

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ТОРГОВЕЛЬНО- ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ

Кафедра комп'ютерних наук та інформаційних систем

ВИПУСКНИЙ КВАЛІФІКАЦІЙНИЙ ПРОЕКТ

на тему:

**«Розробка програмного забезпечення багатокритеріальної
оцінки та відбору банків для розміщення грошових вкладів
фізичних осіб»**

Студента 4 курсу, 9 групи,

спеціальності

122 «Комп'ютерні науки»

Музикантов

Андрій

Дмитрович

підпис студента

Науковий керівник
кандидат технічних наук,
доцент

підпис керівника

Демідов Павло
Георгійович

Гарант освітньої програми
кандидат технічних наук, доцент

підпис керівника

Демідов Павло
Георгійович

Київ 2020

Київський національний торговельно-економічний університет

Факультет інформаційних технологій
Кафедра комп'ютерних наук та інформаційних систем
Спеціальність 122 «Комп'ютерні науки»

Зав. кафедри _____

Затверджую
Пурський О.І.
«20» грудня 2019р.

Завдання
на випускний кваліфікаційний проект студенту

Музикантову Андрію Дмитровичу

(прізвище, ім'я, по батькові)

- Тема випускного кваліфікаційного проекту
«Розробка програмного забезпечення багатокритеріальної оцінки та відбору банків для розміщення грошових вкладів фізичних осіб»
Затверджена наказом ректора від «04» грудня 2019 р. № 4111
- 2. Строк здачі студентом закінченого проекту _____
- 3. Цільова установка та вихідні дані до проекту
Мета проекту: розробка алгоритмів, інструментів та засобів для багатокритеріальної оцінки та відбору банків для розміщення грошових вкладів фізичних осіб.
Об'єкт дослідження: процеси депозитної діяльності банків та взаємовідносини з клієнтами.
Предмет дослідження: моделі, методи та інформаційні технології в комп'ютерній системі багатокритеріальної оцінки та відбору банків для розміщення грошових вкладів фізичних осіб.
- 4. Перелік графічного матеріалу _____
- 5. Консультанти по проекту із зазначенням розділів, за якими здійснюється консультування:

Розділ	Консультант (прізвище, ініціали)	Підпис, дата	
		Завдання видав	Завдання прийняв
1	Демідов П.Г.	15.12.2019 р.	15.12.2019 р.
2	Демідов П.Г.	15.12.2019 р.	15.12.2019 р.
3	Демідов П.Г.	15.12.2019 р.	15.12.2019 р.

6. Зміст випускного кваліфікаційного проекту (перелік питань за кожним розділом)

ВСТУП

РОЗДІЛ 1. ТЕОРЕТИЧНІ АСПЕКТИ ПРИЙНЯТТЯ РІШЕНЬ ВІДНОСНО БАНКІВ ДЛЯ РОЗМІЩЕННЯ КОШТІВ ФІЗИЧНИМИ ОСОБАМИ

1.1. Характеристика проблематики відбору кращого банку для розміщення клієнтом грошових коштів

1.2. Теорія нечітких множин та нечіткої логіки, методи багатокритеріального аналізу

1.3. Огляд програмного інструментарію та пакетів розробки нечітких систем

РОЗДІЛ 2. ПРОЕКТУВАННЯ КОМП'ЮТЕРНОЇ СИСТЕМИ ОЦІНКИ ТА ВІДБОРУ БАНКІВ ДЛЯ РОЗМІЩЕННЯ КОШТІВ ФІЗИЧНИМИ ОСОБАМИ

2.1. Постановка задачі

2.2. Метод максиміної згортки з урахуванням різних ваг критеріїв

2.3. Аналітичний апарат розрахунку фінансових показників банківської діяльності

2.4. Методика формування матриці попарного порівняння критеріїв та розрахунку їх вагових коефіцієнтів

2.5. Формування функцій приналежності на основі методу найменших квадратів

РОЗДІЛ 3. РОЗРОБКА КОМП'ЮТЕРНОЇ СИСТЕМИ ОЦІНКИ ТА ВІДБОРУ БАНКІВ ДЛЯ РОЗМІЩЕННЯ КОШТІВ ФІЗИЧНИМИ ОСОБАМИ

3.1. Алгоритм реалізації методу максиміної згортки для критеріїв з різними вагами

3.2. Алгоритм розрахунку ваг критеріїв

3.3. Характеристика програмного забезпечення комп'ютерної системи оцінки та вибору банків для розміщення коштів фізичними особами

3.4. Технологія використання розробленого додатку

ВИСНОВКИ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

7. Календарний план виконання проекту

№ Пор.	Назва етапів випускної кваліфікаційної роботи	Строк виконання етапів роботи	
		За планом	фактично
1	2	3	4
1	<i>Вибір теми випускного кваліфікаційного проекту</i>	01.10.2019	01.10.2019
2	<i>Розробка та затвердження завдання на випускний кваліфікаційний проект</i>	15.12.2019	15.12.2019
3	<i>Вступ</i>	03.02.2020	
4	<i>РОЗДІЛ 1. Теоретичні аспекти прийняття рішень відносно банків для розміщення коштів фізичними особами</i>	28.02.2020	
5	<i>РОЗДІЛ 2. Проектування комп'ютерної системи оцінки та відбору банків для розміщення коштів фізичними особами</i>	06.04.2020	
6	<i>РОЗДІЛ 3. Розробка комп'ютерної системи оцінки та відбору банків для розміщення коштів фізичними особами</i>	12.05.2020	
7	<i>Висновки</i>	15.05.2020	
8	<i>Здача випускного кваліфікаційного проекту на кафедру науковому керівнику</i>	20.05.2020	
9	<i>Попередній захист випускної кваліфікаційної роботи</i>	03.06.2020	
11	<i>Виправлення зауважень, зовнішнє рецензування випускної кваліфікаційної роботи</i>	09.06.2020	
12	<i>Представлення готової випускної кваліфікаційного проекту на кафедру</i>	12.06.2020	
13	<i>Публічний захист випускного кваліфікаційного проекту</i>	За розкладом роботи ЕК	

Анотація

У випускному кваліфікаційному проекті здійснено комплексну розробку програмного забезпечення багатокритеріальної оцінки та відбору банків для розміщення грошових вкладів фізичних осіб. Теоретично обґрунтовано основні положення щодо проблематики залучення депозитів, питань застосування нечітких множин та нечіткої логіки у банківській діяльності. Здійснено огляд програмного інструментарію та пакетів розробки нечітких множин. Охарактеризовано та практично застосовано методи максимінної згортки з урахуванням різних ваг критеріїв та формування матриці попарного порівняння критеріїв з розрахунком їх вагових коефіцієнтів. Створено програмний додаток для вибору кращого банку для розміщення депозиту.

Ключові слова: нечітка множина, багатокритеріальний аналіз, критерії оптимізації, метод максимінної згортки, фінансові показники, метод формування матриці попарних порівнянь критеріїв.

Anotation

The graduation qualification project is devoted to development of multicriteria evaluation and selection software for banks to place cash deposits of individuals. The main provisions on the issue of attracting deposits, the use of fuzzy sets and fuzzy logic in banking are theoretically substantiated. An overview of software tools and packages for the development of fuzzy sets. Methods of maximum convolution taking into account different weights of criteria and formation of a matrix of pairwise comparison of criteria with calculation of their weight coefficients are characterized and practically applied. A software application has been created to select the best bank to place a deposit.

Keywords: fuzzy set, multicriteria analysis, optimization criteria, maximum convolution method, financial indicators, method of forming a matrix of pairwise comparisons of criteria.

ЗМІСТ

ВСТУП.....	9
РОЗДІЛ 1. ТЕОРЕТИЧНІ АСПЕКТИ ПРИЙНЯТТЯ РІШЕНЬ ВІДНОСНО БАНКІВ ДЛЯ РОЗМІЩЕННЯ КОШТІВ ФІЗИЧНИМИ ОСОБАМИ.....	13
1.1. Характеристика проблематики відбору кращого банку для розміщення клієнтом грошових коштів.....	14
1.2 Теорія нечітких множин та нечіткої логіки, методи багатокритеріального аналізу.....	19
1.3 Огляд програмного інструментарію та пакетів розробки нечітких систем.....	27
РОЗДІЛ 2. ПРОЕКТУВАННЯ КОМП'ЮТЕРНОЇ СИСТЕМИ ОЦІНКИ ТА ВІДБОРУ БАНКІВ ДЛЯ РОЗМІЩЕННЯ КОШТІВ ФІЗИЧНИМИ ОСОБАМИ	32
2.1 Постановка задачі.....	32
2.2 Метод максимінної згортки з урахуванням різних ваг критеріїв.....	34
2.3. Аналітичний апарат розрахунку фінансових показників банківської діяльності.....	36
2.4. Методика формування матриці попарного порівняння критеріїв та розрахунку їх вагових коефіцієнтів.....	41
2.5. Формування функцій приналежності на основі методу найменших квадратів.....	44
РОЗДІЛ 3. РОЗРОБКА КОМП'ЮТЕРНОЇ СИСТЕМИ ОЦІНКИ ТА ВІДБОРУ БАНКІВ ДЛЯ РОЗМІЩЕННЯ КОШТІВ ФІЗИЧНИМИ ОСОБАМИ	52
3.1. Алгоритм реалізації методу максимінної згортки для критеріїв з різними вагами.....	52
3.2. Алгоритм розрахунку ваг критеріїв.....	54

3.3. Характеристика програмного забезпечення комп'ютерної системи оцінки та вибору банків для розміщення коштів фізичними особами	57
3.4. Технологія використання розробленого додатку.....	60
ВИСНОВКИ.....	64
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	66
ДОДАТОК.....	70

ВСТУП

Депозитні кошти відіграють важливу роль не тільки в діяльності банків, а й виступають стимулюючим чинником економічного зростання країни. З одного боку, вони виступають одним із основних джерел формування ресурсів у банківській системі, а з іншого - мобілізація тимчасово вільних коштів суб'єктів господарювання і заощаджень населення та їх розміщення в банках у вигляді депозитних вкладень дозволяє трансформувати їх у реальний капітал шляхом здійснення банком інших напрямів діяльності, а саме проведення кредитних, інвестиційних та інших операцій. Таким чином забезпечується потреба економіки у додаткових ресурсах.

Разом з тим внаслідок впливу сучасної фінансової кризи, нестабільності курсу національної валюти, негативних політичних та соціальних явищ в Україні відбувся значний відтік депозитних коштів як фізичних, так і юридичних осіб. Це в свою чергу завдало серйозного удару не лише по стабільності банківської системи, а й по ефективності національної економіки загалом.

В свою чергу низька поінформованість населення щодо депозитної політики банків, системи гарантування вкладів фізичних осіб аж ніяк не сприяє підвищенню інтересу населення до розміщення власних заощаджень в якості депозитів з метою не лише збереження коштів, а й отримання стабільного доходу. Відсутність реальної можливості порівняння банків, показників їх діяльності, переваг та недоліків розміщення депозиту в тому чи іншому банку є однією з основних перешкод в планомірному розвитку цього напрямку банківської діяльності в нашій країні.

За цих умов проблема можливості правильного розуміння фізичними особами всіх переваг розміщення заощаджень в якості депозиту за відсутності спеціальних знань або спеціальної галузевої освіти, можливості об'єктивного порівняння значної кількості показників діяльності одразу декількох банків із застосуванням методів багатокритеріального аналізу для виявлення кращого банку для розміщення депозиту стає надзвичайно актуальною.

Серед зарубіжних вчених дослідженням специфіки депозитних вкладів та різних аспектів формування та реалізації депозитної політики банків займалися такі вчені, як: А. Маршал, Е. Долан, П.Роуз та інші. До вітчизняних дослідників окресленої проблематики слід віднести: М. Алексеєнко, О. Васюренко, О. Дмитрієву, І. Івасіва, Ю. Коробова, О. Маркову, І. Мельникову, А. Мороза, А. Тютюнника тощо.

Апарат нечіткої логіки досліджувався у роботах таких вчених як Л. Заде., А.В. Матвійчуком, О.О. Недосекіним, О.В. Морозом, А.О. Свентухом та іншими.

Незважаючи на ґрунтовні наукові розробки з проблематики формування депозитної політики банків, методів математичного та економічного планування, питанням розробки окремої моделі, яка дозволила б споживачу за відсутності спеціальних знань та за наявності реальних фінансових показників банківських установ, визначити найкращий банк для розміщення депозиту, не приділялось належної уваги. Більш того, розробка спеціального програмного забезпечення, яке на підставі математичних розрахунків з використанням значної кількості критеріїв дозволить визначити найкращий банк для розміщення депозиту, дозволяє досягти найбільшої об'єктивності, виключити суб'єктивний фактор в прийнятті рішення, а також досягти довгострокових економічних цілей щодо підвищення рівня довіри населення до банків та збільшення обсягів депозитів. Зазначене вище зумовило актуальність даного дослідження.

Метою проекту є розробка алгоритмів, інструментів та засобів для багатокритеріальної оцінки та відбору банків для розміщення грошових вкладів фізичних осіб. Для досягнення поставленої мети необхідно було вирішити наступні завдання:

провести комплексне аналітичне дослідження проблематики відбору кращого банку для розміщення депозиту;

дослідити аспекти використання теорії нечітких множин та нечіткої логіки, методів багатокритеріального аналізу в банківській діяльності;

охарактеризувати наявний програмний інструментарій та пакети розробки нечітких систем;

дослідити аналітичний апарат розрахунку фінансових показників банківської діяльності;

виявити особливості методики формування матриці попарного порівняння критеріїв та розрахунку їх вагових коефіцієнтів;

сформувати функцію приналежності на основі методу найменших квадратів;

розробити програмне забезпечення відбору кращого банку для розміщення грошового вкладу фізичним особами;

визначити технологію використання розробленого додатку.

Об'єкт дослідження - процеси депозитної діяльності банків та взаємостосунки з клієнтами.

Предмет дослідження - моделі, методи та інформаційні технології в комп'ютерній системі багатокритеріальної оцінки та відбору банків для розміщення грошових вкладів фізичних осіб.

Методи дослідження. Теоретичною основою дослідження є загальнонауковий аналітичний метод, а також системний підхід за напрямом дослідження. Інформаційну базу дослідження становлять дані офіційної звітності Національного банку України, окремих банків України. Для практичного вирішення поставлених задач використовувалися такі методи:

загальнонауковий аналітичний метод;

методи математичного моделювання;

методи багатокритеріальної оцінки даних;

методи алгоритмічного програмування для створення програмного забезпечення для відбору найкращого банку для розміщення грошових вкладів фізичних осіб.

Практичне значення. Отримані результати, можуть бути використані в практичній діяльності банків з метою залучення депозитів від населення.

Програмна реалізація запропонованої моделі відбору кращого банку для розміщення коштів фізичною особою надає можливості використання всієї множини даних офіційної фінансової звітності банків, що виключає можливість викривлення моделі та вплив суб'єктивного фактору при оцінці зазначених вихідних даних.

РОЗДІЛ 1.

ТЕОРЕТИЧНІ АСПЕКТИ ПРИЙНЯТТЯ РІШЕНЬ ВІДНОСНО БАНКІВ ДЛЯ РОЗМІЩЕННЯ КОШТІВ ФІЗИЧНИМИ ОСОБАМИ

1.1. Характеристика проблематики відбору кращого банку для розміщення клієнтом грошових коштів

На сучасному етапі розвитку банківської системи України суттєво зростає роль і місце депозитної діяльності, від інтенсивності і результативності якої залежать насамперед рівень забезпеченості банку фінансовими ресурсами та можливості здійснення активних операцій за обсягом та структурою. Від об'єму, видів, структури та ціни залучених ресурсів залежать можливості банку підтримувати свою ліквідність і отримувати доходи та прибуток в необхідному обсязі. Саме тому, сьогодні значний інтерес викликають питання, пов'язані із формуванням депозитних ресурсів, вивченням умов створення сприятливого середовища та методами щодо активізації їх залучення тощо.

Успішність депозитної політики банку залежить від багатьох факторів, зокрема від обізнаності споживача в сфері фінансових послуг, в тому числі правильного розуміння ним сутності депозитних послуг, їх мети, якості надання тощо. Отже, основними фундаментальними напрямками депозитної політики банку є забезпечення, з одного боку, умов для задоволення потреб клієнтів в депозитних послугах, які пропонуються банком, а з іншого - отримання прибутку за рахунок найбільш вигідного розміщення залучених коштів в активні операції.

Разом з тим, якою б продуктивною не була депозитна політика банку, з іншого боку завжди є споживач, від рівня обізнаності якого залежить сам факт можливості залучення його заощаджених коштів у якості депозиту.

Інформація щодо надійності та вигідності грошового вкладу, показників успішності того чи іншого банку, правильне розуміння сутності депозиту та його мети, має бути відкритою та доступною для споживача, адже лише за умови її

наявності в повному обсязі фізична особа може зробити усвідомлений вибір кращого банку для розміщення грошового вкладу.

Як слушно зазначає Н.Г. Радченко, в наукових працях вітчизняних економістів досліджуються показники, що відображають ефективність окремих напрямків депозитної діяльності банку [1, с. 161]. Натомість комплексного дослідження критеріїв ефективної депозитної діяльності банків з метою можливості їх рейтингування за цими ж критеріями, а тим більше дослідження критеріїв відбору кращого банку для розміщення депозиту як цілісної системи ані в економічній, ані в комп'ютерній науці досі немає.

Для того щоб визначити певну систему критеріїв, яка дозволила б споживачу на основі конкретних показників діяльності банку визначити серед них найкращий, потрібно в першу чергу дослідити ряд проблем, з якими стикаються банки при формуванні депозитного портфелю, а також виявити як саме ці проблеми впливають на реальні показники діяльності банків. Адже саме системний аналіз цих показників та їх рейтингування за певними критеріями і мають лягти в основу відбору кращого банку для розміщення грошового вкладу фізичною особою.

