;
    label Test = "Weight Function";
    yaxis max = 1;
run;
ods pdf close;
```

3.2 Програмні характеристики продукту

Спираючись на особистий досвід, автори роботи можуть заявити, що в індустрії обробки і аналізу результатів клінічних досліджень все ще часто використовується модель життєвого циклу «водоспад». Це пов'язано з тим, що форма більшості результатів, які необхідно передати у регулюючі організації, визначена наперед ще у протоколі клінічного дослідження, який створюється ще до моменту, коли на дослідження запрошують пацієнтів [14]. Ці результати також часто є стандартними для схожих досліджень, а, значить, відомими на досить ранніх етапах розробки. Нарешті, самі програми для аналізу даних зазвичай є невеликими (максимум кілька тисяч рядків коду), що полегшує їх модифікацію при необхідності. Дана програма не була виключенням, і також використовувала цю модель при розробці, відповідаючи усім вищезазначеним критеріям.

З огляду на невеликий об'єм програм, їх часто пише один програміст. З метою забезпечення максимального рівня якості, що має практичний сенс, зазвичай програми, які відповідають за критично важливі частини аналізу перевіряють шляхом написання аналогічних програм програмісти, які не користуються кодом вихідної програми. Тож те, що дана програма написана лише одним автором, і була протестована ним же в цілому відповідає прийнятим в індустрії підходам розробки.

Дана програма розроблялась у парадигмі монолітної архітектури, зважаючи на її порівняно невелику складність і обмежену функціональність. Система приймає набір параметрів зі значеннями з наперед заданої множини, і провидить над ними одну і ту саму обробку даних.

Швидкодія програми є достатньою для практичного застосування. Під час тестування програма генерувала результати тестувань 100 000 симуляцій для дослідження з 500 пацієнтами в середньому близько 190 секунд. Для практичного

застосування цілком достатньо навіть тисячі симуляцій, які програма генерує за три-п'ять секунд.

Для розробки програми використовувалась де-факто стандартна мова програмування у галузі статистичного аналізу клінічних досліджень SAS [15].

SAS (Statistical Analysis System) – мова спеціально розроблена для аналізу даних та статистичних обчислень, що використовується в наукових дослідженнях, медичній діагностиці, фінансовому аналізі та інших областях. SAS має широкий набір функцій для обробки даних, включаючи імпорт, фільтрацію, сортування, агрегацію та аналіз. Одним з основних переваг SAS є його ефективність у роботі з великими обсягами даних, що дозволяє швидко та ефективно проводити складні аналітичні операції.

Мова SAS навіть в базовій конфігурації включає в себе різноманітні вбудовані функції та процедури для роботи з даними, що спрощує процес аналізу та забезпечує точні та надійні результати. Крім того, SAS підтримує роботу з різними типами даних, включаючи числові, текстові, дати та інші, що робить його універсальним інструментом для різноманітних завдань обробки даних і статистичного аналізу.

Програмний продукт для даного дослідження був реалізований як макрос. Макроси в SAS - це потужний інструмент для автоматизації та стандартизації програмного коду. Вони дозволяють створювати параметризовані фрагменти коду, які можна використовувати повторно у різних частинах програми або навіть у різних програмах. Програму у вигляді макросу можна вважати плагіном у мові програмування SAS, оскільки її можливо запустити без додаткових маніпуляцій у будь-якому середовищі SAS, яке відповідає системним вимогам.

Слід відзначити, що хоча SAS і є потужним інструментом та стандартом в індустрії аналізу даних та статистичних обчислень клінічних досліджень, він має деякі недоліки. Одним з найбільш відомих недоліків є високі витрати на ліцензії та обслуговування програмного забезпечення SAS, що може бути проблемою для малих компаній або неприбуткових організацій. Крім того, в порівнянні з іншими сучасними інструментами для аналізу даних (R/Python), SAS може мати більш