Основною проблемою з формування депозитного портфелю, яка на даний момент потребує негайного вирішення, є відтік грошових коштів з депозитних рахунків клієнтів, який ще називають депозитним ризиком. За класифікацією Національного банку України (Далі - НБУ) даний ризик не виділяється в окрему категорію, проте досить багато вчених приділяють увагу вивченню саме цього виду ризику. Адже ідентифікація ризику в подальшому дасть змогу розвинути методи його оцінки, а, як наслідок, і методи управління [2, с. 16].

Так як ризик це не лише ймовірність втрати, а й ймовірність отримати додаткову вигоду, то при прийнятті обґрунтованих доцільних рішень банк зможе отримати додаткову вигоду. Отже для стабільної роботи банку даний ризик необхідно мінімізувати та заохочувати клієнтів, щоб вони не тільки достроково не вилучали кошти, а й після закінчення строку дії договору продовжили його дію.

Також однією з проблем є недовіра клієнтів до стабільності та надійності роботи банків. На сучасному етапі банківська діяльність характеризується залежністю ефективної діяльності банку від наявної бездоганної репутації, відгуків про високу якість обслуговування клієнтів. Все це свідчить про необхідність орієнтації на задоволення потреб клієнтів при одночасному дотриманні усіх встановлених норм ліквідності та прибутковості [3].

Ще однією проблемою для банку є диверсифікація депозитного портфелю банку. Завдання банку полягає в поєднанні прибутковості та стабільності ресурсної бази, а саме її складової — депозитів фізичних осіб.

Як відзначає Ю.П. Макаренко, для підвищення стабільності депозитної бази в першу чергу необхідно вжити заходів на макрорівні, створивши певні сприятливі умови. Це повинні бути заходи НБУ зі стабілізації грошово-кредитного та валютного ринків України, які б дійсно дали змогу забезпечити стабільне функціонування всього банківського сектору. До них, на думку автора, слід віднести в тому числі й заходи по стабілізації курсу національної валюти до долара США та створення умов для припливу зовнішніх інвестицій, які дали б змогу розвиватися підприємствам, а отримані ними прибутки вкладати на депозитні рахунки, що сприяло б збільшенню обсягу депозитного портфелю банку та розвитку на основі даних ресурсів кредитної діяльності, що є основним джерелом доходу для будь-якого банку [2, с. 17].

Що стосується ціноутворення, то єдиним напрямом залучення додаткових коштів юридичних та фізичних осіб та збереження вже існуючих клієнтів, які мають намір забрати свої кошти, є підвищення депозитних ставок. Проте це рішення повинно бути обґрунтованим, адже депозити як вид банківських операцій не приносять доходу для банку. Тому необхідно розробити модель, для кожного банку окремо, звертаючи увагу на специфіку його діяльності, яка б дала змогу оцінити, який максимальний відсоток банк може запропонувати своєму клієнту, щоб при цьому порівнявши доходи та витрати, отримати прибуток.

І нарешті ще одним проблемним напрямом в цій сфері є диференціація тарифних ставок для окремих вкладників як перспективний метод ціноутворення банківських депозитів. Адже для клієнтів, які вже довгий час зберігають кошти в даному банку і через несприятливі умови раптом вирішили їх забрати, можна запропонувати підвищення дохідності за депозитом. При цьому ризик зняття вкладу значно менший порівняно з новим клієнтом.

Отже як бачимо, проблема оцінки ефективності депозитної політики банку надзвичайно багатогранна. Разом з тим, найбільш важливим у ній є питання про критерії. При цьому сукупність критеріїв, якими керується банк як одна зі сторін депозитних відносин для оцінки ефективності своєї депозитної політики, та сукупність критеріїв, якими керується споживач як інша сторона депозитних відносин при відборі кращого банку для розміщення власних заощаджених коштів, дещо відрізняються. Так перші неодмінно слугують відправними, адже саме за ними визначається ступінь надійності та стійкості банку як фінансової установи. А другі вже включають в себе показники, що впливають як логічний наслідок ефективної депозитної політики банку, такі як розмір процентної ставки, розмір депозитного портфелю, лояльність клієнтів тощо.

Критерії оцінки ефективності депозитної політики самого банку з одного боку, та критерії відбору споживачем кращого банку для розміщення грошового вкладу з іншого боку можна визначити як раніше обрані точки або нормативи в загальній програмі діяльності банку, за якими проводиться визначення ефективності формування та використання ресурсної бази, а також оцінка якості депозитних послуг банку споживачем.

В економічній науці здебільшого досліджено саме критерії ефективності депозитної політики банку. Погоджуємось з думкою, що доцільною є концентрація на тому, що вибір показників для оцінки процесу залучення капіталу банку має спиратися не на суб'єктивні думки аналітиків, а на встановлення суворої залежності від цих показників депозитної політики банку [4, с. 60].

Основними вимогами, яким мають задовольняти показники, використовувані при оцінці ефективності депозитної політики банку, є: обумовленість чинниками, що впливають на процес залучення банком вільних ресурсів; зіставність між собою за розмірністю; відсутність суперечностей при зміні значень [5, с. 69]. Стисло розглянемо ті відправні показники (критерії), які найчастіше виділяються в економічній науці, при характеристиці депозитної політики банку.

Одним з основних завдань, які визначені Комплексною програмою розвитку банківської системи України, є підвищення ефективності управління процесом залучення вільних коштів, удосконалення систем формування ресурсної бази з метою максимізації прибутковості активів і мінімізації ризику діяльності банку. Іншими словами мова йде про такий критерій як економічна вигідність.

Визначаючи економічну вигідність депозитних операцій, що проводяться, необхідно здійснювати аналіз зобов'язань за їх видами з урахуванням вартості фінансових ресурсів у рамках загальноприйнятої класифікації. Такий аналіз проводиться з використанням методу коефіцієнтів. За основу беруться показники, що дозволяють охарактеризувати залежність між капіталом і прибутковістю банку в аспекті збільшення власних коштів і їх ліквідності при певній величині активів. Отже даний критерій базується на концепції довгострокової рівноваги, яка припускає, що чим вище ліквідність, тим міцніше фінансовий стан банку, його капітальна база, і навпаки: чим нижче ліквідність, тим менш стійкий банк [6, с.211].

Наступний показник - це коефіцієнт відношення зобов'язань банку і капіталу. Він відображає величину залучених банком ресурсів на кожну одиницю капіталу і розраховується як відношення зобов'язань до капіталу. Зростання його значення свідчить про підвищення банком активності у сфері залучення ресурсів, але в той же час може сигналізувати про зниження надійності і стійкості банку [7, с. 52]. Тому логічним доповненням до цього коефіцієнта є показники надійності,

які відображають, наскільки вкладення банку у працюючі активи захищені власним капіталом банку і навпаки - рівень надійності капіталу в пасивах банку.

Наступна група показників дозволяє проаналізувати ефективність різних напрямів депозитної політики, до них належать:

1. коефіцієнт, що характеризує ефективність використання банком залучених ресурсів;
2. коефіцієнт, що відображає ефективність операцій банку;
3. коефіцієнт, який демонструє ефективність використання власного капіталу;
4. коефіцієнт, який визначає розмір прибутку відносно валюти балансу, або ефективність використання ресурсів [8, с. 206].

Слід зазначити, що всі, вищезазначені коефіцієнти, які характеризують депозитну базу банку, є односпрямованими, тобто більше їх значення відповідає кращій якості. Оцінюючи ефективність депозитної політики банку з точки зору економічної вигідності, необхідно чітко і детально відстежувати стан кредитної і депозитної політики банку в сукупності. Для цього, перш за все необхідно розрахувати коефіцієнт співвідношення кредитів і депозитів. Цей коефіцієнт характеризує здатність банку залучати депозити для проведення своїх кредитних операцій і можливості банку розміщувати залучені кошти в кредити. Невід'ємною частиною при розрахунку даного коефіцієнта є аналіз і планування залишку коштів на коррахунку банку [9, с. 55].

У процесі розрахунку економічної вигідності депозитної політики, що проводиться, і встановленні тісного взаємозв'язку між кредитними і депозитними операціями важливим моментом є проведення оцінки обґрунтованості залучення міжбанківських кредитів. Для цього необхідно розрахувати декілька коефіцієнтів:

- коефіцієнт максимального розміру отриманих міжбанківських кредитів;
- коефіцієнт максимального розміру розміщених міжбанківських кредитів.

Для визначення стабільності залучених коштів необхідно розрахувати такі показники: термін зберігання депозитних коштів і рівень осідання коштів на

депозитних рахунках. Чим вище термін зберігання і рівень осідання коштів, тим ефективніше депозитна політика банку. Якщо значення коефіцієнта більше одиниці, то депозитна політика банку вважається успішною [10, с. 103]. А отже такий банк ставатиме більш надійним й з точки зору споживача при відборі ним кращого банку для розміщення депозиту.

Така система критеріїв та показників оцінки ефективності депозитної діяльності банку є основою для визначення системи критеріїв відбору кращого банку для розміщення грошового вкладу, які буде досліджено в пункту 2.1. цієї роботи.

Отже, основними проблемами ефективного розміщення грошових вкладів фізичними особами в умовах нестабільності фінансової системи України і постійних ринкових коливань є такі: 1) неконтрольований відтік грошових коштів з депозитних рахунків клієнтів (депозитний ризик); 2) занадто лояльні законодавчі вимоги щодо диверсифікації депозитного портфелю банку, що в свою чергу підвищує ризиковість активних банківських операцій; 3) низький рівень лояльності населення до депозитних вкладів (що обумовлено недовірою клієнтів до стабільності та надійності роботи банків); 4) низька обізнаність споживачів щодо основних показників діяльності банків як фінансових установ та ступеню їх надійності.

1.2. Теорія нечітких множин та нечіткої логіки, методи багатокритеріального аналізу

В наш час через значні фінансові і економічні коливання ринку, а також через те, що соціальні, технологічні та економічні фактори впливу на банківську сферу стали менш передбачуваними, формування правильної стратегії розвитку, а також фінансової політики (зокрема депозитної політики) стає дедалі більш складним завданням для будь-якого банку. В умовах постійної нестабільності економічної ситуації в країні банки вимушені працювати в умовах невизначеності щодо майбутнього фінансового стану.

Для вивчення проблем, пов'язаних з функціонуванням будь-яких організацій (зокрема, банків), використовуються методи, що дозволяють домогтися найкращого розуміння явищ, що виникають кожного дня. Традиційно таке розуміння дійсності ґрунтувалося на понятті точності, і для її кількісної оцінки зверталися до класичних математичних схем.

Зараз, коли сучасний світ вимагає від керівництва організації використання реалістичного підходу до управління та прийняття рішень, стає актуальним застосування теорій, які дозволили б представляти явища без спотворення, обумовленого прагненням до точності і чіткості. Виникає потреба в застосуванні систем, які дали б можливість активного використання різних думок осіб, які здійснюють планування і приймають рішення, а також нечіткої інформації, вираженої словами. Однією з таких теорій, які допомагають вирішувати дані проблеми, є теорія нечітких множин, яка являє собою розділ математики, добре пристосований для відображення суб'єктивного і невизначеного.

Термін «нечітка множина» (fuzzy set) вперше з'явився в 1965 році, коли професор Каліфорнійського університету Лотфі Заде розширив двозначну оцінку 0 або 1 до необмеженої багатозначною оцінки в інтервалі $[0,1]$ і вперше ввів поняття «нечітка множина».

В даний час методи нечіткої логіки застосовуються в банківській сфері для наступних цілей:

1. Аналіз ризиків інвестиційних проектів. Використання методів нечіткої логіки для аналізу ризиків пов'язано з тим, що на всіх етапах свого життєвого циклу інвестиційні проекти супроводжуються невизначеністю, яка має на увазі неможливість повною мірою визначити цілі і сформулювати вимоги проекту, неповноту знання всіх параметрів, обставин і ситуацій. Невизначеність тут є причиною ризиків.
2. Опис бізнес-процесів. Теорія нечіткої логіки є новим підходом до опису бізнес-процесів, в яких присутня невизначеність, яка ускладнює і навіть виключає застосування точних кількісних методів і підходів.

3. Оцінка кредитоспроможності фізичних осіб. Використання методів теорії нечітких множин дає можливість створити систему складання оцінки кредитоспроможності з безперервною шкалою. Така шкала за своєю точністю буде ближчою до оцінки групи експертів, ніж та, що побудована з використанням методів, заснованих на класичній логіці.

4. Оптимізація і прийняття рішень. Використання систем підтримки прийняття рішень, побудованих на основі теорії нечітких множин, дозволяє підвищити ступінь обґрунтованості прийнятих рішень. Дані системи допомагають розглянути всі можливі сценарії розвитку подій.

5. Оцінка ефективності IT-сервісів. Зараз більшість банків розробляють підходи до управління IT-сервісами, спираючись на бібліотеку IT Infrastructure Library (ITIL), що описує кращі з застосовуваних на практиці способи організації роботи підрозділів або компаній. Методика з управління рівнем сервісу, пропонована ITIL, ґрунтується лише на чітких формулюваннях. У разі, коли параметри ефективності не можуть бути представлені чіткими величинами, а можуть бути виражені лише експертними оцінками, управління рівнем сервісу по ITIL стає складним або неможливим. І для оцінки ефективності IT-сервісів застосовуються методи нечіткої логіки.

Враховуючи широкий спектр застосування теорії нечітких множин в банківській сфері, зокрема при визначенні критеріїв відбору споживачем кращого банку для розміщення депозиту та правильної їх оцінки, важливим є чітке розуміння самого терміну "нечітка множина", а також методу багатокритеріального аналізу.

Нечіткість інформації обумовлена наявністю в описах задач процесів прийняття рішень понять і відношень з нестрогими межами, а також висловлень з багатозначною шкалою істинності. Об'єкт може належати до класу, що описується даним поняттям, відношенням або висловленням, може не відноситися до нього, але можливі й проміжні градації приналежності. Поняття і відношення, що описують такі класи, відносяться в математичній науці до нечітких [11, с.81].

Під нечіткою множиною прийнято розуміти спробу математичної формалізації нечіткої інформації з метою її використання при побудові математичних моделей складних систем. В основі цього поняття лежить уявлення про те, що елементи, які складають дану множину та володіють загальною властивістю, можуть володіти цією властивістю в різному ступені і, отже, належати до даної множини в різному ступені. При цьому підході твердження "елемент x належить даній множині" втрачає сенс, оскільки необхідно вказати "наскільки сильно" або з яким ступенем даний елемент належить даній множині.

Розглянемо основні елементи теорії нечітких множин [13, с. 84]. Нехай U - повна множина, що охоплює всі об'єкти деякого класу. Нечітка підмножина F множини U , яке в подальшому будемо називати нечіткою множиною, визначається через функцію приналежності $F(u)$, $u \in U$. Ця функція відображає елементи U , множини U на множину дійсних чисел відрізка $[0,1]$, які вказують ступінь приналежності кожного елемента нечіткій множині F .

Якщо повна множина U складається з кінцевого числа елементів U_i , $i = 1, 2, \dots, n$, то нечітку множину F можна представити в наступному вигляді:

$$F = \mu_F(u_1)/u_1 + \mu_F(u_2)/u_2 + \dots + \mu_F(u_n)/u_n \quad (1.1)$$

де "+" означає не додавання, а, скоріше, об'єднання: символ "/" показує, що значення F відноситься до елемента, наступного за ним (а не означає поділ на U_i).

У разі, якщо множина U є безперервною, F можна записати як інтеграл:

$$F = \int_U \mu_F(u) / u. \quad (1.2)$$

Варто зазначити, за допомогою нечітких описів строгою мовою математики можна формулювати та розв'язувати навіть такі задачі, в яких присутні не тільки числові значення, але й лінгвістичні висловлювання, тобто нечислові змінні, а також оцінювати ефективність функціонування системи через поєднання кількісних і якісних показників, розглядаючи їх не тільки у статиці, але й динаміці.

Основним поняттям теорії нечіткої логіки є змінна: числова або лінгвістична. Загальна процедура побудови моделі об'єкта з використанням методів нечітких множин, яка описана вище, включає створення множини значень змінних для можливих станів об'єкту і відповідно множини оцінок (числових чи лінгвістичних) цих значень.

Що стосується лінгвістичних змінних, то слід виділити певні особливості їх використання. Згідно Лотфі Заде, лінгвістичною називається змінна, значеннями якої є слова або вирази природної чи штучної мови.

Нечіткі описи у структурі методу аналізу ризику з'являються у зв'язку з невпевненістю експерта, яка виникає у процесі різноманітного роду класифікацій[12, с.20]. Наприклад, коли треба провести межу між середнім та низьким рівнем значень параметру. Тоді застосування нечітких описів має наступне значення:

1. Експерт будує лінгвістичну змінну зі своєю терм-множиною значень. Наприклад: змінна "Рівень ризику банкрутства" може володіти терм-множиною значень "Дуже низький, Низький, Середній, Високий, Дуже високий".

2. Експерт обирає відповідну для неї кількісну ознаку, для того щоб конструктивно зробити опис лінгвістичної змінної - наприклад, сконструйований показник рівня ризику банкрутства, який приймає значення від нуля до одиниці.

3. Експерт кожному значенню лінгвістичної змінної (яке, за своєю будовою, є нечіткою підмножиною значень інтервалу $(0,1)$ галузі значень показників рівня ризику банкрутства) зіставляє функцію приналежності рівня ризику банкрутства тій або іншій нечіткій підмножині.