важку для вивчення криву навчання, оскільки має власну специфіку синтаксису та функціональності. Також варто відзначити, що у порівнянні з деякими відкритими або безкоштовними альтернативами, SAS може бути менш гнучким у термінах розширення та адаптації до специфічних потреб користувача. Для конкуренції з відкритими та безкоштовними альтернативами SAS запустили безкоштовні версії для студентів та науковців, які використовують їх з неприбутковими цілями — зараз це SAS OnDemand for Academics [16].

3.3 Приклади обчислювальних експериментів та аналіз результатів

У цьому розділі ми продемонструємо приклади двох гіпотетичних досліджень, докладно опишемо всі параметри макроса та результати його роботи.

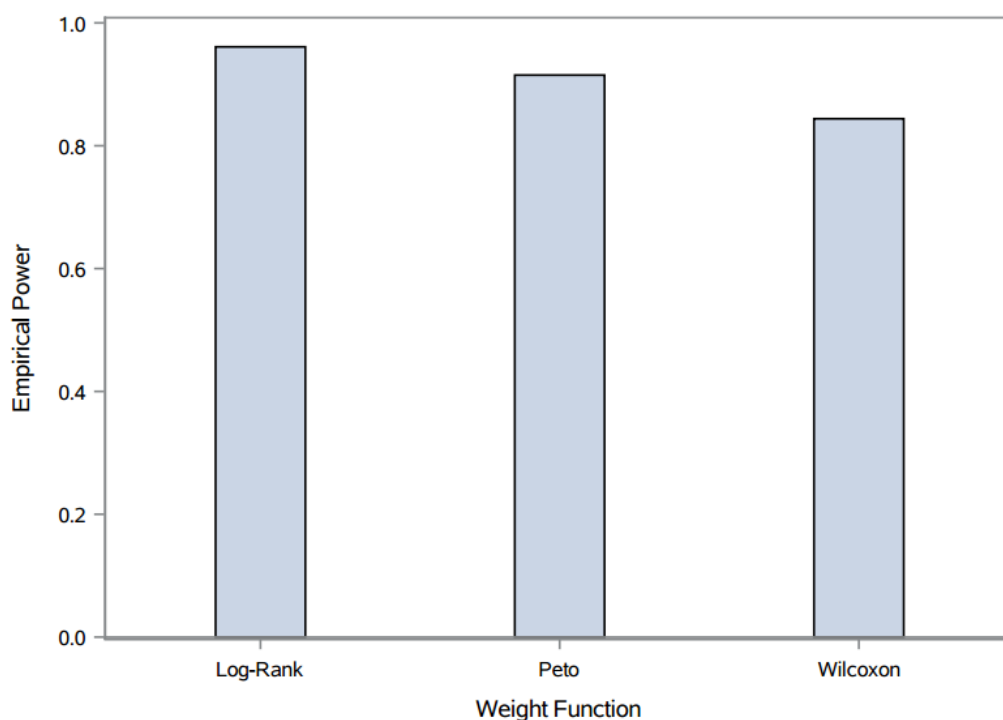
Дизайн дослідження	Виклик макросу
	%logrank_weight_sim
Засів для функції, яка генерує випадкові дані	(seed_rand = 2019,
Кількість симуляцій	simulation_n = 1000,
Розмір вибірки, кількість пацієнтів у дослідженні	study_n = 200,
Пропорція підгруп у групі	sgrp1_prop = 0.3,
Розподіл, експоненційний/Вейбула	data_distr = "Exponential",
Найвище значення ризику для експоненційного розподілу	exp_h_hr = 6,
Найнижче значення ризику для експоненційного розподілу	exp_l_hr 1 = 1,
Пропорція пацієнтів, які будуть зацензовані	cnsr_prop = 0.5,
Тип функції ризику для першої підгрупи	sgrp1_hf = "INC",

Тип функції ризику для другої підгрупи	sgrp2_hf = "DECR",
Шлях для збереження датасетів з результатами та графіків	save_loc = %str(root/home/work_losy/output),
Назва датасету з вхідними даними для тестування	sim_ds = exp_2019_7,
Вагові функції для лог-ранг тесту	weight = LOGRANK PETO WILCOXON,
Назва датасету з вихідними даними про потужність тестів	power_smry_ds = exp_smry_4,
Назва файлу для графіка з результатами	plot_file = exp_2019_report_4);

Таблиця 7. Перший приклад виклику макроса.

Summary of empirical power profile for log-rank test with different weight functions

Test	Empirical Power	Sample Size	Simulation Number	Subgroup Ratio	Censor Proportion	Subgroup 1 Hazard Rate Change type	Subgroup 2 Hazard Rate Change type	Survival Data Distribution Assumption	Distribution Parameter Information
Log-Rank	0.96	200	1000	0.3 vs. 0.7	0.5	Hazard Rate Increase	Hazard Rate Decrease	Exponential	Hazard rate at highest: 6. Hazard rate at lowest: 1
Peto	0.92	200	1000	0.3 vs. 0.7	0.5	Hazard Rate Increase	Hazard Rate Decrease	Exponential	Hazard rate at highest: 6. Hazard rate at lowest: 1
Wilcoxon	0.84	200	1000	0.3 vs. 0.7	0.5	Hazard Rate Increase	Hazard Rate Decrease	Exponential	Hazard rate at highest: 6. Hazard rate at lowest: 1

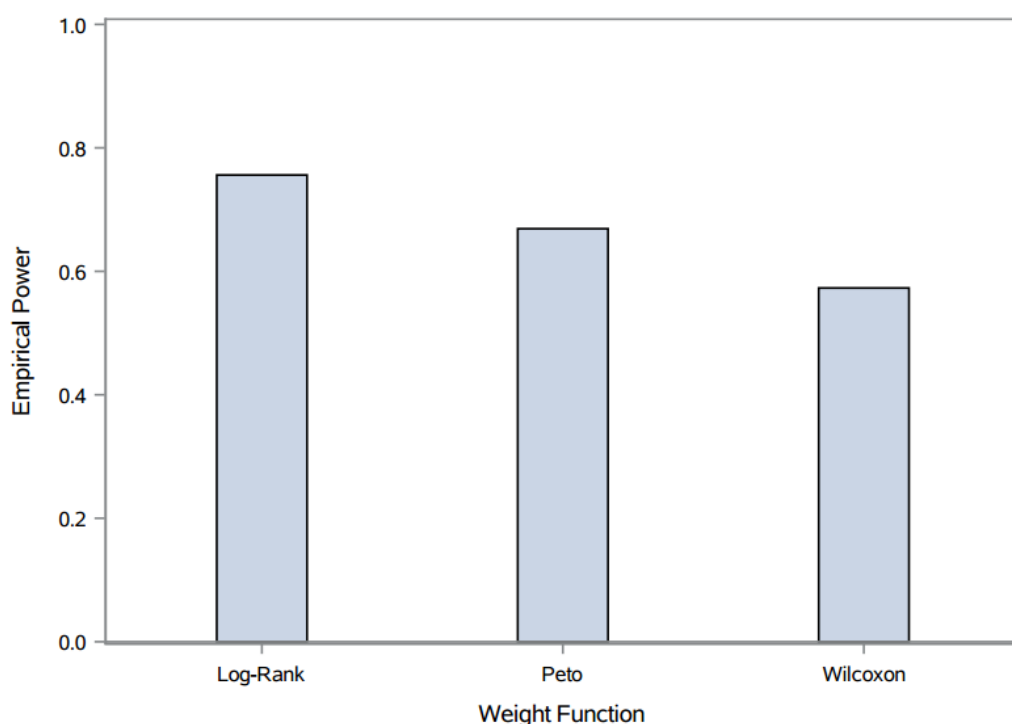


Ілюстрація 2. Вивід програми після виклику макроса з Таблиці 7.

В даному прикладі ми використали експоненційний розподіл зі зростаючою функцією ризику для 30% пацієнтів які складали першу підгрупу дослідної групи, та спадною для 70% пацієнтів, які складали другу підгрупу дослідної групи. Результати показують значиму, хоч і не велику, перевагу класичного лог-ранг тесту над варіантами Пето та Вілкоксона у цих умовах. Ці результати залишаються стабільними при кількості симуляцій приблизно більше п'ятдесяти. Зменшення ж розміру вибірки веде до помітного просідання в потужності при $N < 100$. Оскільки вибір тесту, найбільш підходящого під припущення щодо функції ризиків у клінічних дослідженнях буде, швидше за все, робитись саме з ціллю мінімізації кількості пацієнтів у дослідженні, цей параметр є ключовим для порівняння продуктивності тестів.

Summary of empirical power profile for log-rank test with different weight functions

Test	Empirical Power	Sample Size	Simulation Number	Subgroup Ratio	Censor Proportion	Subgroup 1 Hazard Rate Change type	Subgroup 2 Hazard Rate Change type	Survival Data Distribution Assumption	Distribution Parameter Information
Log-Rank	0.76	100	1000	0.3 vs. 0.7	0.5	Hazard Rate Increase	Hazard Rate Decrease	Exponential	Hazard rate at highest: 6. Hazard rate at lowest: 1
Peto	0.67	100	1000	0.3 vs. 0.7	0.5	Hazard Rate Increase	Hazard Rate Decrease	Exponential	Hazard rate at highest: 6. Hazard rate at lowest: 1
Wilcoxon	0.57	100	1000	0.3 vs. 0.7	0.5	Hazard Rate Increase	Hazard Rate Decrease	Exponential	Hazard rate at highest: 6. Hazard rate at lowest: 1



Ілюстрація 3. Вивід програми після виклику макроса з Таблиці 7, з розміром вибірки 100.

Розглянемо тепер експеримент з параметрами, в яких проявляється перевага альтернативних вагових функцій над стандартним лог-ранг тестом.

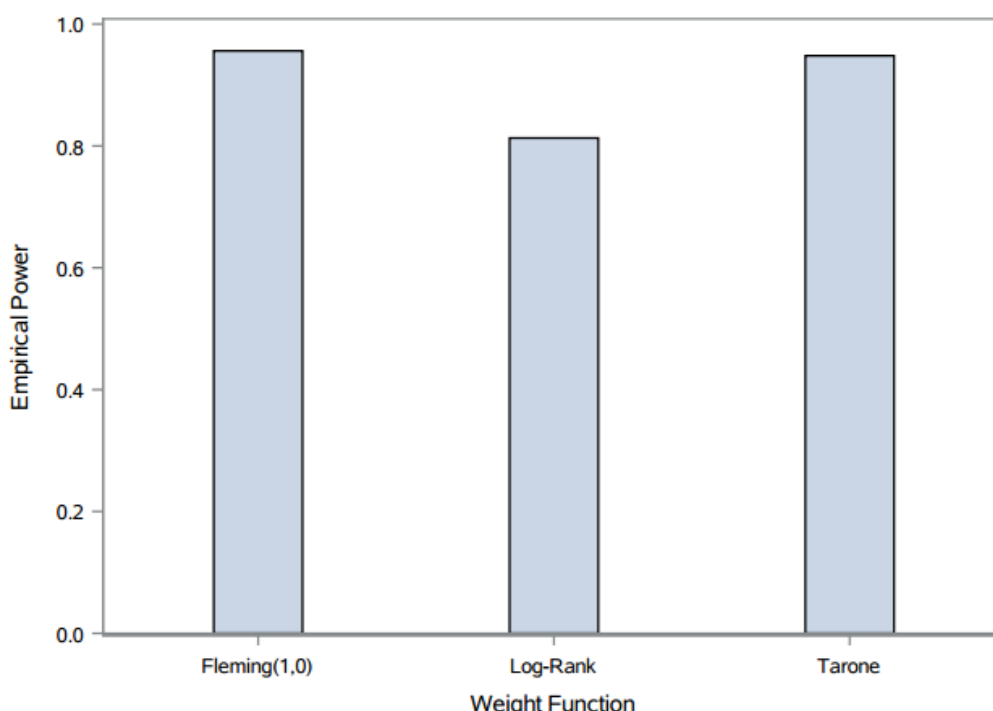
Дизайн дослідження	Виклик макросу
	%logrank_weight_sim
Засів для функції, яка генерує випадкові дані	(seed_rand = 20200101,
Кількість симуляцій	simulation_n = 1000,
Розмір вибірки, кількість пацієнтів у дослідженні	study_n = 200,
Пропорція підгруп у групі	sgrp1_prop = 0.7,
Розподіл, експоненційний/Вейбула	data_distr = "Weibull",
Параметр a у розподілі Вейбула, $a > 0$, при $a = 1$ розподіл стає експоненційним	wb_a = 2,
Параметр b у розподілі Вейбула, $b > 0$	wb_b = 2,
Відношення ризиків, найвищого до найнижчого у розподілі Вейбула	wb_hr = 0.45,
Пропорція пацієнтів, які будуть зацензовані	cnsr_prop = 0.2,
Тип функції ризику для першої підгрупи	sgrp1_hf = "CONV",
Тип функції ризику для другої підгрупи	sgrp2_hf = "CONC",
Шлях для збереження датасетів з результатами та графіків	save_loc = %str(root/home/work_losy/output),
Назва датасету з вхідними даними для тестування	sim_ds = wb_20200101_6,

Вагові функції для лог-ранг тесту	weight = LOGRANK FLEMING (1,0) TARONE,
Назва датасету з вихідними даними про потужність тестів	power_smry_ds = wb_smry_6,
Назва файлу для графіка з результатами	plot_file = wb_20200101_report_6);

Таблиця 8. Другий приклад виклику макроса.

Summary of empirical power profile for log-rank test with different weight functions

Test	Empirical Power	Sample Size	Simulation Number	Subgroup Ratio	Censor Proportion	Subgroup 1 Hazard Rate Change type	Subgroup 2 Hazard Rate Change type	Survival Data Distribution Assumption	Distribution Parameter Information
Fleming(1,0)	0.96	200	1000	0.7 vs. 0.3	0.2	Hazard Rate Convex	Hazard Rate Concave	Weibull	Weibull parameter a: 2, Weibull parameter b: 2, the ratio of the hazard rate at lowest vs. highest is 0.45
Log-Rank	0.81	200	1000	0.7 vs. 0.3	0.2	Hazard Rate Convex	Hazard Rate Concave	Weibull	Weibull parameter a: 2, Weibull parameter b: 2, the ratio of the hazard rate at lowest vs. highest is 0.45
Tarone	0.95	200	1000	0.7 vs. 0.3	0.2	Hazard Rate Convex	Hazard Rate Concave	Weibull	Weibull parameter a: 2, Weibull parameter b: 2, the ratio of the hazard rate at lowest vs. highest is 0.45



Ілюстрація 4. Результат виводу програми з параметрами з Таблиці 8.

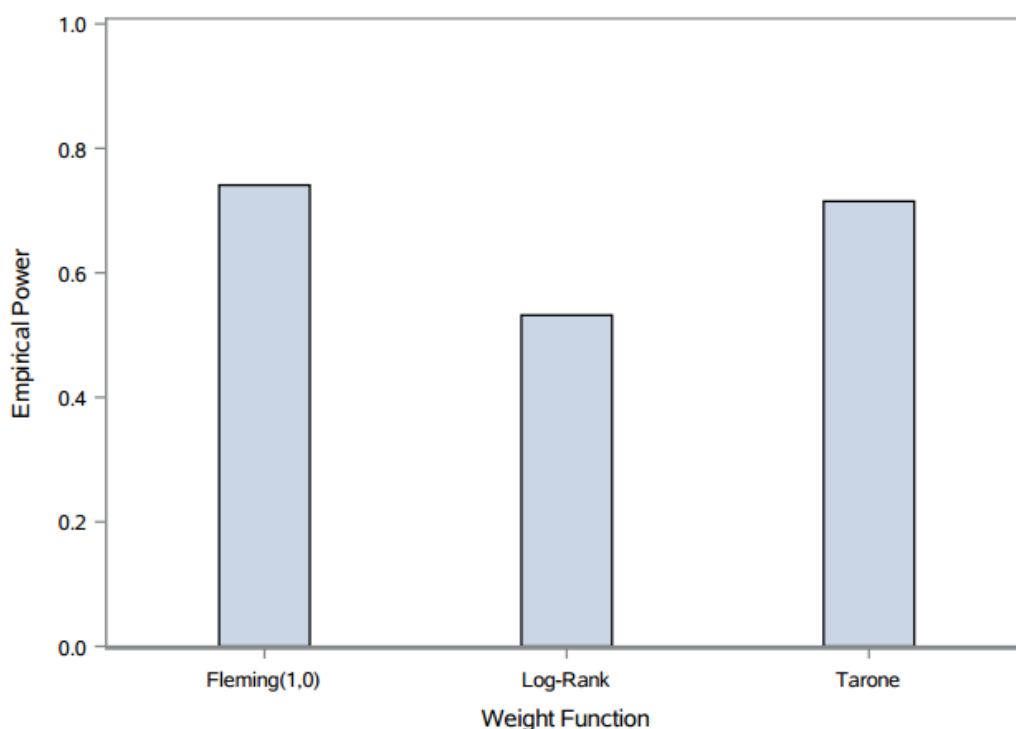
В даному прикладі ми припустили, що наші дані розподілені за Вейбулом (2, 2), з опуклою функцією ризику для першої підгрупи дослідної групи (70% пацієнтів), та увігнутою для другої підгрупи пацієнтів (30%). Ми також зробили

припущення, що лише 20% пацієнтів будуть цензоровані, тобто в даних буде більше подій, ніж у першому прикладі.

При таких припущеннях тест з ваговими функціями Флемінга та Тароне показує себе помітно краще за стандартний лог-ранг. Як і в попередньому прикладі, результати дуже чутливі до розміру вибірки, хоч тести і зберігають свої позиції один відносно одного.

Summary of empirical power profile for log-rank test with different weight functions

Test	Empirical Power	Sample Size	Simulation Number	Subgroup Ratio	Censor Proportion	Subgroup 1 Hazard Rate Change type	Subgroup 2 Hazard Rate Change type	Survival Data Distribution Assumption	Distribution Parameter Information
Fleming(1,0)	0.74	100	1000	0.7 vs. 0.3	0.2	Hazard Rate Convex	Hazard Rate Concave	Weibull	Weibull parameter a: 2, Weibull parameter b: 2. the ratio of the hazard rate at lowest vs. highest is 0.45
Log-Rank	0.53	100	1000	0.7 vs. 0.3	0.2	Hazard Rate Convex	Hazard Rate Concave	Weibull	Weibull parameter a: 2, Weibull parameter b: 2. the ratio of the hazard rate at lowest vs. highest is 0.45