Нечіткі множини широко застосовуються для формалізації лінгвістичних знань. Розглянемо для прикладу множину процентних ставок, що надаються банками за вкладами. Яким чином можна виділити підмножину високих процентних ставок? В умовах динамічно змінюваного середовища не завжди можливо точно відповісти на це питання, однозначно виділивши множину високих ставок. При використанні апарату теорії нечітких множин вирішити таке

завдання можна навіть за відсутності повної кількісної інформації про оточення. Функція приналежності для елементів нечіткої множини F_1 , відповідних поняттю "високі процентні ставки" (рис. 1.1), буде мати наступний вигляд:

$$\mu_{F_1}(u) = 0,1/30 + 0,2/35 + 0,4/40 + 0,6/45 + 0,8/50 + 1/60 + 1/70. \quad (1.3)$$

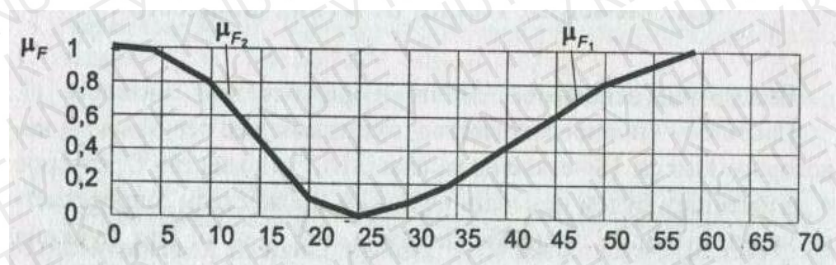


Рис 1.1. Функції приналежності нечітких множин

Функція приналежності до нечіткої множини низьких процентних ставок запишеться наступним чином:

$$\mu_{F_2}(u) = 1/0 + 1/5 + 0,8/10 + 0,5/15 + 0,2/20. \quad (1.4)$$

Як бачимо, елементи теорії нечітких множин можуть доволі успішно застосовуватися для прийняття рішень. Експертні оцінки альтернативних варіантів за критеріями можуть бути представлені як нечіткі множини або числа, виражені за допомогою функцій належності. Для упорядкування нечітких чисел існує безліч методів, які відрізняються один від одного способом згортки і побудови нечітких відносин. Останні можна визначити як відносини переваги між об'єктами[13, с. 85]. Розглянемо одну з математичних постановок задач прийняття оптимальних рішень на основі теорії нечітких множин.

Майже будь-яке складне технічне завдання прийняття рішення є багатокритерійним, оскільки при виборі найкращого варіанту доводиться враховувати багато різних вимог, і серед цих критеріїв зустрічаються такі, що суперечать один одному. Проте, майже всі математичні методи оптимізації призначені для знаходження екстремуму однієї функції – тобто для знаходження оптимального розв'язання за одним критерієм. Тому найчастіше намагаються звести багатоцільове завдання до одноцільового. Ця процедура в

більшості випадків призводить до серйозного спотворення сутності проблеми і, отже, до невиправданої заміни одного завдання іншим.

Якщо при розв'язанні одноцільових задач методологічних проблем не виникає, а можливі тільки обчислювальні труднощі, то інакше йде справа з багатокритеріальними задачами. Тут основні нюанси пов'язані з наступною проблемою: що слід вважати якнайкращою альтернативою в завданні з декількома цільовими функціями, які суперечливі між собою, і досягають максимуму в різних точках множини альтернатив?

Цілі багатокритеріальної задачі можуть знаходитися одна з одною в наступних відносинах[14, с. 15]:

1. Цілі взаємно нейтральні. В такій ситуації система може стосовно окремих цілей характеризуватися і розглядатися незалежно.
2. Цілі кооперуються. Тут, як правило, систему вдається розглянути стосовно однієї мети, а інші досягаються одночасно.
3. Цілі конкурують. В цьому випадку одну з цілей можна досягти лише за рахунок іншої.

Якщо цілі частково нейтральні, частково кооперовані і частково конкурують між собою, то завдання формулюється таким чином, що потрібно брати до уваги тільки конкуруючі цілі. Розгляд нейтральних або кооперативних цілей не представляє особливих труднощів, так що проблеми, орієнтовані на множину цілей, перш за все повинні бути розглянуті в частині конкуруючих цілей, якщо всі вони разом не можуть бути виражені одновимірним параметром[13, с. 87].

В даному випадку критерії визначають деякі поняття, а оцінки альтернатив є мірою відповідності цим поняттям. Нехай є множина альтернатив $A = \{a_1, a_2, \dots, a_m\}$ і безліч критеріїв $C = \{C_1, C_2, \dots, C_n\}$, при цьому оцінки альтернатив по кожному i -му критерію представлені нечіткими множинами:

$$C_i = \{C_i(a_1)/C_i(a_2), \dots, C_i(a_m)/a_m\} \quad (1.5)$$

Правило вибору кращої альтернативи можна уявити як перетин нечітких множин, які відповідають критеріям:

$$D = C_1 \cap C_2 \cap \dots \cap C_n. \quad (1.6)$$

Операція перетину нечітких множин може бути реалізована різними способами. Іноді перетин виконується як множення, але зазвичай цій операції відповідає взяття мінімуму:

$$\mu_D(a_j) = \min_{i=1, \dots, n} \mu_{C_i}(a_j), j = 1, \dots, m. \quad (1.7)$$

Кращою вважається альтернатива a^* , що має найбільше значення функції приналежності

$$\mu_D(a^*) = \max_{j=1, \dots, m} \mu_D(a_j). \quad (1.8)$$

Якщо критерії C_i мають різну важливість, то їх внесок в загальне рішення можна уявити як зважений перетин:

$$D = C_1^{a_1} \cap C_2^{a_2} \cap \dots \cap C_n^{a_n}, \quad (1.9)$$

де a_i - вагові коефіцієнти відповідних критеріїв, які повинні відповідати таким вимогам:

$$\alpha_i \geq 0; i = 1, \dots, n; (1/n) \sum_{i=1}^n \alpha_i = 1. \quad (1.10)$$

Коефіцієнти відносної важливості можна визначити, використовуючи процедуру попарного порівняння критеріїв.

Отже, якщо вирішення певної математичної задачі потребує знаходження оптимального рішення на підставі аналізу певної кількості критеріїв, то найкращим в даному випадку є метод багатокритеріального аналізу. При цьому використання нечітких множин обумовлене необхідністю побудови цільової функції по кожному критерію для наявних альтернатив, через можливість необмеженої багатозначної оцінки в інтервалі $[0,1]$.

1.3. Огляд програмного інструментарію та пакетів розробки нечітких систем

Нечітке моделювання є на сьогодні актуальним та перспективним напрямком сучасних інтелектуальних технологій для дослідження складних нелінійних систем та процесів прийняття рішень щодо управління.

Традиційні класичні методи побудови моделей не є ефективними, коли досліджуваний об'єкт чи система не мають повного адекватного формалізованого опису. Сучасна наука має досить потужні математичні засоби для отримання вичерпної інформації у випадку вирішення складних нелінійних задач, але розв'язання ними проблематики може призвести до втрати часу та ресурсів і не може гарантувати в принципі побудову адекватної моделі.

На сьогодні у разі вирішення задач моделювання складних систем в економіці найбільш доцільним є використання методів, що орієнтовані на побудову моделей в умовах невизначеності та недостовірності вхідних даних. Технологія нечіткого моделювання саме і побудована на таких методах. Ці методи ґрунтуються на теорії нечітких множин та нечіткої логіки.

Нечітка модель – це інформаційно–логічна модель системи чи підсистеми, що побудована на основі теорії нечітких множин та нечіткої логіки. Нечітке моделювання є складовою системного моделювання і покликане конкретизувати ті чи інші елементи або вузли моделі системи [15, с. 154].

Технологія нечіткого моделювання включає такі етапи:

- аналіз проблемної ситуації;
- структуризація предметної області і створення нечіткої моделі;
- дослідження нечіткої моделі;
- обробка результатів дослідження;
- коригування моделі.

Найбільш поширеними програмними пакетами нечіткого моделювання є додаток (toolbox) FUZZY в системі MATLAB та спеціалізоване програмне середовище FUZZYTECH.

Fuzzy Logic Toolbox - це пакет розширення MATLAB, що містить інструменти для проектування систем нечіткої логіки. Даний пакет дозволяє створювати експертні системи на основі нечіткої логіки, проводити кластеризацію нечіткими алгоритмами, а також проектувати нечіткі нейромережі[16].

Він включає графічний інтерфейс для інтерактивного покрокового проектування нечітких систем, функції командного рядка для розробки програм, а також спеціальні блоки для побудови систем нечіткої логіки в Simulink.

Всі функції пакета написані на відкритій мові MATLAB, що дозволяє контролювати виконання алгоритмів, змінювати вихідний код, а також створювати свої власні функції і процедури.

Ключовими особливостями даного програмного забезпечення є наступні:

- 1) графічний інтерфейс для інтерактивного покрокового проектування нечітких систем;
- 2) функції для створення експертних систем на основі нечіткої логіки;
- 3) підтримка логіки І, АБО і НЕ в налаштованих правилах;
- 4) стандартні типи експертних систем нечіткої логіки (Mamdani, Sugeno);
- 5) функції для нейроадаптивної і нечіткої кластеризації з навчанням;
- 6) включення нечітких систем в Simulink-моделі;
- 7) генерація 3-коду і незалежних додатків, що реалізують системи нечіткої логіки.

Наступним програмним забезпеченням є пакет fuzzyTECH. Програмний пакет fuzzyTECH розроблений і постійно модифікується німецькою компанією Infor'm GmbH (Inform Software Corporation). Він призначений для вирішення різних завдань нечіткого моделювання та управління об'єктами в умовах важко оцінюваних і важко контрольованих поведінок.

У порівнянні з пакетом Fuzzy Logic Toolbox, що входять в Matlab, пакет fuzzyTECH є спеціалізованим засобом, який дозволяє розробляти різноманітні нечіткі системи в графічному режимі, а також транслювати їх в програму на одній з мов програмування, в тому числі і для реалізації її в програмованих мікроконтролерах.

Найбільш просто створювати нечіткі системи, використовуючи засоби Fuzzy Design Wizard. Початковий варіант програми системи створюється за кілька кроків:

- 1) отримання інформації про структуру та об'єкти нечіткої системи;
- 2) складання опису системи шляхом визначення об'єктів, лінгвістичних змінних, функцій приналежності і блоків правил;
- 3) формування нечітких правил для системи.

Далі для поліпшення поведінки системи можна використовувати різні засоби, що дозволяють виконати:

- оптимізацію системи шляхом застосування налагоджувальних процедур;
- оцінку характеристик системи, використовуючи процес обробки або згенеровані дані;
- верифікацію поведінки системи, використовуючи аналітичні інструменти fuzzyTECH.

Після завершення проекту з розробки нечіткої системи можна використовувати інструменти fuzzyTECH для створення документації по проекту і управління ревізією. Компонент Project Information Dialog дозволяє ввести користувачеві первинну інформацію про проект, таку як автор проекту, дата останньої зміни проекту, коментарі тощо. Документація про проект, завантажена в fuzzyTECH, генерується автоматично в RTF-файл протягом декількох хвилин шляхом використання компонента Documentation Generator fuzzyTECH[17, с.230].

Пакет fuzzyTECH має також компонент Revision Control System, який дозволяє завантажувати, зберігати і знищувати файли проекту на стадії ревізії. Управління ревізією зберігає всі важливі кроки розробки в повній історії ревізії, щоб у разі необхідності переглянути ранні стадії розробки.

Детальний опис всіх можливостей і інструментів fuzzyTECH міститься у «Керівництві користувача» і бібліотеці додатків «Fuzzy Application Library».

Дані програмні додатки є універсальними для вирішення задач, пов'язаних із пошуком оптимального рішення. Через це інструментарій цих програмних забезпечень є перенасиченим багатьма функціями, що не використовуються для вирішення багатокритеріальних задач, зокрема задачі відбору кращого банку для розміщення грошового вкладу.

ВИСНОВКИ ЗА РОЗДІЛОМ 1.

Основними проблемами ефективного розміщення грошових вкладів фізичними особами в умовах нестабільності фінансової системи України і постійних ринкових коливань є такі: 1) неконтрольований відтік грошових коштів з депозитних рахунків клієнтів (депозитний ризик); 2) занадто лояльні законодавчі вимоги щодо диверсифікації депозитного портфелю банку, що в свою чергу підвищує ризиковість активних банківських операцій; 3) низький рівень лояльності населення до депозитних вкладів (що обумовлено недовірою клієнтів до стабільності та надійності роботи банків); 4) низька обізнаність споживачів щодо основних показників діяльності банків як фінансових установ та ступеню їх надійності.

В свою чергу наявність цих проблем ускладнює не лише вибір критеріїв відбору кращого банку для розміщення грошового вкладу фізичною особою, але й можливість ранжування цих критеріїв та порівняння їх значущості між собою. Саме це дозволило зробити висновок про оптимальне вирішення цієї задачі за допомогою математичного моделювання з використанням методу багатокритеріального аналізу, спрямованого на знаходження оптимального рішення на підставі аналізу певної кількості критеріїв.

Так, використання нечітких множин обумовлене необхідністю побудови цільової функції по кожному критерію для наявних альтернатив, через можливість необмеженої багатозначної оцінки в інтервалі $[0,1]$. А наявність великої кількості конкурентоспроможних критеріїв призводить до необхідності використання

методу багатокритеріального аналізу, за допомогою якого вирішується поставлена задача знаходження оптимального рішення.

Найбільш поширеними програмними пакетами нечіткого моделювання є додаток (toolbox) FUZZY в системі MATLAB та спеціалізоване програмне середовище FUZZYTECH. Дані програмні додатки є універсальними для вирішення задач, пов'язаних із пошуком оптимального рішення. Через це інструментарій цих програмних забезпечень є перенасиченим багатьма функціями, що не використовуються для вирішення багатокритеріальних задач, зокрема задачі відбору кращого банку для розміщення грошового вкладу.

Це зумовлює необхідність розробки окремого програмного забезпечення, яке б дозволило прийняти оптимальне рішення на підставі експертної оцінки основних показників діяльності банків та інших даних, що мають значення для прийняття рішення (критеріїв відбору), а також побудови матриці попарного порівняння цих критеріїв.

РОЗДІЛ 2.

ПРОЕКТУВАННЯ КОМП'ЮТЕРНОЇ СИСТЕМИ ОЦІНКИ ТА ВІДБОРУ БАНКІВ ДЛЯ РОЗМІЩЕННЯ КОШТІВ ФІЗИЧНИМИ ОСОБАМИ

2.1. Постановка задачі

Процес побудови оптимізаційної задачі включає наступні етапи:

- 1) вибір та визначення критеріїв оптимізації;
- 2) визначення проектних параметрів;
- 3) побудова цільової функції (моделі залежності між критеріями оцінки та проектними параметрами);
- 4) вибір методу розв'язання оптимізаційної задачі.

Для вирішення такої оптимізаційної задачі як відбір кращого банку для розміщення грошового вкладу вважаємо необхідним обрати наступні критерії оцінки: депозитна ставка, розмір депозитного портфеля, довгострокова динаміка депозитів населенню, відповідність капіталу активам, ліквідність банку, динаміка власного капіталу за півріччя, динаміка депозитів населення за 12 місяців, рейтинг прозорості.

Визначемо та надамо коротку характеристику обраних критеріїв.

Першим критерієм є депозитна ставка за депозитом. Під депозитною ставкою мається на увазі ставка, яку банк виплачує власнику депозиту [18, с. 82]. Розмір депозитної ставки визначається депозитним договором (угодою) і залежить від строку депозиту (терміновий, до запитання), виду валюти, суми депозиту та суми мінімального поповнення, виду вкладника (фізична особа, суб'єкт господарювання), а також від порядку виплати доходу. Проценти на банківський вклад нараховуються від дня, наступного за днем надходження вкладу в банк, до дня, який передуює його поверненню вкладникові або списанню з рахунка вкладника з інших підстав.

Наступним критерієм є розмір депозитного портфелю, під яким слід розуміти суму всіх окремо взятих депозитів та нарахованих по них відсотках. Розмір депозитного портфелю характеризує ступінь довіри клієнтів до банку [19, с. 662].

Також до критеріїв відбору кращого банку для розміщення депозиту запропоновано віднести довгострокову динаміку депозитів населенню, динаміку власного капіталу за півріччя та динаміку депозитів населення за 12 місяців. Під динамікою власного капіталу слід розуміти власного капіталу розуміємо приріст (зменшення) грошових коштів і вираженої у грошовій формі частини майна, які належать його власникам, забезпечують економічну самостійність і фінансову стійкість банку, використовуються для здійснення банківських операцій та надання послуг з метою одержання прибутку [18, с. 83]. Під динамікою депозитів слід розуміти приріст (зменшення) обсягів депозитів за певний календарний період (останні 12 місяців; 3 роки — довгострокова динаміка) [19, с. 662].

Наступним критерієм є ліквідність банку. Ліквідність є спроможність окремої банківської установи в цілому забезпечувати своєчасність, повноту та безперервність виконання усіх своїх грошових зобов'язань [20, с. 633].

Також до обраних критеріїв було обрано рівень прозорості банку. Рівень прозорості банку розглядається крізь призму відкритості фінансової звітності, банківських послуг, менеджменту, власників та реагування на запити преси.

Проектними параметрами у даній роботі є конкретні фінансові показники трьох найбільших банків України.

Для побудови цільової функції залежності було запропоновано скористатися методом найменших квадратів. Сутність цього методу полягає у знаходженні таких значень матриці параметрів А моделі, при яких сума квадратів залишків була б мінімальною [21, с. 53]. Для знаходження кращого банку для внесення грошових коштів фізичною особою було вирішено обрати метод максимінної згортки. Сутність принципу максиміна полягає в наступному. При проектуванні складних технічних систем і наявності великого числа власних

критеріїв встановити між ними аналітичний взаємозв'язок дуже складно, а в більшості випадків і неможливо. З використанням цього методу вирішується данна проблема завдяки знаходженню таких значень проектних параметрів, за яких нормовані значення всіх власних критеріїв рівні між собою.

2.2. Метод максимінної згортки з урахуванням різних ваг критеріїв

Основна ідея даного підходу полягає в тому, що власні критерії $f(x)$, $i=1, n$ поєднуються в один інтегральний критерій $F(x) = \Phi (f_1(x), f_2(x), \dots, f_n(x))$, а потім знаходиться максимум чи мінімум цієї цільової функції.

Якщо об'єднання власних критеріїв будується, виходячи з об'єктного взаємозв'язку власних критеріїв і критерію узагальненого, то тоді оптимальне розв'язання оптимізаційної задачі буде коректне. Але таке об'єднання здійснити вкрай складно чи неможливо, тому, як правило, узагальнений критерій є результатом чисто формального об'єднання власних (часткових) критеріїв оптимальності[22, с. 14].