Tarone	0.72	100	1000	0.7 vs. 0.3	0.2	Hazard Rate Convex	Hazard Rate Concave	Weibull	Weibull parameter a: 2, Weibull parameter b: 2. the ratio of the hazard rate at lowest vs. highest is 0.45



Ілюстрація 5. Результат виводу програми з параметрами з Таблиці 8 при розміром вибірки 100.

ВИСНОВКИ

В результаті проведеної роботи була досягнута поставлена ціль. Автори проаналізували класичні підходи до порівняння кривих виживання кількох груп пацієнтів у літературі та в індустрії. Була розроблена програма для аналізу ефективності лог-ранг тесту для даних з різними функціями ризиків у підгрупах дослідної групи та різними ваговими функціями. Були проведені обчислювальні експерименти, які продемонстрували різну ефективність використання вагових функцій в залежності від параметрів дослідження, що доводить можливість використовувати створену програму для оцінки та вибору конкретного варіанта лог-ранг тесту у клінічних дослідженнях.

Також були ідентифіковані потенційні перспективи розвитку досліджень на тему аналізу ефективності статистичних тестів порівняння функцій виживання груп пацієнтів. Створена програма потребує модернізації для моделювання досліджень з більш, ніж двома групами пацієнтів; з іншими розподілами даних; з довільними функціями ризику для досліджених підгруп, або з неконстантною функцією ризику для контрольної групи.

Таким чином, проведена робота дає можливість проводити моделювання та приймати рішення про параметри певного класу клінічних досліджень та статистичних методів їх аналізу, та може бути розширена для збільшення цього класу або охоплення інших статистичних методів аналізу клінічних досліджень.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. «NIH Clinical Research Trials and You: The Basics». US National Institutes of Health. URL: <https://www.nih.gov/health-information/nih-clinical-research-trials-you/basics>.
2. Sackett D (2006). «Clinical Epidemiology: How to Do Clinical Practice Research». Lippincott Williams & Wilkins. ISBN 978-0-7817-4524-6.
3. «The Drug Development Process. Step 3: Clinical trials». US Food and Drug Administration. URL: <https://www.fda.gov/patients/drug-development-process/step-3-clinical-research>
4. Dimasi, Joseph A; Grabowski, Henry G; Hansen, Ronald W (2016). «Innovation in the pharmaceutical industry: New estimates of R&D costs». Journal of Health Economics. 47: p 20–33.
5. «The Solution to Drug Prices». The New York Times. URL: <https://www.nytimes.com/2015/09/09/opinion/the-solution-to-drug-prices.html>
6. «Enrichment Strategies for Clinical TrialsClinical Trials». US Food and Drug Administration. URL: <https://www.fda.gov/files/drugs/published/Enrichment-Strategies-for-Clinical-Trials-%28PDF-%E2%80%93933KB%29.pdf>
7. Simon N; Simon R. «Adaptive enrichment designs for clinical trials». Biostatistics, Volume 14, Issue 4, September 2013, p 613–625.
8. Thall P (2021). «Adaptive enrichment designs in clinical trials». Annual Review of Statistics and Its Application, 8, p. 393-411.
9. Miller, Rupert G. Survival analysis, John Wiley & Sons, 1997. ISBN 0-471-25218-2.
10. Brookmeyer, R. and Crowley, J. (1982), «A Confidence Interval for the Median Survival Time». Biometrics 38, 29–41.
11. Cox, D. R. and Oakes, D. (1984), «Analysis of Survival Data». London: Chapman & Hall.

12. Klein, J. P. and Moeschberger, M. L. (1997). «Survival Analysis: Techniques for Censored and Truncated Data». New York: Springer-Verlag.
13. «SAS/STAT® 12.1 User's Guide: The LIFETEST Procedure». The SAS Institute. URL:
<https://support.sas.com/documentation/onlinedoc/stat/121/lifetest.pdf>
14. Cipriani A, Barbui C. «What is a clinical trial protocol?». Epidemiol Psychiatr Soc. 2010 Apr-Jun;19(2), p. 116-7.
15. Dembe, A.; Partridge, J.; Geist, L. (2011). «Statistical software applications used in health services research: Analysis of published studies in the U.S"». BMC Health Services Research. 11: p 252.
16. «SAS Academic Software». The SAS Institute. URL:
https://www.sas.com/en_us/learn/academic-programs/software.html

ДОДАТКИ

Додаток 1. Код програми на мові SAS

```
%macro logrank_weight_sim (
    seed_rand      = ,      /*The random seed, an integer value for SAS RANDOM
function. Can be used to obtain repeatable results */
    simulation_n    = ,      /*Number of simulations, integer*/
    study_n         = ,      /*Number of patients on the study, integer*/
    sgrp1_prop      = ,      /*The proportion of the first subgroup in the treatment
arm, a decimal value between 0 and 1 (e.g., 0.5).*/
    data_distr      = ,      /*The desired distribution of the data, can be
"EXPONENTIAL" or "WEIBULL" with quotes*/
    exp_h_hr        = ,      /*When data_distr = "EXPONENTIAL": the highest hazard
rate over the time of the study, a positive real value*/
    exp_l_hr        = ,      /*When data_distr = "EXPONENTIAL": the lowest hazard
rate over the time of the study, a positive real value less than exp_h_hr */
    wb_a            = ,      /*When data_distr = "WEIBULL": parameter a value, a
positive real number. Note that if wb_a = 1, the distribution becomes
exponential*/
    wb_b            = ,      /*When data_distr = "WEIBULL": parameter b value, a
positive real number. */
    wb_hr           = ,      /*When data_distr = "WEIBULL": the hazard rate ratio
(lowest to highest), a decimal value between 0 and 1 (e.g., 0.5)*/
    cnsr_prop       = ,      /*The proportion of censored patients, a decimal value
between 0 and 1 (e.g., 0.5).*/
    sgrp1_hf        = ,      /*The hazard function type for the first subgroup of
treatment arm, can be "INCR", "DECR", "CONC", or "CONV"*/
    sgrp2_hf        = ,      /*The hazard function type for the second subgroup of
treatment arm, can be "INCR", "DECR", "CONC", or "CONV" and should not be the same
as the function for the first subgroup*/
    save_loc        = ,      /*Location to save the resulting datasets and
output, e.g. "root/etc" without quotes*/
    sim_ds          = ,      /*Name of the dataset with the simulation data, will be
saved to save_loc*/
    weight          = ,      /*Log-rank weight function (passed into proc lifetest
WEIGHT statement), can be LOGRANK WILCOXON TARONE PETO MODPETO FLEMING (NUM1,
NUM2) and ALL (which includes FLEMING (1, 0)*/
    power_ds        = ,      /*Name of the dataset with test power calculation result,
will be saved to save_loc*/
```

```

        plot_file      = ,      /*Name of the file which contains table and plot with
results, will be saved into save_loc*/
    );

%*****
%**** Simulation of the survival data ****
%*****

%Derive macro variables based on distribution parameters for the simulation of
censoring time;
%let exp_hr_cnsr = (&cnsr_prop./(1 - &cnsr_prop.))*&exp_h_hr.;
%let wb_cnsr_time = &wb_b.*((1-&cnsr_prop.)/&cnsr_prop.)*(1/(&wb_a.));

%***delete duplicated dataset in work library in Interactive SAS;
*proc datasets library=work kill;
*run;
*quit;

data sim_ds (drop = nobs i);
    call streaminit(&seed_rand.);
        do sim_i = 1 to &simulation_n.;
            ID = 0;
            nobs = 0;
            do i = 1 to &study_n.;
                nobs + 1;
                ID + 1;
                subgroup = .;

                %**create randomized event time and censor status for
specified distribution;
                %if %sysfunc(upcase(&data_distr.)) = "EXPONENTIAL" %then %do;
                    evnt_t = rand("EXPONENTIAL", 1/(&exp_h_hr.));
                    cnsr_t = rand("EXPONENTIAL", 1/(&exp_hr_cnsr.));
                %end;

                %else %if %sysfunc(upcase(&data_distr.)) = "WEIBULL" %then %do;
                    evnt_t = rand("WEIBULL", &wb_a., &wb_b.);
                    cnsr_t = rand("WEIBULL", &wb_a., &wb_cnsr_time.); ;
                %end;

                %**update the time and censoring for placebo arm;
                if nobs <= %sysevalf(&study_n.)/2 then do;
                    arm =1;

```

```

        if cnsr_t lt evnt_t then do;
            cnsr = 1;
            surv_t = cnsr_t;
        end;
    else do;
        cnsr = 0;
        surv_t = evnt_t;
    end ;

    if      0 <= nobs <=
%sysevalf(&study_n.)*(0.5*%sysevalf(&sgrp1_prop.)) then subgroup = 1;
        else if %sysevalf(&study_n.)*(0.5*%sysevalf(&sgrp1_prop.)) <
nobs <= %sysevalf(&study_n.)*0.5 then subgroup = 2;

    end ;

    **update time and censoring for treatment arm, set it to
blank and rederive later, and create subgroup indicator;
    else if nobs > %sysevalf(&study_n.)/2 then do;
        arm = 2;
        surv_t = .;
        cnsr      = .;

        if %sysevalf(&study_n.)/2 < nobs <= %sysevalf(&study_n.)*(0.5 +
0.5*%sysevalf(&sgrp1_prop.)) then subgroup = 1;
        else if nobs > %sysevalf(&study_n.)*(0.5 +
0.5*%sysevalf(&sgrp1_prop.)) then subgroup = 2;

    end;

    output;
end;
end;
run;

ods listing close;
ods html close;

%*Get the percentiles of time for treatment arm;
ods output Summary = arm2_perc;
proc means data = sim_ds p20 p40 p60 p80;
    where arm = 2;

```

```

class sim_i subgroup;
var evnt_t;
run;

data arm2_perc_2;
set arm2_perc;
arm = 2;
drop NOBS;
run;

%**bring the event times from percentiles into the dataset;
proc sort data = arm2_perc_2;
by sim_i arm subgroup;
run;

proc sort data = sim_ds;
by sim_i arm subgroup;
run;

data prefinal;
merge arm2_perc_2 sim_ds;
by sim_i arm subgroup;
if arm = 2 then time_perc_arm2 = 1*(evnt_t lt evnt_t_P20)+
2*(evnt_t_P20 le evnt_t lt evnt_t_P40)+
3*(evnt_t_P40 le evnt_t lt evnt_t_P60)+
4*(evnt_t_P60 le evnt_t lt evnt_t_P80)+
5*(evnt_t ge evnt_t_P80);

run;

%**finally, rederive event time and censor status for treatmetn arm based on hazard
function type;
data final_surv_data;
set prefinal;

if arm = 2 and subgroup = 1 then do;

    %if %sysfunc(upcase(&data_distr.)) = "EXPONENTIAL" %then %do;

        %if %sysfunc(lowcase(&sgrp1_hf.)) = "DECR" %then %do;
            evnt_t_2 = evnt_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
&exp_l_hr.)*time_perc_arm2/5);

```

```

        cnsr_t_2 = cnsr_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
&exp_l_hr.)*time_perc_arm2/5);
        %end;

        %else %if %sysfunc(lowercase(&sgrp1_hf.)) = "INCR" %then %do;
            evnt_t_2 = evnt_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
&exp_l_hr.)*(6-time_perc_arm2)/5);
            cnsr_t_2 = cnsr_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
&exp_l_hr.)*(6-time_perc_arm2)/5);
            %end;

        %else %if %sysfunc(lowercase(&sgrp1_hf.)) = "CONC" %then %do;
            evnt_t_2 = evnt_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
&exp_l_hr.)*abs(3-time_perc_arm2)/2);
            cnsr_t_2 = cnsr_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
&exp_l_hr.)*abs(3-time_perc_arm2)/2);
            %end;

        %else %if %sysfunc(lowercase(&sgrp1_hf.)) = "CONV" %then %do;
            evnt_t_2 = evnt_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
&exp_l_hr.)*abs(2-abs(3-time_perc_arm2))/2);
            cnsr_t_2 = cnsr_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
&exp_l_hr.)*abs(2-abs(3-time_perc_arm2))/2);
            %end;
        %end;

        %else %if %sysfunc(uppercase(&data_distr.)) = "WEIBULL" %then %do;

            %if %sysfunc(lowercase(&sgrp1_hf.)) = "DECR" %then %do;
                evnt_t_2 = evnt_t * (1-(1-
&wb_hr.)*time_perc_arm2/5)**(1/(&wb_a.-1));
                cnsr_t_2 = cnsr_t * (1-(1-
&wb_hr.)*time_perc_arm2/5)**(1/(&wb_a.-1));
                %end;

            %else %if %sysfunc(lowercase(&sgrp1_hf.)) = "INCR" %then %do;
                evnt_t_2 = evnt_t * (1-(1-&wb_hr.)*(6-
time_perc_arm2)/5)**(1/(&wb_a.-1));
                cnsr_t_2 = cnsr_t * (1-(1-&wb_hr.)*(6-
time_perc_arm2)/5)**(1/(&wb_a.-1));