В залежності від того, яким чином власні критерії поєднуються в узагальнений критерій оптимальності розрізняють наступні види узагальнених критеріїв: адитивний критерій; мультиплікативний критерій та максимінний (мінімаксий) критерій.

Маючи декілька рішень, кожне з яких отримано на основі врахування багатьох критеріїв оптимізації, необхідно їх порівняти між собою, або надати перевагу одному з них. Для розв'язання цієї задачі будується функція корисності, яка дає змогу визначити показник ефективності рішення і процес надання переваги зводиться до порівняння чисел-значень.

При цьому особа, що приймає рішення, враховує, що один набір значень локальних критеріїв володіє перевагою над іншими, якщо йому відповідає більше значення функції переваги.

Максимінний (мінімаксий) критерій працює за принципом компромісу, який ґрунтується на ідеї рівномірності. Сутність принципу максиміна полягає в

наступному. При проектуванні складних технічних систем і наявності великого числа власних критеріїв встановити між ними аналітичний взаємозв'язок дуже складно, а в більшості випадків і неможливо. Тому намагаються знайти такі значення проектних (параметрів) $x = \{x^1, x^2, \dots, x^m\}$, при яких нормовані значення всіх власних критеріїв рівні між собою $a f(X) = K$, де $a f(X)$ - нормоване значення i -го часткового критерію; K — константа[23, с.120].

При великій кількості власних критеріїв через складні взаємозв'язки між елементами складної системи, домогтися виконання зазначеного вище співвідношення дуже складно. Тому, на практиці, так варіюють значення проектних змінних проектування $x_1, x_2, \dots, x_m$, при яких послідовно «підтягуються» ті нормативні критерії, чисельні значення яких у вихідному рішенні виявилися найменшими. Оскільки ця операція виробляється в області компромісу, підтягування “відстаючого” критерію неминуче призводить до зниження значень частини інших критеріїв. Але при проведенні ряду кроків можна домогтися визначеного ступеня зрівноважування суперечливих власних критеріїв, що і є метою принципу максиміна[24, с.45].

Формально принцип максиміна формулюється в такий спосіб: вибрати такий набір змінних $X^{(0)} \in X$, при якому реалізується максимум з мінімальних нормованих значень власних критеріїв, тобто $F(X^{(0)}) = \max \min f_i(X)$.

Такий принцип вибору $X^{(0)}$ іноді називають гарантований результат. Він запозичений з теорії ігор, де є основним принципом. Якщо часткові критерії необхідно мінімізувати, то найбільш відстаючим критерієм є той, який приймає максимальне значення. В цьому випадку застосовують принцип мінімакса: $F(X^{(0)}) = \min \max f_i(X)$.

Перші кроки застосування мінімаксного підходу подібні до застосування максимінного підходу. Тобто, необхідно отримати значення часткових критеріїв оптимізації для кожної альтернативи і звести до безрозмірного виду шляхом ділення на номінальне значення кожного критерію оптимальності. Далі для кожної альтернативи обирається максимальне значення безрозмірного значення критерію.

Наступний крок алгоритму передбачає вибір мінімального значення з отриманих максимальних значень.

Отже, введемо переваги та недоліки використання методів на основі згорток для розв'язання задач багатокритеріальної оптимізації.

Так перевагами методу максимінної згортки є:

- 1) широка застосовуваність в процесі автоматизованого проектування;
- 2) наявність можливості максимізувати запас по працездатності технічного вибору;
- 3) наявність можливості досліджувати область слабо ефективних рішень (оптимальних за Слейтером) шляхом зміни значень вагових коефіцієнтів.

Основний недолік зазначеного методу полягає в тому, що в процесі розв'язання задач багатокритеріальної оптимізації доводиться вирішувати багато однокритеріальних задач, які є складними та нелінійними і інколи зробити це практично неможливо.

2.3. Аналітичний апарат розрахунку фінансових показників банківської діяльності

На сьогодні аналіз фінансових показників діяльності банку є обов'язковим елементом функціонування фінансової системи будь-якої країни. В свою чергу належне підтвердження фінансової стійкості є складовим елементом моніторингу фінансового стану банку. Серед основних сучасних підходів до організації моніторингу фінансового стану банків можна віднести наступні: 1) коефіцієнтний аналіз й аналіз однорідних груп (Німеччина, США); 2) рейтингові системи оцінки, що передбачають ранжування банків (США, Італія, Франція, Росія, Україна, Польща); 3) комплексні оцінки банківського ризику (Великобританія, Нідерланди); 4) статистичні моделі (системи «попереднього реагування») (США); 5) макропруденційний аналіз (країни – члени МВФ) [25, с. 67].

Практично всі системи оцінки є своєрідними модифікаціями рейтингових систем оцінки банків. Кожна система має чіткий інструментарій, критерії,

рекомендації щодо оцінювання фінансової стійкості, що забезпечує можливість формування повних та достовірних висновків про стан кожного елементу системи.

В розрізі даного наукового дослідження особливу увагу слід приділити таким фінансовим показникам, як розмір депозитного портфелю, довгострокова динаміка депозитів населенню, динаміка депозитів населенню за 12 місяців, відповідність капіталу активам, динаміка власного капіталу за півріччя, ліквідність банку та рейтинг його прозорості. Всі вони в сукупності дозволяють фізичній особі зробити висновок щодо доцільності розміщення депозиту в конкретному банку.

Якщо депозитна ставка як чіткий числовий показник є абсолютною величиною, яка визначається на підставі рішення відповідного органу банку, то всі інші показники, які визначені як критерії оптимізації в даній роботі, є такими, значення яких є результатом певних математичних розрахунків та фінансового аналізу.

Аналітичним апаратом розрахунку кожного з вищезазначених показників є матеріали фінансової звітності кожного окремого банку. Обрахування кожного з взятих в даній роботі критеріїв оптимізації здійснюється на підставі конкретних фінансових показників банківської діяльності по наявним формулам. Результати цих обрахунів відзначаються у фінансовій звітності банку за певний період, тобто є такими показниками, які потребують систематичного оновлення за певний період (півріччя або рік).

Розглянемо більш детально особливості розрахунку числового значення кожного критерію оптимізації.

Найбільш простим у визначенні є розмір депозитного портфелю банку, адже являє собою суму всіх окремо взятих депозитів та нарахованих по них відсотках. І визначається він у мільйонах гривень. Під депозитним портфелем в науці розуміють сукупність коштів на депозитних рахунках клієнтів, залучених банком на договірній основі. Формування депозитного портфеля слід розглядати як безперервний циклічний процес, що складається з основних етапів:

- аналіз (дослідження «поведінки» грошових коштів на рахунках клієнтів);
- планування (орієнтація на залучення певних груп клієнтів, і види депозитів);
- робота по залученню клієнтів (впровадження нових продуктів, гнучка тарифна політика і індивідуальна робота з клієнтами);
- контроль[26].

Все це зумовлює постійну змінюваність розміру депозитного портфелю банку. Але з точки зору визначення цього параметра для відбору кращого банку для розміщення депозиту важливою є чітка величина, визначувана в мільйонах гривень, на кінець певного звітного періоду, а саме за попередній квартал діяльності, адже саме квартал є мінімальним звітним періодом для банків.

Так, наприклад розмір депозитного портфелю Приватбанку можна відобразити за допомогою наступних графіків[27] (рис. 2.1), (рис. 2.2):



Рис. 2.1. Депозитний портфель Приватбанку за вкладами в національній валюті.

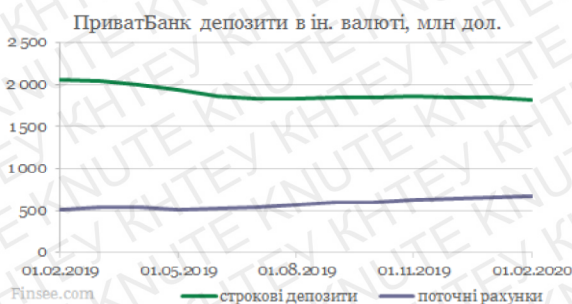


Рис. 2.2 Депозитний портфель Приватбанку за вкладами в іноземній валюті.

Відповідно до офіційних даних станом на 01 лютого 2020 року Приватбанком всього залучено коштів клієнтів: 235,2 млрд грн в еквіваленті, а це перше місце в Україні серед всіх банківських установ, в т.ч.

гривневі кошти фізичних осіб – 123,3 млрд грн – 1 місце;

валютні кошти фізичних осіб – 2,5 млрд дол. США в еквіваленті – 1 місце;

кошти юридичних осіб – 48,4 млрд грн в еквіваленті – 3 місце.

Наступними критеріями оптимізації, використаними в даній роботі, є довгострокова динаміка депозитів населенню, динаміка депозитів населення за 12 місяців та динаміка власного капіталу за півріччя. Всі вони мають відсоткове значення, та визначаються на підставі співвідношення певного показника за різні проміжки часу. Важливим є те, що будь-яка динаміка як показник може мати позитивне значення, коли мова йде про приріст, та негативне, тобто значення зі знаком "-", коли мова йде про падіння певного показника за наступний проміжок часу порівняно із попереднім.

Це можна побачити на прикладі динаміки депозитів населення за останні 10 років за сумарними показниками всіх банків України(рис. 2.3).



Рис 2.3. Динаміка за видами вкладів

Цікавим є те, що динаміка короткострокових депозитів є позитивною, а от динаміка довгострокових депозитів (більше 1 року) на жаль є негативною. В

розрізі даного наукового дослідження всі динамічні показники розраховуються щодо кожного окремого банку.

Динаміка депозитів населення за 12 місяців є показником, який розраховується самим банком та підлягає офіційному опублікуванню.

Знову ж на прикладі Приватбанку слід зазначити, що відповідно до даних офіційної звітності самого банку строкові гривневі депозити ПриватБанку протягом року зросли лише на 7,31% (рис. 1). Збільшення залишків на поточних рахунках було суттєвішим – 25.76 %. При цьому з початку 2020 року це зростання сповільнилось.

Що стосується динаміки валютних депозитів, то за минулий рік Приватбанк втратив 11,57% строкових депозитів у валюті (рис. 2). Водночас клієнти набагато активніше почали розміщувати валютні кошти на вимогу. Вони за аналогічний період зросли майже на третину.

Наступним критерієм оптимізації є динаміка власного капіталу банку. Так на прикладі Приватбанку розрахуємо цей показник. Оскільки для визначення кращого банку для розміщення депозиту фізичною особою важливим є приріст (чи навпаки зменшення) власного капіталу банку як основи гарантованості повернення вкладу, то обрахування динаміки власного капіталу за останнє півріччя як найбільш інформативний період для цього показника є необхідним, виходячи з офіційних статистичних даних, що обнародуються самим банками.

Отже, динаміка власного капіталу Приватбанку за останнє півріччя становить 42,27%(рис. 2.4).

Період	Показник, тис. грн.	Зміна, тис. грн.
I квартал 2020	64 706 237	+10 177 306 ↑
IV квартал 2019	54 528 931	+9 048 410 ↑
III квартал 2019	45 480 521	+7 575 621 ↑

Рис 2.4. Динаміка власного капіталу

Наступним критерієм є ліквідність банку, яку слід розглядати як здатність банку виконувати свої зобов'язання не тільки своєчасно, але й без суттєвих втрат.

Можливість забезпечення ліквідності банку пов'язана з наявністю надійної капітальної бази; високою якістю активів і депозитів, їх збалансованістю за строками; помірною залежністю від зовнішніх джерел; позитивним іміджем банку; високим рівнем менеджменту тощо.

Ліквідність банку – це його спроможність забезпечувати своєчасне виконання своїх грошових зобов'язань, завдяки збалансованості між строками та сумами погашення розміщених активів і строками та сумами виконання зобов'язань, а також строками та сумами інших джерел і напрямів використання коштів. З метою підтримання ліквідності банк повинен мати певний резерв коштів для виконання непередбачених зобов'язань, які можуть бути обумовлені як змінами стану грошового ринку, так і фінансовим станом клієнтів або банків-партнерів[29].

І нарешті такий критерій як рейтинг прозорості ґрунтується на комплексному аналізі доступності, повноти та актуальності найважливішої, з точки зору клієнта, банківської інформації. В ході дослідження з метою рейтингування всебічно оцінюють відкритість фінансової звітності, послуг, менеджменту, структури власності в інтернет-мережі, а також здатність банків оперативно реагувати на журналістські *e-mail*-запити. Цей рейтинг проводиться, як правило, найбільш впливовими економічними виданнями країни[28].

2.4. Методика формування матриці попарного порівняння критеріїв та розрахунку їх вагових коефіцієнтів

Найбільш поширеними є експертні методи встановлення ваг показників. Їх перевага – простота застосування, недолік – можлива недостовірність даних, певна суб'єктивність, трудомісткість збору та обробки результатів.

На сьогодні існує багато методів обробки результатів експертиз (методів експертного оцінювання та прогнозування): попарних зіставлень (порівнянь); надання переваг; рангів; нечітких експертних оцінок; аналізу ієрархій та їх удосконалені чи узагальнені варіанти. Крім того, існують методи організації

проведення експертиз (інтерв'ю, анкетування, метод Делфі, метод комісій та ін.) і методи визначення компетентності експертів та вагомості їх оцінок. Нижче наведено поетапно один з таких методів[30, с. 10].

Етап 1. Організатори експертизи кожному експерту виставляють за певним набором критеріїв деяку оцінку за бальною шкалою, не обов'язково однаковою для всіх критеріїв. Такими критеріями можуть бути, наприклад, фах за освітою з точки зору галузі дослідження, рівень освіти, стаж роботи в галузі, стаж роботи в досліджуваному регіоні, досвід участі в подібних експертизах, рівень ознайомленості з методиками дослідження тощо.

Етап 2. Обчислюється сума балів за всіма критеріями для кожного експерта C_i ($i = 1, 2, 3, \dots, N$), де N – кількість експертів.⁴²

Етап 3. Для кожного експерта визначається відносне значення C_i^* за формулою:

$$C_i^* = C_i / C_{\max} \quad (2.1)$$

де $C_{\max}$ – найбільша сумарна кількість балів, яку може набрати експерт.

Описаний метод визначає компетентність експертів шляхом їх взаємної оцінки. Досить часто використовують також методи самооцінки. За допомогою першого методу дуже часто отримується інформація про конфронтації та коаліції між експертами, за допомогою інших – інформація про рівень самовпевненості експерта. Саме з цих причин недоцільно обмежуватися лише взаємною оцінкою при визначенні компетентності експертів.

Визначення вагових коефіцієнтів на основі попарних порівнянь. Докладна числова інформація про досліджувані об'єкти не є обов'язковою, а порівняння можна здійснювати за якісними критеріями: “більше-менше”, “краще-гірше”, без уточнення, у скільки разів або на скільки більше чи краще. Метод попарних порівнянь було розроблено американським математиком Т. Сааті. Кожен показник, підгрупа або група показників співставляються з іншими (показниками,

підгрупами та групами відповідно) на основі шкали відносної значущості (табл. 2.1).

Експерти, виконуючи порівняння показників, заповнюють таблицю попарних порівнянь. Оскільки значення показників невідомі заздалегідь, порівняння виконується на основі суб'єктивних суджень, які чисельно оцінюються за шкалою. Необхідно відмітити, що кожна клітинка таблиці містить не одне число (безпосередню оцінку експерта), а сукупність чисел, з урахуванням того, що порівняння показників проводиться опосередковано через порівняння їх з певним допоміжним фактором. Врахування цих додаткових порівнянь дозволяє підвищити надійність отриманих результатів, або зменшити кількість необхідних експертів[30, с. 30].

Таблиця 2.1 Шкала відносної значущості показник

Значення	Відповідність характеристик	Пояснення
1	Рівноцінність	Однакове значення двох характеристик
2	Помірна перевага одного над іншим	Досвід і думка експерта дають помірну перевагу однієї характеристики над іншою
3	Істотна перевага	Досвід і думка експерта дають істотну перевагу однієї характеристики над іншим
4	Значна перевага	Перевага однієї характеристики над іншою значна
5	Сильна перевага	Однозначна перевага однієї характеристики над іншою

Результатом порівняння є певне числове значення: показник, який має перевагу, оцінюється цілим числом (a_{ij}), протилежний – оберненим до нього ($1/a_{ij}$). Наприклад, якщо показник 1 за рішенням експерта має “помірну перевагу” над показником 2, то перший оцінюється числом $a_{12} = 2$, а другий – $a_{21}=1/2$. Отримані значення записуються у вигляді матриці попарних порівнянь (табл. 2.2). Таблиця заповнюється по рядках (при порівнянні показників 1 з 2, оцінка 2 ставиться на перетині рядка 1 і стовпця 2, а оцінка 1/2 – на перетині рядка 2 і стовпця 1). На

діагоналі заповненої таблиці завжди будуть стояти одиниці – порівняння показника самого з собою буде давати рівноцінність.

Таблиця 2.2 Матриця попарних порівнянь показників

№ характеристик	1	2	3	4
1	1	a12	a13	a14
2	$a_{21}=1/a_{12}$	1	a23	a24
3	$a_{31}=1/a_{13}$	$a_{32} = 1/a_{23}$	1	a34
4	$a_{41}=1/a_{14}$	$a_{42} = 1/a_{24}$	$a_{43} = 1/a_{34}$	1

Таблиця містить $n \times n$ оцінок – кожна характеристика має декілька оцінок, число яких залежить від кількості альтернатив, за якими проводиться порівняння. Задача визначення вагових коефіцієнтів показників зводиться до знаходження головного власного вектора матриці.

Отже, перевагами методу попарних порівнянь та його модифікацій є: 1) простота у формуванні початкових матриць; 2) чітке математичне обґрунтування здійснюваних операцій; 3) можливість переходу до інших представлень експертної інформації (ранжування, еталонні оцінки тощо). До недоліків цього методу можна віднести громіздкість: при збільшенні числа показників буде зростати число одиничних пар порівнянь.

2.5. Формування функцій приналежності на основі методу найменших квадратів

Перший чіткий і стислий виклад методу найменших квадратів був опублікований Легендером у 1805 р. Ця методика описується як алгебраїчна процедура пристосування лінійних рівнянь до даних, а Легендер демонструє новий метод, аналізуючи ті самі дані, що і Лаплас для форми землі. Значення методу Легендера про найменші квадрати негайно визнали провідні астрономи та геодезисти того часу.

У 1809 р. Карл Фрідріх Гаус опублікував свій метод обчислення орбіт небесних тіл. У цій роботі він стверджував, що володіє методом найменших квадратів з 1795 року. Це, природно, призвело до пріоритетної суперечки з

Легендером. Однак, на заслугу Гаусса, він вийшов за межі Легендера і досяг успіху в поєднанні методу найменших квадратів з принципами ймовірності та нормального розподілу. Йому вдалося виконати програму Лапласа щодо визначення математичної форми щільності ймовірності спостережень, залежно від кінцевої кількості невідомих параметрів, і визначити метод оцінки, який мінімізує помилку оцінки. Гаусс показав, що середнє арифметичне дійсно є найкращою оцінкою параметра розташування шляхом зміни як щільності ймовірності, так і методу оцінки. Потім він вирішив проблему, запитавши, яку форму має мати щільність і який метод оцінки слід використовувати для отримання середнього арифметичного в якості оцінки параметра розташування. У цій спробі він винайшов нормальний розподіл.

Після цього коротенького вступу перейдемо до викладу безпосередньо методу найменших квадратів (далі - МНК). Нехай відомо, що вихідний параметр процесу, який вивчається, позначимо його y , лінійно залежить від вхідного параметра x (суцільна пряма лінія на рис. 2.5.).

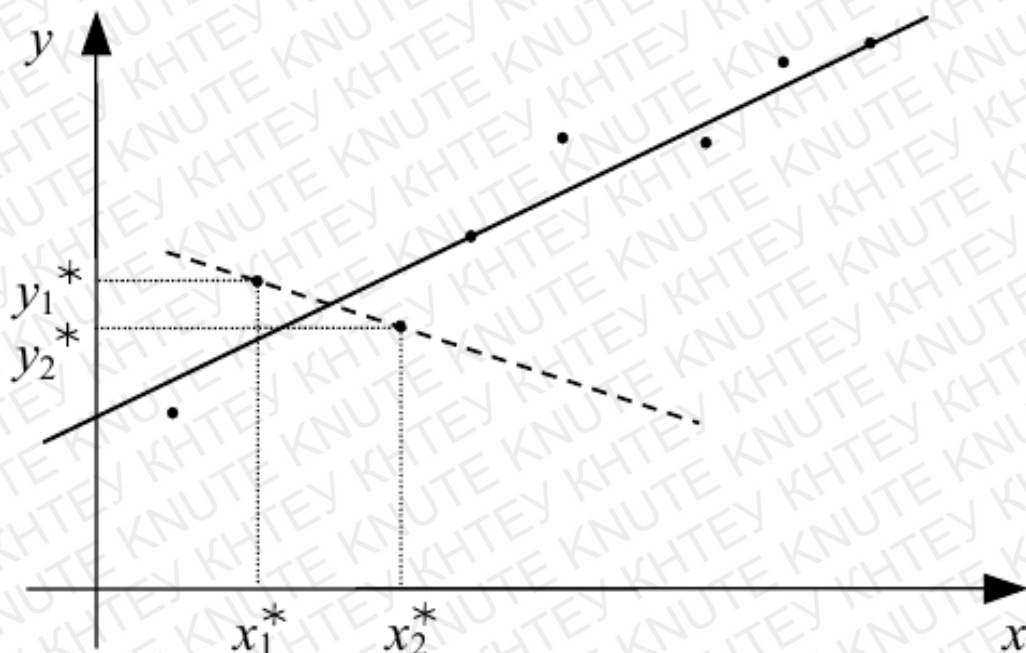


Рис. 2.5. Графік функції $F(x)$

Тобто припустимо, що статична характеристика цього процесу може бути подана у вигляді

$$y = ax + b, \quad (2.2)$$

Де a і b — коефіцієнти, для визначення числових значень яких необхідно, як мінімум, задати два значення x_1, x_2 вхідній величині x і заміряти відповідні їм значення y_1, y_2 вихідної величини y , оскільки лише під час виконання цих умов для моделі (2.2) можна скласти систему двох алгебраїчних рівнянь із двома невідомими a і b

$$\begin{cases} y_1 = ax_1 + b, \\ y_2 = ax_2 + b. \end{cases} \quad (2.3)$$

Але результати будь-яких експериментальних вимірювань несуть у собі похибки, обумовлені класом точності вимірювальних засобів, дією різноманітних завад, неточністю зчитування показів приладів, округленням під час приведення даних до однакових умов обробки інформації — список умов виникнення похибок можна продовжити, але для обґрунтування МНК цього досить.

Тож через наявність цих похибок в експериментальних значеннях x_1, x_2, y_1, y_2 безпосередній розв'язок системи рівнянь відносно a та b може нести в собі похибку в 10, 100, 1000 і більше відсотків.

Наприклад, якщо використати лише значення $x_1^*, y_1^*; x_2^*, y_2^*$ для розв'язання системи рівнянь (2.3), то похибка буде вже не у відсотках, а у характері функціональної залежності (пунктирна лінія на рис. 2.5).

У свій час Гаусс запропонував інший спосіб визначення коефіцієнтів a, b моделі (2.2). Він запропонував сформулювати суму квадратів різниць $\sum N$ між теоретично заданими за допомогою рівняння (2.2) значеннями вихідної координати y при значеннях аргументу $x_i, i = 1, N$ та її експериментальними значеннями y_i :

$$\Sigma^N = \sum_{i=1}^N (y(x_i) - y_i)^2, \quad (2.4)$$

а потім знайти такі значення коефіцієнтів a , b рівняння, котрі мінімізують вираз (2.4).

Від цієї процедури і назва методу — метод найменших квадратів.

З курсу математичного аналізу відомо, що для знаходження мінімуму якоїсь функції необхідно взяти від неї похідну, прирівняти цю похідну до нуля і розв'язати отримане рівняння — його корінь задає значення аргументу, за якого функція досягає мінімуму, а само значення функції у цій точці, якщо вона опукла донизу, задає її мінімальне значення.

Згідно з цією ідеєю, підставимо у вираз (2.4) замість $y(x_i)$ його значення з (2.2) і візьмемо від отриманого виразу частинні похідні за b та a , які прирівняємо до нуля, тобто

$$\Sigma^N = \sum_{i=1}^N (ax_i + b - y_i)^2, \quad (2.5)$$

$$\begin{cases} \frac{\partial \Sigma^N}{\partial b} = \sum_{i=1}^N 2 (ax_i + b - y_i)(1) = 0, \\ \frac{\partial \Sigma^N}{\partial a} = \sum_{i=1}^N 2 (ax_i + b - y_i)(x_i) = 0. \end{cases} \quad (2.6)$$

Із (2.6) після низки нескладних перетворень отримаємо:

$$\begin{cases} b \cdot N + a \sum_{i=1}^N x_i = \sum_{i=1}^N y_i, \\ b \sum_{i=1}^N x_i + a \sum_{i=1}^N x_i^2 = \sum_{i=1}^N y_i x_i. \end{cases} \quad (2.7)$$

Розв'язавши систему рівнянь (2.7) відносно b і a , отримаємо такі їх значення, які мінімізують суму квадратів відхилень експериментально вимірених значень величин x_i , y_i від теоретично заданих згідно з вибраною функціональною залежністю.

Рівняння, що входять у систему (2.7), називають нормальними рівняннями Гаусса. Коефіцієнтами у них є суми, які «згладжують» дію похибок вимірювань величин x , y і зменшують їх вплив на оцінки параметрів b , a . Завдяки цьому підвищується точність їх визначення.

А тепер припустимо, що поле точок (x_i, y_i) експериментально визначених величин x , y має такий вигляд, як це показано на рис. 3.2.

Із цього рисунка видно, що середньою лінією цього поля, яка віддзеркалює «в середньому» функціональну залежність y від x , є парабола

$$\begin{cases} \frac{\partial \Sigma^N}{\partial b} = \sum_{i=1}^N 2 (ax_i + b - y_i)(1) = 0, \\ \frac{\partial \Sigma^N}{\partial a} = \sum_{i=1}^N 2 (ax_i + b - y_i)(x_i) = 0. \end{cases} \quad (2.8)$$

$$y = ax^2 + bx + c,$$

параметри якої a , b , c також доцільно визначати за допомогою МНК.

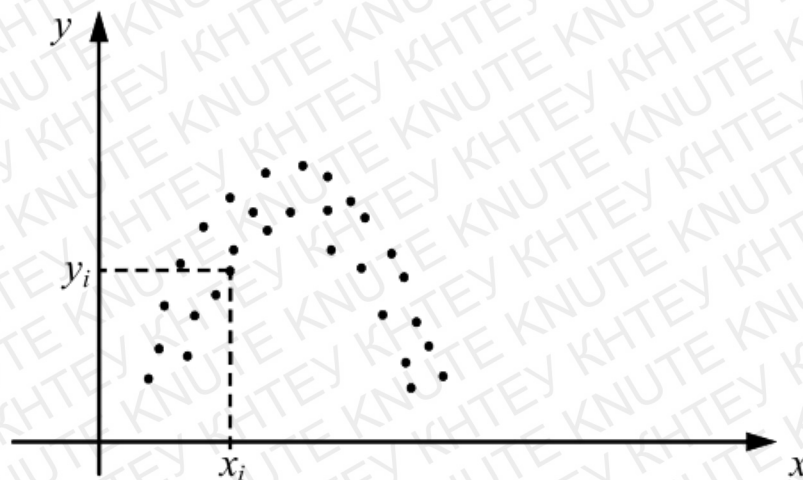


Рисунок 3.2 — Поле точок (x_i, y_i) експериментально визначених величин x , y

Для отримання нормальних рівнянь Гаусса у цьому випадку підставимо (2.8) у (2.4), що дасть вираз

$$\Sigma^N = \sum_{i=1}^N \left(ax_i^2 + bx_i + c - y_i \right)^2. \quad (2.9)$$

Далі від цього виразу візьмемо частинні похідні за c , b , a та прирівняємо їх нулю, що дасть систему рівнянь:

$$\begin{cases} \frac{\partial \Sigma^N}{\partial c} = \sum_{i=1}^N 2 \left(ax_i^2 + bx_i + c - y_i \right) \cdot (1) = 0, \\ \frac{\partial \Sigma^N}{\partial b} = \sum_{i=1}^N 2 \left(ax_i^2 + bx_i + c - y_i \right) \cdot (x_i) = 0, \\ \frac{\partial \Sigma^N}{\partial a} = \sum_{i=1}^N 2 \left(ax_i^2 + bx_i + c - y_i \right) \cdot (x_i^2) = 0. \end{cases} \quad (2.10)$$

Після спрощень в системі (2.10) матимемо систему нормальних рівнянь Гаусса (2.11), розв'язавши яку отримаємо оптимальні за критерієм мінімуму відхилень експерименту від теорії значення параметрів c , b , a математичної моделі (2.8) функціональної залежності величини y від величини x , заданих експериментальним полем точок (x_i, y_i) :

$$\begin{cases} cN + b \sum_{i=1}^N x_i + a \sum_{i=1}^N x_i^2 = \sum_{i=1}^N y_i, \\ c \sum_{i=1}^N x_i + b \sum_{i=1}^N x_i^2 + a \sum_{i=1}^N x_i^3 = \sum_{i=1}^N y_i x_i, \\ c \sum_{i=1}^N x_i^2 + b \sum_{i=1}^N x_i^3 + a \sum_{i=1}^N x_i^4 = \sum_{i=1}^N y_i x_i^2. \end{cases} \quad (2.11)$$

З усього викладеного у даному підрозділі можна зробити такі висновки:

- 1) за вибраної структури моделі функціональної залежності оптимальні значення її параметрів у межах заданого діапазону значень аргументу та його функції найбільш просто і ефективно визначаються за допомогою МНК;
- 2) вибір виду та структури моделі є прерогативою дослідника;
- 3) у класі заданих структур МНК дозволяє отримати не лише оптимальні значення коефіцієнтів, але й оптимальну структуру моделі.

ВИСНОВКИ ЗА РОЗДІЛОМ 2.

Виділено наступні етапи процесу побудови оптимізаційної задачі з відбору кращого банку для розміщення депозиту фізичними особами: 1) вибір та визначення критеріїв оптимізації; 2) визначення проектних параметрів; 3) побудова цільової функції (моделі залежності між критеріями оцінки та проектними параметрами); 4) вибір методу розв'язання оптимізаційної задачі.

Надано коротку характеристику кожного з критеріїв оптимізації (а саме депозитної ставки, розміру депозитного портфеля, довгострокової динаміки депозитів населенню, відповідності капіталу активам, ліквідності банку, динаміки власного капіталу за півріччя, динаміки депозитів населення за 12 місяців, рейтингу прозорості).

Для знаходження кращого банку для внесення грошових коштів фізичною особою було вирішено обрати метод максимінної згортки. Сутність принципу максіміна полягає в наступному. При проектуванні складних технічних систем і наявності великого числа власних критеріїв встановити між ними аналітичний взаємозв'язок дуже складно, а в більшості випадків і неможливо. З використанням цього методу вирішується дана проблема завдяки знаходженню таких значень проектних параметрів, за яких нормовані значення всіх власних критеріїв рівні між собою.

Виділено наступні переваги методу максимінної згортки при вирішенні задачі відбору кращого банку для розміщення клієнтом грошових коштів:

- 1) широка застосовність в процесі автоматизованого проектування;

2) наявність можливості максимізувати запас по працездатності програмного додатку;

З'ясовано, що обрані в роботі критерії оптимізації є або числовими, або лінгвістичними змінними. У зв'язку з цим використовується різні вихідні аналітичні дані. Якщо депозитна ставка як чіткий числовий показник є абсолютною величиною, яка визначається на підставі рішення відповідного органу банку, то всі інші показники, які визначені як критерії оптимізації в даній роботі, є такими, значення яких є результатом певних математичних розрахунків та фінансового аналізу. Аналітичним апаратом розрахунку кожного з вищезазначених показників є матеріали фінансової звітності кожного окремого банку.

Перевагами методу попарних порівнянь та його модифікацій є: 1) простота у формуванні початкових матриць; 2) чітке математичне обґрунтування здійснюваних операцій; 3) можливість переходу до інших представлень експертної інформації (ранжування, еталонні оцінки тощо). До недоліків цього методу можна віднести громіздкість: при збільшенні числа показників буде зростати число одиничних пар порівнянь.

За вибраної структури моделі функціональної залежності оптимальні значення її параметрів у межах заданого діапазону значень аргументу та його функції найбільш просто і ефективно визначаються за допомогою МНК. У класі заданих структур МНК дозволяє отримати не лише оптимальні значення коефіцієнтів, але й оптимальну структуру моделі.

РОЗДІЛ 3. РОЗРОБКА КОМП'ЮТЕРНОЇ СИСТЕМИ ОЦІНКИ ТА ВІДБОРУ БАНКІВ ДЛЯ РОЗМІЩЕННЯ КОШТІВ ФІЗИЧНИМИ ОСОБАМИ

3.1. Алгоритм реалізації методу максимінної згортки для критеріїв з різними вагами

У попередніх розділах були визначені критерії, за якими буде проведено відбір кращого банку для внесення грошових вкладів фізичними особами, використовуючи метод багатокритеріального аналізу, а саме максимінний. Отже, розберемо алгоритм використання максимінної згортки(рис. 3.1).

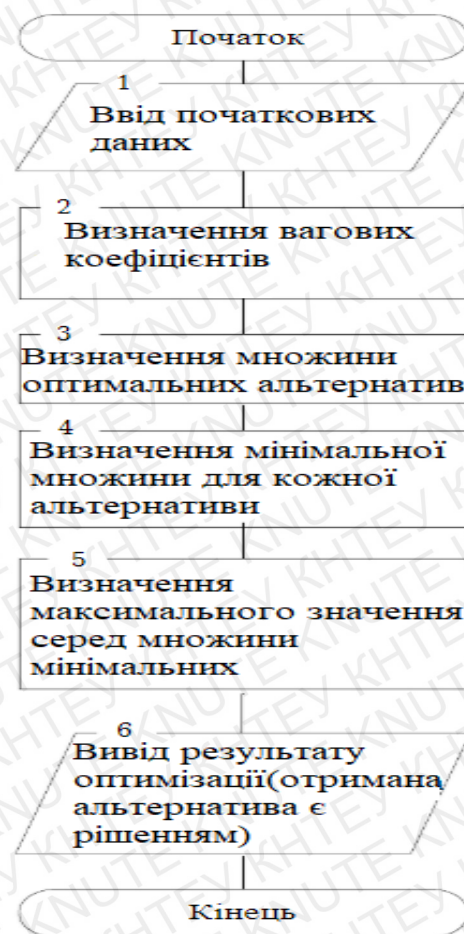


Рис.3.1. Блок-схема алгоритму реалізації максимінної згортки

На першому кроці алгоритму отримуємо входні дані(табл. 3.1).

Таблиця 3.1. Вхідні дані

Назва банку	ПриватБанк	ОщадБанк	Райффайзен Банк Аваль
розмір депозитного портфелю	235 200 млн грн	208 631 млн грн	79 190 млн грн
довгострокова динаміка депозитів населенню	12%	10%	8%
динаміка депозитів населення за 12 місяців	10%	8%	6%
ліквідність банку	B	B	A
відповідність капіталу активам	95%	98%	90%
динаміка власного капіталу за півріччя	17%	14%	15%
рейтинг прозорості	A	B	C
депозитна ставка	11%	10%	9%

У таблиці 3.1. вказані критерії відбору кращої альтернативи а також, в кожній комірці розміщено значення часткового критерія оптимальності для i -го варіанту рішення часткового критерія.