            %end;

```

```

        %else %if %sysfunc(lowcase(&sgrp1_hf.)) = "CONC" %then %do;
            evnt_t_2 = evnt_t * (1-(1-&wb_hr.)*abs(time_perc_arm2-
3)/2)**(1/(&wb_a.-1)) ;
            cnsr_t_2 = cnsr_t * (1-(1-&wb_hr.)*abs(time_perc_arm2-
3)/2)**(1/(&wb_a.-1));
        %end;

        %else %if %sysfunc(lowcase(&sgrp1_hf.)) = "CONV" %then %do;
            evnt_t_2 = evnt_t * (1-(1-&wb_hr.)*abs(2-abs(time_perc_arm2-
3))/2)**(1/(&wb_a.-1)) ;
            cnsr_t_2 = cnsr_t * (1-(1-&wb_hr.)*abs(2-abs(time_perc_arm2-
3))/2)**(1/(&wb_a.-1)) ;
        %end;

    %end;

end;

else if arm = 2 and subgroup = 2 then do;

    %if %sysfunc(upcase(&data_distr.)) = "EXPONENTIAL" %then %do;

        %if %sysfunc(lowcase(&sgrp2_hf.)) = "DECR" %then %do;
            evnt_t_2 = evnt_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
&exp_l_hr.)*time_perc_arm2/5);
            cnsr_t_2 = cnsr_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
&exp_l_hr.)*time_perc_arm2/5);
        %end;

        %else %if %sysfunc(lowcase(&sgrp2_hf.)) = "INCR" %then %do;
            evnt_t_2 = evnt_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
&exp_l_hr.)*(6-time_perc_arm2)/5);
            cnsr_t_2 = cnsr_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
&exp_l_hr.)*(6-time_perc_arm2)/5);
        %end;

        %else %if %sysfunc(lowcase(&sgrp2_hf.)) = "CONC" %then %do;
            evnt_t_2 = evnt_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
&exp_l_hr.)*abs(3-time_perc_arm2)/2);
            cnsr_t_2 = cnsr_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
&exp_l_hr.)*abs(3-time_perc_arm2)/2);

```

```

%end;

%else %if %sysfunc(lowercase(&sgrp2_hf.)) = "CONV" %then %do;
    evnt_t_2 = evnt_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
&exp_l_hr.)*abs(2-abs(3-time_perc_arm2))/2);
    cnsr_t_2 = cnsr_t * &exp_h_hr./(&exp_h_hr. - (&exp_h_hr.-
&exp_l_hr.)*abs(2-abs(3-time_perc_arm2))/2);
%end;

%end;

%else %if %sysfunc(uppercase(&data_distr.)) = "WEIBULL" %then %do;

%if %sysfunc(lowercase(&sgrp2_hf.)) = "DECR" %then %do;
    evnt_t_2 = evnt_t * (1-(1-
&wb_hr.)*time_perc_arm2/5)**(1/(&wb_a.-1));
    cnsr_t_2 = cnsr_t * (1-(1-
&wb_hr.)*time_perc_arm2/5)**(1/(&wb_a.-1));
%end;

%else %if %sysfunc(lowercase(&sgrp2_hf.)) = "INCR" %then %do;
    evnt_t_2 = evnt_t * (1-(1-&wb_hr.)*(6-
time_perc_arm2)/5)**(1/(&wb_a.-1));
    cnsr_t_2 = cnsr_t * (1-(1-&wb_hr.)*(6-
time_perc_arm2)/5)**(1/(&wb_a.-1));

%end;

%else %if %sysfunc(lowercase(&sgrp2_hf.)) = "CONC" %then %do;
    evnt_t_2 = evnt_t * (1-(1-&wb_hr.)*abs(time_perc_arm2-
3)/2)**(1/(&wb_a.-1)) ;
    cnsr_t_2 = cnsr_t * (1-(1-&wb_hr.)*abs(time_perc_arm2-
3)/2)**(1/(&wb_a.-1));
%end;

%else %if %sysfunc(lowercase(&sgrp2_hf.)) = "CONV" %then %do;
    evnt_t_2 = evnt_t * (1-(1-&wb_hr.)*abs(2-abs(time_perc_arm2-
3))/2)**(1/(&wb_a.-1)) ;
    cnsr_t_2 = cnsr_t * (1-(1-&wb_hr.)*abs(2-abs(time_perc_arm2-
3))/2)**(1/(&wb_a.-1)) ;

```

```

                                %end;

                                %end;

                                end;

                                if arm = 2 and subgroup in (1 2) then do;

                                    if cnsr_t_2 lt evnt_t_2 then do;
                                        cnsr = 1;
                                        surv_t = cnsr_t_2;
                                    end;

                                    else do;
                                        cnsr = 0;
                                        surv_t = evnt_t_2;
                                    end ;
                                end;

                                end;

run;

***SAS library to save results;
libname save_loc "&save_loc.";

data save_loc.&sim_ds.;
    set final_surv_data (drop = evnt_t_ : cnsr_t_2 time_perc_arm2);
run;

%*****;
%**** Simulation completed ****;
%*****;

%*****;
%**** Perform the log-rank test ****;
%*****;

ods output HomTests = logrank_res_pre;
proc lifetest data = save_loc.&sim_ds.;
    by sim_i;
    time surv_t * cnsr (1);
    strata /test =&weight. group = arm;
run;

```



```

data logrank_pwr;
  set logrank_res_pre;
  if ProbChiSq le 0.05 then diff_found = 1;
  else if ProbChiSq gt 0.05 then diff_found = 0;
run;

%*****;
%**** Log-rank test run ****;
%*****;

%*****;
%**** Create the resulting table and plot ****;
%*****;

ods output CrossTabFreqs = res_ds_pre;
proc freq data = logrank_pwr;
  table Test * diff_found/nocol nopercnt;
run;

data save_loc.&power_ds;
  set res_ds_pre(rename = (RowPercent = pct));
  where not missing(test) and diff_found = 1 ;
  pct = pct / 100;
  study_n = &study_n.;
  sim_n = &simulation_n.;
  sgrp_prop = "&sgrp1_prop. vs. %sysevalf(1-&sgrp1_prop.)";
  cnsr_prop = &cnsr_prop.;
  sgrp1_hf = "Hazard Rate "||&sgrp1_hf.;
  sgrp2_hf = "Hazard Rate "||&sgrp2_hf.;
  data_distr = &data_distr.;

  %if %sysfunc(upcase(&data_distr.)) = "EXPONENTIAL" %then %do;
    distr_param = "Highest hazard rate: &exp_h_hr.. Lowest hazard rate:
&exp_l_hr.";
  %end;

  %else %if %sysfunc(upcase(&data_distr.)) = "WEIBULL" %then %do;
    distr_param = "Weibull a: &wb_a., Weibull b: &wb_b.. Lowest to highest hazard
rate ratio is &wb_hr.";
  %end;

label

```

```

pct = "Empirical Power"
study_n = "Number of Patients"
sim_n = "Number of Simulations"
sgrp_prop = "Ratio of Subgroups"
cnsr_prop = "Proportion of Censored Events"
sgrp1_hf = "Hazard Rate Function Type for Subgroup 1"
sgrp2_hf = "Hazard Rate Function Type for Subgroup 2"
data_distr = "Distribution for Survival Data"
distr_param = "Parameters for Distribution";

drop Table _TYPE_ _TABLE_ Missing;

run;

options orientation=portrait;
ods pdf file = "&save_loc./&plot_file..pdf" startpage = never;
title "Empirical Power of Log-Rank Test with Various Weight Functions";
Proc print data = save_loc.&power_ds (drop = Frequency diff_found) noobs label;
run;

title " ";
ods graphics/ noborder;
proc sgplot data = save_loc.&power_ds;
    vbar Test/response = pct barwidth = 0.25;
    label Test = "Weight Function";
    yaxis max = 1;
run;
ods pdf close;

%mend;

****Macro invocation example 1:
• Simulation number: 1000
• Sample size: 200
• Two arms, and randomization ratio: 1 vs. 1
• Proportion of Subgroups: 0.3 vs. 0.7
• Survival data distribution assumption: Exponential
• Hazard rate highest in treatment period: 6
• Hazard rate lowest in treatment period: 1
• Censor Proportion: 0.5
• Subgroup 1 in arm 2 hazard rate changing type: Increase
• Subgroup 2 in arm 2 hazard rate changing type: Decrease

```

- Weight functions for log-rank test: LOGRANK PETO WILCOXON;

```
%logrank_weight_sim
(seed_rand = 2019,
simulation_n = 1000,
study_n = 200,
sgrp1_prop = 0.3,
data_distr = "EXPONENTIAL",
exp_h_hr = 6,
exp_l_hr = 1,
wb_a = ,
wb_b = ,
wb_hr = ,
cnsr_prop = 0.5,
sgrp1_hf = "INC",
sgrp2_hf = "DECR",
save_loc = %str(root/home/work/work_losy/output),
sim_ds = exp_2019_7,
weight = LOGRANK PETO WILCOXON,
power_smry_ds = exp_smry_4,
plot_file = exp_2019_report_4);
```

****Macro invocation example 2:

- Simulation number: 1000
- Sample size: 200
- Two arms, and randomization ratio: 1 vs. 1
- Proportion of Subgroups: 0.3 vs. 0.7
- Survival data distribution assumption: Weibull
- Parameter a in Weibull distribution: 2
- Parameter b in Weibull distribution: 2
- Hazard rate ratio, lowest to highest in Weibull distribution: 0.45
- Censor Proportion: 0.2
- Subgroup 1 in arm 2 hazard rate changing type: Convex
- Subgroup 2 in arm 2 hazard rate changing type: Concave
- Weight functions for log-rank test: LOGRANK FLEMING(1,0) TARONE;

```
%logrank_weight_sim
(seed_rand = 20200101,
simulation_n = 1000,
study_n = 200,
sgrp1_prop = 0.7,
```

```

data_distr  = "WEIBULL",
exp_h_hr    = ,
exp_l_hr    = ,
wb_a        = 2,
wb_b        = 2,
wb_hr       = 0.45,
cnsr_prop   = 0.2,
sgrp1_hf    = "CONV",
sgrp2_hf    = "CONC",
save_loc     = %str(root/home/work/work_losy/output),
sim_ds       = wb_20200101_6,
weight       = LOGRANK FLEMING (1,0) TARONE,
power_smry_ds = wb_smry_6,
plot_file    = wb_20200101_report_6);

```

АНОТАЦІЯ

Дана дипломна робота складається зі вступу, основної частини, висновка, списку використаних джерел та додатка. Обсяг роботи складає 52 сторінки. Основна частина займає 31 сторінку та містить 8 таблиць та 5 ілюстрацій. Документ містить 15 посилань на використані джерела.

Дане дослідження присвячене вивченню продуктивності лог-ранг тесту при використанні різних вагових функцій у клінічних дослідженнях. Метою дослідження була розробка методів симуляції даних про виживання у клінічних досліджень з різними параметрами, та оцінка ефективності використання варіантів лог-ранг тесту для них.

Результатом дослідження є програмний продукт, написаний мовою програмування SAS, який може бути застосований для досягнення мети дослідження (Додаток 1). Були проведені обчислювальні експерименти, які показують перевагу тих чи інших варіантів лог-ранг тесту для симуляцій з різними параметрами.

Ключові слова: лог-ранг тест, аналіз виживання, біостатистика, клінічні дослідження.

ANNOTATION

This thesis consists of an introduction, the core part, conclusion, references and an appendix. It contains 52 pages, of which the core part takes 31 page, including 8 tables and 5 illustrations. The document uses 15 references to external sources.

The subject of this research is assessing the performance of the log-rank test using different weights for clinical trials. The goal of the research was to develop methods of simulation the survival data in clinical trials with different parameters and assessment of the effectiveness of using various versions of log-rank test for them.

As a result, the program was developed using SAS, which can be used to achieve the goal of the research (Appendix 1). Several computational experiments were conducted, demonstrating the advantages that some versions of the log-rank test have for simulations with particular parameters.

Keywords: log-rank test, survival analysis, biostatistics, clinical trials.