На наступному кроці необхідно визначити вагові коефіцієнти по кожному критерію. Алгоритм розрахунку вагових коефіцієнтів буде розглянуто в наступному пункті.

Тепер переходимо до наступного кроку. Множина оптимальних альтернатив B з урахуванням різної важливості критеріїв якості визначається шляхом перетину нечітких множин наступним чином(рис. 3.2.)

$$B = F_1^{\beta_1} \cap F_2^{\beta_2} \cap \dots \cap F_6^{\beta_6};$$

$$\mu_B(a^*) = \max \mu_B(a_j).$$

Рис. 3.2. Перетин нечітких множин

Знайдемо множину оптимальних альтернатив с урахуванням вагових критеріїв:

$$B = \{ \min \{ 0,9^{1,416}; 0,8^{0,536}; 0,9^{0,608}; 0,75^{2,616}; 0,58^{0,624}; 0,7^{1,168}; 0,9^{0,632}; 1,0^{0,4} \}$$

$$\min \{ 0,81^{1,416}; 0,7^{0,536}; 0,75^{0,608}; 0,75^{2,616}; 0,6^{0,624}; 0,57^{1,168}; 0,72^{0,632}; 0,75^{0,4} \}$$

$$\min \{ 0,73^{1,416}; 0,3^{0,536}; 0,6^{0,608}; 1^{2,616}; 0,55^{0,624}; 0,61^{1,168}; 0,54^{0,632}; 0,5^{0,4} \} \}.$$

Множина оптимальних варіантів B має наступний вигляд:

$$\max \mu_B(a_j) = \max(0,471; 0,471; 0,524)) \quad (3.1)$$

Таким чином, кращою альтернативою є банк Райффайзен Банк Аваль друге місце поділили між собою ПриватБанк та ОщадБанк.

3.2. Алгоритм розрахунку ваг критеріїв

У попередньому пункті під час розгляду алгоритму реалізації максимінної згортки необхідно було визначити вагові коефіцієнти критеріїв. У цьому пункті буде розглянуто алгоритм розрахунку ваг критеріїв. Одним із можливих способів отримання вагових коефіцієнтів полягає в побудові матриці попарних порівнянь критеріїв. Отже, надалі буде надано пошаговий алгоритм розрахунку вагових коефіцієнтів.

Розпочинаємо з побудови матриці для критеріїв, що були використані для вирішення задачі відбору кращого банку(табл.3.2).

Слід зауважити, що матриця попарних порівнянь у даному випадку є позитивно визначеною, зворотно-симетричною та має ранг, який дорівнює 1. Особливістю зворотно-симетричної матриці є те, що: на головній діагоналі завжди повинна знаходитися оцінка, що дорівнює 1 (рівна важливість напрямків), тобто $a_{ij} = a_{ji}$ а при $i = j$; завжди необхідно дотримуватися відношення, що відповідає умові: якщо при порівнянні i -го напрямку з j -м ставиться оцінка a_{ij} , то при

порівнянні j -го напрямку з i -м оцінка a_{ji} повинна бути зворотною a_{ij} , тобто $a_{ji} = 1 / a_{ij}$.

Таблиця 3.2. Матриця попарних порівнянь критеріїв

Вибір банку	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8
F_1	1	7	3	4	1/4	1/3	5	2
F_2	1/7	1	1	1/2	1/7	1/2	3	2
F_3	1/3	1	1	1/2	1/4	1/2	4	1
F_4	1/4	2	2	1	1/5	1	1/6	3
F_5	4	7	4	5	1	3	3	2
F_6	3	2	2	1	1/3	1	2	2
F_7	1/5	1/3	1/4	6	1/3	1/2	1	7
F_8	1/2	1/2	1	1/3	1/2	1/2	1/7	1

Робота експерта полягає у заповненні матриці парних порівнянь, шляхом попарного порівняння критеріїв, спираючись на суб'єктивні судження, які чисельно оцінюються за шкалою Сааті (табл. 3.3) та знаходженні вектору ваг напрямків зв'язку – вектору Сааті[31, с.67].

Таблиця 3.3. Шкала відносної важливості

Інт-сть важливості	Якісна оцінка	Пояснення
0	Незрівнянність	Немає сенсу порівнювати елементи
1	Рівна важливість елементів, що порівнюються	Елементи є рівними за значимістю
3	Помірна (слаба) перевага одного над іншим	Існують показання про перевагу одного елемента над іншим, але вони не є переконливими
5	Сильна (суттєва) перевага	Існують гарні докази і логічні критерії, котрі можуть показати, що елемент є більш важливим
7	Очевидна перевага	Існує переконливий доказ більшої значимості одного елемента над іншим
9	Абсолютна перевага	Максимально підтверджується відчутність одного елемента над іншим
2,4,6,8	Проміжні значення між двома сусідніми оцінками	Коли є необхідним компроміс

Наступним кроком знаходження вагових коефіцієнтів є визначення власного вектору матриці. Проте, процедура визначення власного вектору матриці є досить трудомісткою задачею, тому цілком доцільно вирішити задачу знаходження вагових коефіцієнтів напрямків зв'язку шляхом застосування методу розрахунку середнього геометричного відстані між оцінками об'єктів:

$$\hat{K}_i = \sqrt[n]{\prod_{j=1}^n a_{ij}} / \sum_{i=1}^n \sqrt[n]{\prod_{j=1}^n a_{ij}}, \quad (3.2)$$

де K_i - нормалізована оцінка j -го напрямку зв'язку, j – позначення напрямку у рядку матриці парних порівнянь; $i, j \in 1, n$. Отже, алгоритм обчислення середнього геометричного складається з наступних кроків [32]:

Крок 1. Знайти добуток елементів кожного рядку матриці.

Крок 2. Розрахувати корінь n -ої степені з кожного знайденого добутку.

Крок 3. Знайти суму елементів

$$\sum_{i=1}^n \sqrt[n]{\prod_{j=1}^n a_{ij}} \quad (3.3)$$

розрахованих значень

Крок 4. Поділити значення, що були отримані на кроці 2 на суму, яка була розрахована на кроці 3.

Сума всіх нормалізованих оцінок (вектору Сааті) дорівнює 1 [31, с. 69].

І останнім кроком алгоритму є множенням нормалізованих оцінок (вектору Сааті) на число критеріїв n (табл. 3.4).

Таблиця 3.4. Власний вектор матриці полярних порівнянь критеріїв та їх вагові коефіцієнти

	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8
Значення A	0,177	0,067	0,076	0,078	0,327	0,146	0,079	0,05
Значення B	1,416	0,536	0,608	0,624	2,616	1,168	0,632	0,4

Таким чином, було знайдено вагові коефіцієнти для кожного критерію

3.3. Характеристика програмного забезпечення комп'ютерної системи оцінки та вибору банків для розміщення коштів фізичними особами

Для реалізації програмного забезпечення багатокритеріальної оцінки та відбору банків для розміщення грошових вкладів фізичними особами було обрано мову програмування C#.

C# - це мова програмування, що була розроблена компанією Microsoft, як основна мова розробки для платформи .NET. Це дозволяє створювати та компілювати програми навіть без середовищ розробки, наприклад Visual Studio. Мова має строгу статичну типізацію, підтримує поліморфізм, перевантаження операторів, вказівники на функції-члени класів, атрибути, події, властивості, винятки, коментарі у форматі XML.

C# є мовою програмування з C-подібним синтаксисом. Нововведенням C# стала можливість легшої взаємодії, порівняно з мовами-попередниками, з кодом програм, написаних на інших мовах, що є важливим при створенні великих проєктів. Якщо програми на різних мовах виконуються на платформі .NET, .NET вирішує проблему щодо сумісності програм. Також C# подолав проблему множинного наслідування та виводу типів.

Середовищем для розробки додатку було обрано Visual Studio. До переваг відносно інших середовищ можна віднести: можливість налагодження, підтримка безлічі мов програмування при розробці та інтуїтивний стиль кодування. Програма була розроблена застосовуючи технологію WPF. Windows Presentation Foundation (WPF) - це платформа для користувача інтерфейсу для створення клієнтських додатків для настільних систем. Платформа розробки WPF підтримує широкий набір компонентів для розробки додатків, включаючи модель додатки, ресурси, елементи управління, графіку, макет, прив'язки даних, документи і безпеку. Ця платформа є частиною платформи .NET.

У процесі розробки архітектури додатку було використано патерн MVVM(MODEL-VIEW-VIEWMODEL) (Рис.3.3). Адже побудувати призначений для користувача інтерфейс (UI), не використовуючи зазначений вище патерн, було

б майже неможливим. У такому разі було б взято форму, додано на неї віджети, а логіку написано в коді. Такий код, що описує логіку View, жорстко пов'язаний з призначенням для користувача інтерфейсом, так як він безпосередньо взаємодіє з елементами на екрані. Це хороший, але прямолінійний підхід. Він застосовується лише для дуже простих інтерфейсів. Коли логіка стає складнішою, підтримка такого UI стає великою проблемою.

Корінь проблеми полягає в тому, що побудова UI таким способом порушує принцип єдиної відповідальності (single responsibility principle), який говорить: «У класу повинна бути тільки одна причина для зміни». Якщо UI-компонент містить код для відображення, логіки і даних, то у нього є кілька причин для зміни. Наприклад, потрібно змінити тип призначеного для користувача елемента, який використовується для відображення даних, то зміни не повинні вплинути на логіку. Однак оскільки логіка так тісно пов'язана з елементами управління, її теж доведеться міняти. Це є порушенням принципу єдиної відповідальності. Таким чином, якщо форма містить код для відображення елементів управління, логіку інтерфейсу (що відбувається при натисканні кнопки) і дані для відображення на екрані, ви зіткнетеся з такими проблемами:

- ускладнення підтримки;
- зміни в UI, логікою чи даних, швидше за все, спричиняють за собою зміни в інших частинах. Тому вносити правки набагато складніше, що ускладнює підтримку;
- погіршення тестованості;
- логіка і дані програми можуть бути написані таким чином, щоб кожен компонент міг бути протестований окремо. Однак код, пов'язаний з призначенням для користувача інтерфейсом, погано піддається модульному тестуванню, тому що для цього часто потрібна участь користувача для запуску логіки в UI. Крім того, будь-яка візуалізація часто вимагає оцінки з боку людини, що все «виглядає правильно». Відзначимо, що існують рішення для автоматизації тестування користувацького інтерфейсу.

Однак вони лише імітують взаємодію з користувачем. Як правило, вони складніші в налаштуванні і обслуговуванні, ніж unit-тести, і найчастіше використовуються при інтеграційному тестуванні, так як для цього потрібно запуск всієї програми;

- зменшення можливості перевикористання;
- якщо ваш UI-код змішаний з кодом логіки і даних, то його стає набагато складніше перевикористати;
- ViewModel не може спілкуватися зі View безпосередньо. Замість цього вона являє легко зв'язуються властивості і методи у вигляді команд. View може прив'язуватися до цих властивостей, щоб отримувати інформацію з ViewModel і викликати на ній команди (методи). Це не вимагає того, щоб View знала про ViewModel. XAML Databinding використовує рефлексію, щоб зв'язати View і ViewModel. Таким чином, можливо використовувати будь-яку ViewModel для View, яка надає необхідні якості.

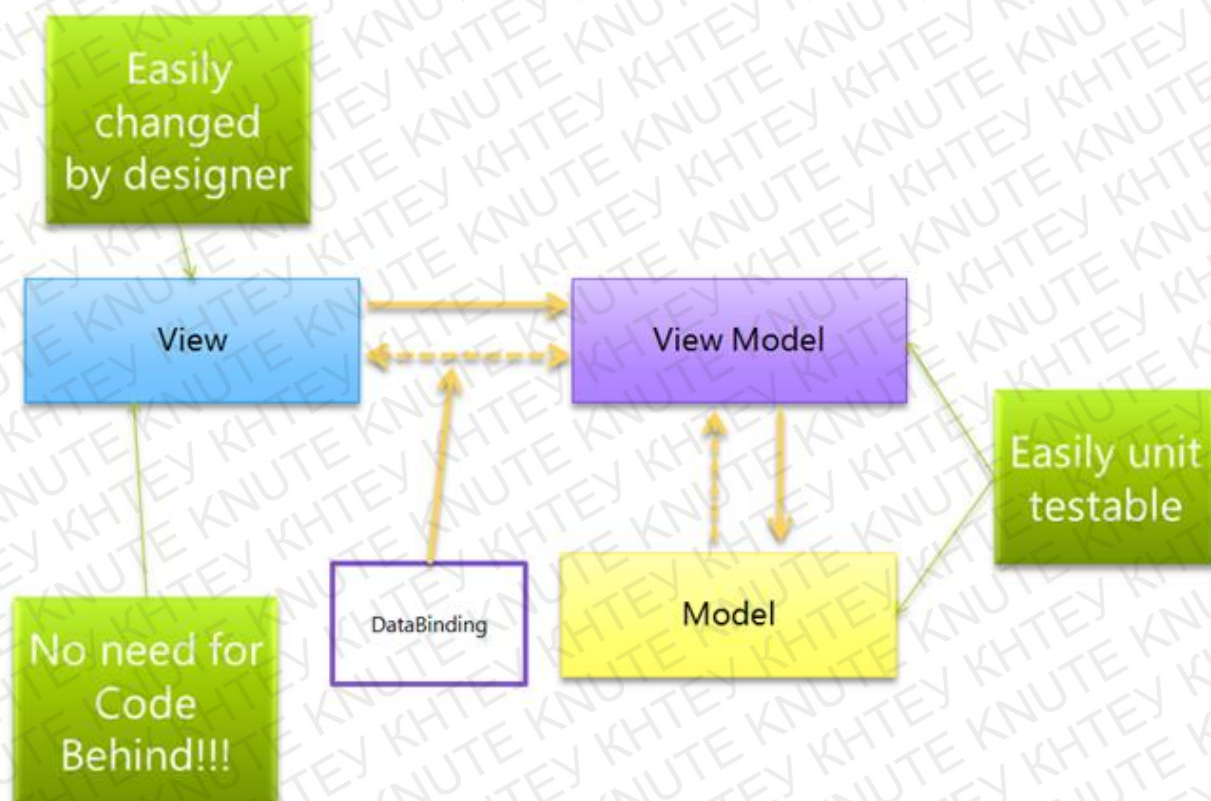


Рис.3.3. Патерн MVVM

До переваг цього патерна, коли він застосовується до WPF, слід віднести:

- отримується повністю тестована логічна модель додатку;
- оскільки ViewModel надає View всю необхідну інформацію в зручному вигляді, то саме уявлення може бути досить простим. А дизайнер може експериментувати із зовнішнім виглядом і стилем в редакторі Expression Blend і змінювати його, не впливаючи на призначений для користувача інтерфейс.

3.4. Технологія використання розробленого додатку

Для отримання доступу до функцій додатку, що реалізує можливість багатокритеріальної оцінки та відбору банків для розміщення грошових вкладів фізичних осіб необхідно запустити файл **VKR.exe**, після чого відкриється вікно додатку(рис. 3.4).

список банків

оберіть потрібні

PrivatBank	OschadBank	Raiffeisen Bank Aval 4	
процентна ставка 11	процентна ставка 10	процентна ставка 9	процентна ставка 123
розмір депозитного портфелю 235200	розмір депозитного портфелю 208631	розмір депозитного портфелю 79190	розмір депозитного портфелю 123
довгострокова динаміка депозитів населенню 12	довгострокова динаміка депозитів населенню 10	довгострокова динаміка депозитів населенню 8	довгострокова динаміка депозитів населенню 123
відповідність капіталу активам 95	відповідність капіталу активам 98	відповідність капіталу активам 90	відповідність капіталу активам 123
ліквідність банку b	ліквідність банку b	ліквідність банку a	ліквідність банку d
динаміка власного капіталу за півріччя 17	динаміка власного капіталу за півріччя 14	динаміка власного капіталу за півріччя 15	динаміка власного капіталу за півріччя 123
динаміка депозитів населення за 12 місяців 10	динаміка депозитів населення за 12 місяців 8	динаміка депозитів населення за 12 місяців 6	динаміка депозитів населення за 12 місяців 123
рейтинг прозорості a	рейтинг прозорості b	рейтинг прозорості c	рейтинг прозорості a
натисніть щоб додати до порівняння	натисніть щоб додати до порівняння	натисніть щоб додати до порівняння	натисніть щоб додати до порівняння

список критеріїв

процентна ставка	1
розмір депозитного портфелю	2
довгострокова динаміка депозитів населенню	3
відповідність капіталу активам	4
ліквідність банку	5

Рис. 3.4. Вікно додатку

Для початку роботи програмне забезпечення отримує інформацію з JSON-файлу **Bank.json**(рис.3.5.). У цьому файлі зазначені показники банків за усіма представленими у роботі критеріями. Такими як депозитна ставка, розмір депозитного портфелю, довгострокова динаміка депозитів населенню,

відповідність капіталу активам, ліквідність банку, динаміка власного капіталу за півріччя, динаміка депозитів населення за 12 місяців, рейтинг прозорості(рис. 3.6)

```
[
  {
    "BankId": "1",
    "BankName": "PrivatBank",
    "Interest_rate": "11",
    "The_size_of_the_deposit_portfolio": "235200",
    "Long_term_dynamics_of_retail_deposits": "12",
    "Capital_adequacy_of_assets": "95",
    "Bank_liquidity": "b",
    "Dynamics_of_equity_for_six_months": "17",
    "Dynamics_of_household_deposit_for_12_months": "10",
    "Transparency_rating": "a"
  },
  {
    "BankId": "2",
    "BankName": "OschadBank",
    "Interest_rate": "10",
    "The_size_of_the_deposit_portfolio": "208631",
    "Long_term_dynamics_of_retail_deposits": "10",
    "Capital_adequacy_of_assets": "98",
    "Bank_liquidity": "b",
    "Dynamics_of_equity_for_six_months": "14",
    "Dynamics_of_household_deposit_for_12_months": "8",
    "Transparency_rating": "b"
  },
  {
    "BankId": "3",
    "BankName": "Raiffeisen Bank Aval",
    "Interest_rate": "9",
    "The_size_of_the_deposit_portfolio": "79190",
    "Long_term_dynamics_of_retail_deposits": "8",
    "Capital_adequacy_of_assets": "90",
    "Bank_liquidity": "a",
    "Dynamics_of_equity_for_six_months": "15",
    "Dynamics_of_household_deposit_for_12_months": "6",
    "Transparency_rating": "c"
  }
]
```

Рис. 3.5. JSON-файл з даними по кожному банку

Серед наданого переліку банків користувач обирає потрібні. Для цього йому необхідно натиснути на кнопку додавання (рис3.7.)

PrivatBank

процентна ставка
11

розмір депозитного портфелю
235200

довгострокова динаміка депозитів населенню
12

відповідність капіталу активам
95

ліквідність банку
b

динаміка власного капіталу за півріччя
17

динаміка депозитів населення за 12 місяців
10

рейтинг прозорості
a

Натисніть щоб додати до порівняння

Рис. 3.6. Додавання банку до порівняння

список критеріїв

процентна ставка	1
розмір депозитного портфелю	2
довгострокова динаміка депозитів населенню	3
відповідність капіталу активам	4
ліквідність банку	5
динаміка власного капіталу за півріччя	6
динаміка депозитів населення за 12 місяців	7
рейтинг прозорості	8

Рис 3.7. Список критеріїв порівняння

Після цієї операції обраний банк буде додано до списку порівняння(рис.3.8.)

список обраних банків

PrivatBank

OschadBank

Raiffeisen Bank Aval

Рис. 3.8. Список обраних банків

Останнім кроком після того як користувач обере усі необхідні банки необхідно натиснути на кнопку "Обчислити" (рис.3.9.). Отриманий результат є рішення поставленої задачі відбору кращого банку для розміщення грошових коштів фізичною особою.

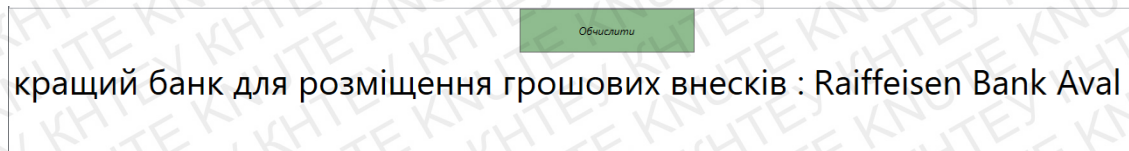


Рис.3.9. Результати розрахунку.

Таким чином, розроблено і реалізовано за допомогою сучасних програмних засобів додаток для вибору кращого банку для внесення грошових коштів фізичними особами, проста у використанні, має інтуїтивно зрозумілий інтерфейс, забезпечує інтерактивний діалог користувача системи в процесі її використання. Представлений програмний додаток не потребує спеціалізованих знань користувачів, оновлення даних відбувається досить легко. На основі цієї програми можливо проводити аналіз необмеженої кількості банків.

ВИСНОВКИ

У випускній кваліфікаційній роботі представлено результати теоретичних і прикладних досліджень, що полягають у розробці програмного забезпечення багатокритеріальної оцінки та відбору банків для розміщення грошових вкладів фізичних осіб. В результаті проведених досліджень були отримані такі висновки:

1. Основними проблемами ефективного розміщення грошових вкладів фізичними особами в умовах нестабільності фінансової системи України і постійних ринкових коливань є такі: 1) неконтрольований відтік грошових коштів з депозитних рахунків клієнтів (депозитний ризик); 2) занадто лояльні законодавчі вимоги щодо диверсифікації депозитного портфелю банку, що в свою чергу підвищує ризиковість активних банківських операцій; 3) низький рівень лояльності населення до депозитних вкладів (що обумовлено недовірою клієнтів до стабільності та надійності роботи банків); 4) низька обізнаність споживачів щодо основних показників діяльності банків як фінансових установ та ступеню їх надійності. В свою чергу наявність цих проблем ускладнює не лише вибір критеріїв відбору кращого банку для розміщення грошового вкладу фізичною особою, але й можливість ранжування цих критеріїв та порівняння їх значущості між собою. Саме це дозволило зробити висновок про оптимальне вирішення цієї задачі за допомогою математичного моделювання з використанням методу багатокритеріального аналізу, спрямованого на знаходження оптимального рішення на підставі аналізу певної кількості критеріїв.

2. При відборі кращого банку для розміщення грошового внеску фізичними особами необхідно використовувати сучасні інформаційні технології, що дозволяє шляхом використання актуальної і надійної фінансової інформації за допомогою побудови цільової функції по кожному критерію для наявних альтернатив та через можливість необмеженої багатозначної оцінки великої кількості конкурентоспроможних критеріїв, отримати найбільш об'єктивний і неупереджений результат.

3. Охарактеризував матрицю попарного порівняння в розрізі порівняння таких критеріїв оптимізації як депозитна ставка, розмір депозитного портфеля, довгострокова динаміка депозитів населенню, відповідність капіталу активам, ліквідність банку, динаміка власного капіталу за півріччя, динаміка депозитів населення за 12 місяців, рейтинг прозорості з метою визначення кращого варіанту із запропонованих альтернатив (тобто кращого банку).

4. Здійснено програмну реалізацію методу максимінної згортки та методу попарних порівнянь для визначення кращого банку для внесення грошових коштів фізичними особами.

5. Розроблене і реалізоване за допомогою сучасних програмних засобів програмне забезпечення для виявлення кращого банку для внесення депозиту фізичними особами. Завдяки інтуїтивно зрозумілому інтерфейсу дуже легкий у використанні для користувачів, що не володіють знаннями у цій предметній області

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Н.Г. Радченко. Методичні підходи до оцінки ефективності депозитної політики банку // збірник наукових праць таврійського державного агротехнологічного університету (економічні науки). № 27. - 2014. - С. 161-166.
2. Ю. П. Макаренко, Т. О. Сагач. Проблеми формування депозитного портфеля банків умовах нестабільності депозитної бази та шляхи їх вирішення // інвестиції: практика та досвід. № 1. - 2016. - С. 15-18.
3. Гарбар Ж.В., Малясова І.В. Особливості формування депозитного портфелю банківських установ [Електронний ресурс]: Вінницький торговельно-економічний інститут КНТЕУ. - 2014. - Режим доступу: <http://intkonf.org/ken-garbar-zhv-malyasova-iv-osoblivosti-formuvannya-depozitnogo-portfelyu-bankivskih-ustanov/>
4. Руцишин Н.М. Суть, значення депозитної політики банку та особливості її формування в сучасних умовах // Збірник II Всеукраїнської конференції Львівської комерційної Академії. - 2014. - С. 59-62.
5. Мокрій О.Г. Критерії оцінки ефективності депозитної політики банку // збірник тез наукових робіт учасників ііі всеукраїнської науково-практичної інтернет-конференції «наукові засади розвитку знань економічної теорії», том іі. 21-22 квітня 2016. - С. 58-62.
6. Нэгл Т.Т. Стратегия и тактика ценообразования / Т.Т. Нэгл, Р.К. Хондел.- СПб.: Питер, 2001.
7. Онишко С. Деякі проблеми кредитного забезпечення потреб економіки інноваційного типу: досвід України // Вісник НБУ. - 2007. - №4. - С.50-53.
8. Максимова А.В. Оцінка фінансової стійкості банку: методичні підходи до аналізу та проблеми їх застосування // науковий вісник херсонського державного університету. Серія "Економічні науки". - Випуск 8. Частина 3. - 2014. - С. 205-208.
9. Вольська С.П. Комплексна оцінка фінансової стійкості банку: зб. матеріалів III Міжнар. наук.-практ. конф. «Економіка підприємництва: теорія і

практика» », Київ, 21 жовт. 2010 р./ ДВНЗ «Київ. нац. екон. ун-т ім. Вадима Гетьмана» / С.П. Вольська. – К. : КНЕУ, 2010. – С. 54–56.

10. Шпаковська Н.І. Методичний підхід до оцінки фінансової стійкості банків. / Н.І. Шпаковська // Науковий вісник: Фінанси, банки, інвестиції. – 2013. – № 3. – С. 100-105.

11. Паночишин Ю.М., Рум'янцева К.Є. Використання математичного апарату теорії нечітких множин і нечіткої логіки в задачі оцінки фінансової стійкості комерційних банків // вісник запорізького національного університету. - № 2. - 2011. - С. 80-86.

12. Заде Л. Понятие лингвистической переменной и ее применение к принятию приближенных решений. – М.: Мир, 1976. – 167 с.

13. Андрейчиков А.В. Андрейчикова О.Н. Анализ, синтез, планирование решений в экономике. - М.: Финансы и статистика, 2002. - 368 с.

14. Теслюк В.М., Загарюк Р.В. Методи багатокритеріальної оптимізації: Ч.1. Конспект лекцій з курсу "Методи багатокритеріальної оптимізації" для студентів спеціальності 8.05010103 "Системне проектування". -Львів: Видавництво Львівської політехніки, 2012. -52 с.

15. Кирик, В. В. Математичний апарат штучного інтелекту в електроенергетичних системах [Електронний ресурс] : підручник / В. В. Кирик ; М-во освіти і науки України, Нац. техн. ун-т України «КПІ ім. Ігоря Сікорського». – Електронні текстові дані (1 файл: 5,52 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, Вид-во «Політехніка» 2019.

16. FUZZY LOGIC TOOLBOX [Електронний ресурс] Режим доступу: <https://matlab.ru/products/fuzzy-logic-toolbox>

17. Станкевич, Л. А. Интеллектуальные системы и технологии : учебник и практикум для бакалавриата и магистратуры / Л. А. Станкевич. — Москва : Издательство Юрайт, 2018. — 397 с.

18. Руда О. Л. Депозитна політика банку та механізм її реалізації // Економіка та держава. -№ 11. - 2018. С. 81-84.

19. Матвеев В.В., Гайдаржийська О.М., Василенко В.В. Дослідження депозитного портфелю як інструменту ресурсної політики банку // Молодий вчений. - № 3 (55). - 2018. - С. 660-664.

20. Литвинюк М.В., Демиденко В.І. Ліквідність банку та банківської системи як показник ефективності діяльності банку та її вплив на прибутковість комерційного банку // Економіка і суспільство. - Випуск 10. - 2017. - С. 631-636.

21. Теслюк В.М., Загарюк Р.В. Методи багатокритеріальної оптимізації: Ч.1. Конспект лекцій з курсу —Методи багатокритеріальної оптимізації для студентів спеціальності 8.05010103 —Системне проектування. — Львів: Видавництво Національного університету —Львівська політехніка, 2012. — 64 с.

22. Теслюк В.М. Градієнтні методи розв'язання оптимізаційних задач. Конспект лекцій з курсу «Методи синтезу та оптимізації» для студентів базового напрямку 6.050101 «Комп'ютерні науки», Ч.3. -Львів: Самвидав. реєстр. номер №4947 від 27.05.2013, Львівська політехніка, 2013. -67 с.

23. Штойер Р. Многокритериальная оптимизация. Теория, вычисления и приложения.- М.: Радио и связь,1992.- 504 с.

24. Сангинова, О. В. Многокритериальная оптимизация в задачах формирования кожевенных и меховых материалов / О. В. Сангинова, А. Г. Данилкович, С. В. Брановицкая // Scientific Journal «ScienceRise». – 2014. – № 2 (2). – С. 43–50.

25. Дзюблюк О.В. Фінансова стійкість банків як основа ефективного функціонування кредитної системи : [монографія] / О.В. Дзюблюк, Р.В. Михайлюк. – Тернопіль, 2009. – 316 с.

26. Гарбар Ж.В., Малясова І.В. Особливості формування депозитного портфелю банківських установ // Режим доступу: <http://intkonf.org/ken-garbar-zhvmalyasova-iv-osoblivosti-formuvannya-depozitnogo-portfelyu-bankivskih-ustanov/>

27. Електронний ресурс:

<https://finsee.com/%D0%BF%D1%80%D0%B8%D0%B2%D0%B0%D1%82%D0%B1%D0%B0%D0%BD%D0%BA-%D0%B4%D0%B5%D0%BF%D0%BE%D0%B7%D0%B8%D1%82%D0%B8/>

28. Гумен І. Складові банківських рейтингів: науково-практичний аспект // Вісник НБУ. – 2000. – №1. – С. 57-60.

29. Іващук О. Концептуальні підходи до ліквідності банку як об'єкту фінансового управління [Текст] / О. Іващук // Галицький економічний вісник. – 2010. – № 2 (27). – С. 163–169.

30. Методологія експертного оцінювання: конспект лекцій / Уклад. Новосад В. П., Селіверстов Р. Г. – К.: Вид-во НАДУ, 2008. – 48 с.

31. Барабаш О.В. Побудова структури мережі передачі даних за критерієм максимуму функціональної стійкості / О.В. Барабаш // Збірник наукових праць «Проблеми інформатизації та управління». – К.: НАУ, 2003. – № 8. – С. 66 – 71.

32. Саати Т. Принятие решений. Метод анализа иерархий / Т. Саати.. – М.: Радио и связь, 1991. – 224 с.

ДОДАТОК

Програмний код реалізації додатку

```
using Ninject;
using System;
using System.Collections.Generic;
using System.Collections.ObjectModel;
using System.IO;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Input;
using VKR.Infrastructure.Command;
using VKR.Infrastructure.Service;
using VKR.Models;

namespace VKR.ViewModels
{
    class MainViewModel : Notifier
    {
        MainWindow mainwindow;

        public ICommand AnalyzeCommand { get; set; }
        public ICommand CheckedCommand { get; set; }
        public ICommand CriteriaCommand { get; set; }
        public Bank SelectedBank { get; set; }
```

```
private String bankBest;
```

```
public String BankBest
```

```
{  
    get => bankBest;  
    set  
    {  
        bankBest = value;  
        Notify();  
    }  
}
```

```
private double result;
```

```
public double Result
```

```
{  
    get => result;  
    set  
    {  
        result = value;  
        Notify();  
    }  
}
```

```
private ObservableCollection<Bank> allbanks = new ObservableCollection<Bank>();
```

```
private ObservableCollection<Bank> checkedBanks = new  
ObservableCollection<Bank>();
```

```
public ObservableCollection<Bank> CheckedBanks
```

```
{  
    get => checkedBanks;  
    set  
    {  
        checkedBanks = value;  
        Notify();  
    }  
}
```

```
private IIOService<Bank> iOServiceBank;
```

```
private ObservableCollection<Bank> banks = new ObservableCollection<Bank>();
```

```
public ObservableCollection<Bank> Banks
```

```
{  
    get => banks;  
    set  
    {  
        banks = value;  
        Notify();  
    }  
}
```

```
private ObservableCollection<Criteria> criterias = new ObservableCollection<Criteria>();
```

```
public ObservableCollection<Criteria> Criterias
```

```
{  
    get => criterias;  
    set  
    {
```

```

        criterias = value;
        Notify();
    }
}

private ObservableCollection<double> weights = new ObservableCollection<double>();
public ObservableCollection<double> Weights
{
    get => weights;
    set
    {
        weights = value;
        Notify();
    }
}

public MainViewModel([Named("Bank")]IIOService<Bank> itemBank)
{
    mainwindow = new MainWindow();
    BankBest = "";
    iOServiceBank = itemBank;
    allbanks = iOServiceBank.Load();
    Banks = allbanks;
    Criterias = Criteria.GetCriterias();
    InitilizeCommands();
    mainwindow.DataContext = this;
    mainwindow.Show();
}

```

```
}
```

```
public void InitilizeCommands()
```

```
{
```

```
    CheckedCommand = new RelayCommand(x => checkedBanks.Add(SelectedBank));
```

```
    AnalyzeCommand = new RelayCommand(x => CriteriaWeights());
```

```
}
```

```
public void CriteriaWeights()
```

```
{
```

```
    Criterias = new ObservableCollection<Criteria>(Criterias.OrderBy(x => x.Value));
```

```
    CriteriaWeight criteriaWeight = new CriteriaWeight();
```

```
    criteriaWeight.Weights[0, 0] = 1;
```

```
    criteriaWeight.Weights[0, 1] = 7;
```

```
    criteriaWeight.Weights[0, 2] = 3;
```

```
    criteriaWeight.Weights[0, 3] = 4;
```

```
    criteriaWeight.Weights[0, 4] = 1.0 / 4;
```

```
    criteriaWeight.Weights[0, 5] = 1.0 / 3;
```

```
    criteriaWeight.Weights[0, 6] = 5;
```

```
    criteriaWeight.Weights[0, 7] = 2;
```

```
    criteriaWeight.Weights[1, 1] = 1;
```

```
    criteriaWeight.Weights[1, 2] = 1;
```

criteriaWeight.Weights[1, 3] = 1.0 / 2;

criteriaWeight.Weights[1, 4] = 1.0 / 7;

criteriaWeight.Weights[1, 5] = 1.0 / 2;

criteriaWeight.Weights[1, 6] = 3;

criteriaWeight.Weights[1, 7] = 2;

criteriaWeight.Weights[2, 2] = 1;

criteriaWeight.Weights[2, 3] = 1.0 / 2;

criteriaWeight.Weights[2, 4] = 1.0 / 4;

criteriaWeight.Weights[2, 5] = 1.0 / 2;

criteriaWeight.Weights[2, 6] = 4;

criteriaWeight.Weights[2, 7] = 1;

criteriaWeight.Weights[3, 3] = 1;

criteriaWeight.Weights[3, 4] = 1.0 / 5;

criteriaWeight.Weights[3, 5] = 1;

criteriaWeight.Weights[3, 6] = 1.0 / 6;

criteriaWeight.Weights[3, 7] = 3;

criteriaWeight.Weights[4, 4] = 1;

criteriaWeight.Weights[4, 5] = 3;

criteriaWeight.Weights[4, 6] = 3;

criteriaWeight.Weights[4, 7] = 2;

criteriaWeight.Weights[5, 5] = 1;

criteriaWeight.Weights[5, 6] = 2;

```
criteriaWeight.Weights[5, 7] = 2;
```

```
criteriaWeight.Weights[6, 6] = 1;
```

```
criteriaWeight.Weights[6, 7] = 7;
```

```
criteriaWeight.Weights[7, 7] = 1;
```

```
for (int i = 0; i < 8; i++)
```

```
{
```

```
    for (int j = 0; j < 8; j++)
```

```
        if (i == j)
```

```
        {
```

```
            break;
```

```
        }
```

```
        else
```

```
        {
```

```
            criteriaWeight.Weights[i, j] = 1.0 / criteriaWeight.Weights[j, i];
```

```
        }
```

```
    }
```

```
for (int i = 0; i < 8; i++)
```

```
    criteriaWeight.Id[i] = Criterias[i].Id;
```

```
Amin amin = new Amin();
```

```
for (int i = 0; i < 8; i++)
```

```
    amin.Vector[i] = 1;
```

```
for (int i = 0; i < 8; i++)
```

```

    {
        for (int j = 0; j < 8; j++)
        {
            amin.Vector[i] *= criteriaWeight.Weights[i, j];
        }
        amin.Id[i] = criteriaWeight.Id[i];
    }
    for (int i = 0; i < 8; i++)
    {
        amin.Vector[i] = Math.Pow((double)amin.Vector[i], (double)(1.0 / 8));
    }

    double r = amin.Vector.Sum();
    for (int i = 0; i < 8; i++)
        amin.A_min[i] = ((amin.Vector[i] / r) * 8);
//1

checkedBanks = new
ObservableCollection<Bank>(checkedBanks.OrderByDescending(x => x.Interest_rate));

double[] Interest_rate = new double[checkedBanks.Count];
for (int i = 0; i < checkedBanks.Count; i++)
    if (i == 0)
        checkedBanks[i].Interest_rate_pr = 0.9;
    else
        checkedBanks[i].Interest_rate_pr = (double)(checkedBanks[i].Interest_rate * 0.9) /
(double)checkedBanks[0].Interest_rate;
//2

```

```

checkedBanks = new
ObservableCollection<Bank>(checkedBanks.OrderByDescending(x
=>
x.The_size_of_the_deposit_portfolio));

double[] The_size_of_the_deposit_portfolio = new double[checkedBanks.Count];
for (int i = 0; i < checkedBanks.Count; i++)
    if (i == 0)
        checkedBanks[i].The_size_of_the_deposit_portfolio_pr = 0.8;
    else
        checkedBanks[i].The_size_of_the_deposit_portfolio_pr =
        (double)checkedBanks[i].The_size_of_the_deposit_portfolio * 0.8 /
        (double)checkedBanks[0].The_size_of_the_deposit_portfolio;
//3

checkedBanks = new
ObservableCollection<Bank>(checkedBanks.OrderByDescending(x
=>
x.Long_term_dynamics_of_retail_deposits));

double[] Long_term_dynamics_of_retail_deposits = new double[checkedBanks.Count];
for (int i = 0; i < checkedBanks.Count; i++)
    if (i == 0)
        checkedBanks[i].Long_term_dynamics_of_retail_deposits_pr = 0.9;
    else
        checkedBanks[i].Long_term_dynamics_of_retail_deposits_pr =
        (double)checkedBanks[i].Long_term_dynamics_of_retail_deposits * 0.9 /
        checkedBanks[0].Long_term_dynamics_of_retail_deposits;
//4

checkedBanks = new
ObservableCollection<Bank>(checkedBanks.OrderByDescending(x
=>
x.Capital_adequacy_of_assets));

double[] Capital_adequacy_of_assets = new double[checkedBanks.Count];

```

```

for (int i = 0; i < checkedBanks.Count; i++)
    if (i == 0)
        checkedBanks[i].Capital_adequacy_of_assets_pr = 0.6;
    else
        checkedBanks[i].Capital_adequacy_of_assets_pr =
            (double)checkedBanks[i].Capital_adequacy_of_assets * 0.6 /
            checkedBanks[0].Capital_adequacy_of_assets;

//5
checkedBanks = new
ObservableCollection<Bank>(checkedBanks.OrderByDescending(x
=>
x.Dynamics_of_equity_for_six_months));

double[] Dynamics_of_equity_for_six_months = new double[checkedBanks.Count];
for (int i = 0; i < checkedBanks.Count; i++)
    if (i == 0)
        checkedBanks[i].Dynamics_of_equity_for_six_months_pr = 0.7;
    else
        checkedBanks[i].Dynamics_of_equity_for_six_months_pr =
            (double)(checkedBanks[i].Dynamics_of_equity_for_six_months * 0.7 /
            checkedBanks[0].Dynamics_of_equity_for_six_months);

//6
checkedBanks = new
ObservableCollection<Bank>(checkedBanks.OrderByDescending(x
=>
x.Dynamics_of_household_deposit_for_12_months));

double[] Dynamics_of_household_deposit_for_12_months = new
double[checkedBanks.Count];

for (int i = 0; i < checkedBanks.Count; i++)
    if (i == 0)
        checkedBanks[i].Dynamics_of_household_deposit_for_12_months_pr = 0.9;

```

```

else

checkedBanks[i].Dynamics_of_household_deposit_for_12_months_pr =
(double)checkedBanks[i].Dynamics_of_household_deposit_for_12_months * 0.9 /
checkedBanks[0].Dynamics_of_household_deposit_for_12_months;

//7

checkedBanks = new
ObservableCollection<Bank>(checkedBanks.OrderByDescending(x => x.Bank_liquidity));

double[] Bank_liquidity = new double[checkedBanks.Count];
for (int i = 0; i < checkedBanks.Count; i++)
{
    if (checkedBanks[i].Bank_liquidity == 'a')
        checkedBanks[i].Bank_liquidity_pr = 1;
    if (checkedBanks[i].Bank_liquidity == 'b')
        checkedBanks[i].Bank_liquidity_pr = 0.75;
    if (checkedBanks[i].Bank_liquidity == 'c')
        checkedBanks[i].Bank_liquidity_pr = 0.5;
    if (checkedBanks[i].Bank_liquidity == 'd')
        checkedBanks[i].Bank_liquidity_pr = 0.1;
}

//8

checkedBanks = new
ObservableCollection<Bank>(checkedBanks.OrderByDescending(x => x.Transparency_rating));

double[] Transparency_rating = new double[checkedBanks.Count];
for (int i = 0; i < checkedBanks.Count; i++)
{
    if (checkedBanks[i].Transparency_rating == 'a')
        checkedBanks[i].Transparency_rating_pr = 1;

```

```

        if (checkedBanks[i].Transparency_rating == 'b')
            checkedBanks[i].Transparency_rating_pr = 0.75;
        if (checkedBanks[i].Transparency_rating == 'c')
            checkedBanks[i].Transparency_rating_pr = 0.5;
        if (checkedBanks[i].Transparency_rating == 'd')
            checkedBanks[i].Transparency_rating_pr = 0.1;
    }
    ///min
    ObservableCollection<Bmin> bmin = new ObservableCollection<Bmin>();
    for (int i = 0; i < checkedBanks.Count; i++)
    {
        Bmin bminelement = new Bmin();

        for (int j = 0; j < 8; j++)
        {
            if (amin.Id[j] == 1)
                bminelement.Interest_rate_min = Math.Pow(checkedBanks[i].Interest_rate_pr,
amin.A_min[j]);
            if (amin.Id[j] == 2)
                bminelement.The_size_of_the_deposit_portfolio_min =
Math.Pow(checkedBanks[i].The_size_of_the_deposit_portfolio_pr, amin.A_min[j]);
            if (amin.Id[j] == 3)
                bminelement.Long_term_dynamics_of_retail_deposits_min =
Math.Pow(checkedBanks[i].Long_term_dynamics_of_retail_deposits_pr, amin.A_min[j]);
            if (amin.Id[j] == 4)
                bminelement.Capital_adequacy_of_assets_min =
Math.Pow(checkedBanks[i].Capital_adequacy_of_assets_pr, amin.A_min[j]);

```

```

        if (amin.Id[j] == 5)
            bminelement.Bank_liquidity_min =
Math.Pow(checkdBanks[i].Bank_liquidity_pr, amin.A_min[j]);

        if (amin.Id[j] == 6)
            bminelement.Dynamics_of_equity_for_six_months_min =
Math.Pow(checkdBanks[i].Dynamics_of_equity_for_six_months_pr, amin.A_min[j]);

        if (amin.Id[j] == 7)
            bminelement.Dynamics_of_household_deposit_for_12_months_min =
Math.Pow(checkdBanks[i].Dynamics_of_household_deposit_for_12_months_pr, amin.A_min[j]);

        if (amin.Id[j] == 8)
            bminelement.Transparency_rating_min =
Math.Pow(checkdBanks[i].Transparency_rating_pr, amin.A_min[j]);
    }

    bminelement.IdBank = checkdBanks[i].BankId;
    bmin.Add(bminelement);
}

double[] result = new double[checkdBanks.Count];
for (int i = 0; i < checkdBanks.Count; i++)
{
    double[] check = new double[8];
    check[0] = bmin[i].Bank_liquidity_min;
    check[1] = bmin[i].Capital_adequacy_of_assets_min;
    check[2] = bmin[i].Dynamics_of_equity_for_six_months_min;
    check[3] = bmin[i].Dynamics_of_household_deposit_for_12_months_min;
    check[4] = bmin[i].Interest_rate_min;

```

```

        check[5] = bmin[i].Long_term_dynamics_of_retail_deposits_min;

        check[6] = bmin[i].The_size_of_the_deposit_portfolio_min;

        check[7] = bmin[i].Transparency_rating_min;

        result[i] = check.Min();
    }

    int a = 0;

    for (int i = 0; i < checkedBanks.Count; i++)
    {
        if (result[i] == result.Max())

            for (int j = 0; j < checkedBanks.Count; j++)

                if (bmin[i].IdBank == checkedBanks[j].BankId)
                {
                    BankBest = checkedBanks[i].BankName;

                    a++;

                    break;
                }

        if (a > 0)

            break;
    }

    Result = result.Max();
}
}
}

```

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Linq;
using System.Runtime.CompilerServices;
using System.Text;
using System.Threading.Tasks;

namespace VKR.Infrastructer.Command
{
    class Notifier : INotifyPropertyChanged
    {
        public event PropertyChangedEventHandler PropertyChanged;

        public void Notify([CallerMemberName]string prop = "")
        {
            PropertyChanged?.Invoke(this, new PropertyChangedEventArgs(prop));
        }
    }
}

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

```

using System.Windows.Input;

namespace VKR.Infrastructure.Command

{

class RelayCommand : ICommand

{

private readonly Action<object> execute;

private readonly Func<object, bool> canExecute;

public event EventHandler CanExecuteChanged

{

add { CommandManager.RequerySuggested += value; }

remove { CommandManager.RequerySuggested -= value; }

}

public RelayCommand(Action<object> action, Func<object, bool> func = null)

{

execute = action;

canExecute = func;

}

public bool CanExecute(object parameter)

{

return canExecute == null || canExecute(parameter);

```

    }

    public void Execute(object parameter)
    {
        execute(parameter);
    }
}

using Newtonsoft.Json;
using Ninject;
using System;
using System.Collections.Generic;
using System.Collections.ObjectModel;
using System.IO;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using VKR.Infrastructure.Resource;
using VKR.Models;

namespace VKR.Infrastructure.Service
{
    class JSONIOServiceForBanks: IIOService<Bank>
    {
        public ResourcePath Path { get; set; }
    }
}

```

```

public JSONIOServiceForBanks([Named("Bank")]ResourcePath path) => Path = path;

public ObservableCollection<Bank> Load()
{
    return
    JsonConvert.DeserializeObject<ObservableCollection<Bank>>(File.ReadAllText(Path.Path));
}

public void Save(ObservableCollection<Bank> item)
{
    File.WriteAllText(Path.Path, JsonConvert.SerializeObject(item, Formatting.Indented));
}
}

using Newtonsoft.Json;
using Ninject;
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using VKR.Infrastructure.Resource;
using VKR.Models;
using VKR.ViewModels;

```

```

namespace VKR.Infrastructure.Service
{
    class ServiceLocator
    {
        public MainViewModel MainViewModel { get; }

        IKernel kernel = new StandardKernel();

        Config config = JsonConvert.DeserializeObject<Config>(File.ReadAllText("Config.json"));

        public ServiceLocator()
        {
            kernel.Bind<IIOService<Bank>>().To<JSONIOServiceForBanks>().Named("Bank");

            kernel.Bind<ResourcePath>().To<ResourcePath>().Named("Bank").WithPropertyValue("Path", value:
            filePath.Bank);

            MainViewModel = kernel.Get<MainViewModel>();
        }
    }
}

```

```
<Window x:Class="VKR.MainWindow"
```

```
xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
```

```
xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
```

```

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
xmlns:local="clr-namespace:VKR.Infrastructure.Service"
mc:Ignorable="d"

Title="MainWindow" Height="800" Width="1400">
<Window.Resources>

    <ItemsPanelTemplate x:Key="listPanelTemplate">

        <WrapPanel />

    </ItemsPanelTemplate>

</Window.Resources>

<Grid>

    <ScrollViewer Grid.Row="1" Name="ScrollViewer1" VerticalScrollBarVisibility="Auto"
HorizontalScrollBarVisibility="Disabled" Height="800" CanContentScroll="True">

        <StackPanel>

            <Label FontSize="40">список банків</Label>

            <Label>оберіть потрібні</Label>

            <ListView ItemsSource="{Binding Banks}" SelectedItem="{Binding
SelectedBank}" ScrollViewer.HorizontalScrollBarVisibility="Disabled" ItemsPanel="{StaticResource
listPanelTemplate}">

                <ListBox.ItemTemplate>

                    <DataTemplate>

                        <StackPanel>

                            <TextBlock FontSize="40" Text="{Binding BankName}">

                                </TextBlock>

                                <Label>процентна ставка</Label>

                                <TextBlock Text="{Binding Interest_rate}">

                                    </TextBlock>

```

```

<Label>розмір депозитного портфелю</Label>
<TextBlock Text="{Binding The_size_of_the_deposit_portfolio}">
</TextBlock>
<Label>довгострокова динаміка депозитів населенню</Label>
<TextBlock Text="{Binding Long_term_dynamics_of_retail_deposits}">
</TextBlock>
<Label>відповідність капіталу активам</Label>
<TextBlock Text="{Binding Capital_adequacy_of_assets}">
</TextBlock>
<Label>ліквідність банку</Label>
<TextBlock Text="{Binding Bank_liquidity}">
</TextBlock>
<Label>динаміка власного капіталу за півріччя</Label>
<TextBlock Text="{Binding Dynamics_of_equity_for_six_months}">
</TextBlock>
<Label>динаміка депозитів населення за 12 місяців</Label>
<TextBlock
Text="{Binding
Dynamics_of_household_deposit_for_12_months}">
</TextBlock>
<Label>рейтинг прозорості</Label>
<TextBlock Text="{Binding Transparency_rating}">
</TextBlock>
<Button
CommandParameter="{Binding
RelativeSource=
{RelativeSource AncestorType=Window, Mode=FindAncestor}}}"
Height="50" Width="200" Background="Green"
Command="{Binding
RelativeSource={RelativeSource
AncestorType=Window, Mode=FindAncestor}, Path= DataContext.CheckedCommand}">

```

Натисніть щоб додати до порівняння

</Button>

</StackPanel>

</DataTemplate>

</ListBox.ItemTemplate>

</ListView>

<Label FontSize="40">список критеріїв</Label>

<ListView ItemsSource="{Binding Criterias}"
ScrollViewer.HorizontalScrollBarVisibility="Disabled">

<ListBox.ItemTemplate>

<DataTemplate>

<StackPanel Orientation="Horizontal">

<Label Content="{Binding Name}"></Label>

<TextBox Text="{Binding Value}"></TextBox>

</StackPanel>

</DataTemplate>

</ListBox.ItemTemplate>

</ListView>

<Label FontSize="40">список обраних банків</Label>

<ListBox ItemsSource="{Binding CheckedBanks}"
ScrollViewer.HorizontalScrollBarVisibility="Disabled">

<ListBox.ItemTemplate>

<DataTemplate>

<StackPanel Orientation="Horizontal">

<Label Content="{Binding BankName}"></Label>

```

        </StackPanel>
    </DataTemplate>
</ListBox.ItemTemplate>
</ListBox>
<Button      Height="50"      Width="200"      Background="DarkSeaGreen"
FontStyle="Italic" Command="{Binding AnalyzeCommand}">Обчислити</Button>
<StackPanel Orientation="Horizontal">
    <Label FontSize="40">кращий банк для розміщення грошових внесків :
</Label>
    <Label FontSize="40" Content="{Binding BankBest}"></Label>
</StackPanel>
</StackPanel>
</ScrollView>
</Grid>
</Window>

```

```

using System;
using System.Collections.Generic;
using System.Collections.ObjectModel;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace VKR.Models
{

```

```

class Criteria
{
    public int Id { get; set; }

    public int Value { get; set; }

    public string Name { get; set; }

    public static ObservableCollection<Criteria> GetCriteria()
    {
        ObservableCollection<Criteria> criterias = new ObservableCollection<Criteria>();
        criterias.Add((new Criteria { Name = "процентна ставка", Value = 1, Id = 1 }));
        criterias.Add((new Criteria { Name = "розмір депозитного портфелю", Value = 2, Id
= 2 }));
        criterias.Add((new Criteria { Name = "довгострокова динаміка депозитів
населенню", Value = 3, Id = 3 }));
        criterias.Add((new Criteria { Name = "відповідність капіталу активам", Value = 4, Id
= 4 }));
        criterias.Add((new Criteria { Name = "ліквідність банку", Value = 5, Id = 5 }));
        criterias.Add((new Criteria { Name = "динаміка власного капіталу за півріччя",
Value = 6, Id = 6 }));
        criterias.Add((new Criteria { Name = "динаміка депозитів населення за 12 місяців",
Value = 7, Id = 7 }));
        criterias.Add((new Criteria { Name = "рейтинг прозорості", Value = 8, Id = 8 }));
        return criterias;
    }
}

```

