

УДК 658.51

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _____ Комп'ютерних наук _____

Кафедра _____ Системотехніки _____

КВАЛІФІКАЦІЙНА РОБОТА

Пояснювальна записка

_____ *другий (магістерський)* _____

(рівень вищої освіти)

_____ *ГЮИК.506160.025 ПЗ* _____

(позначення документа)

_____ *Дослідження методів формування проектних команд в ІТ-індустрії* _____

(тема)

Виконав:

Студент 2 курсу, групи *ІТІМ-20-1*

Спеціальність *122 Комп'ютерні науки*

(код і повна назва спеціальності)

Тип програми *освітньо-професійна*

(освітньо-професійна або освітньо-наукова)

Освітня програма *Інформаційні технології проектування*

(повна назва освітньої програми)

_____ *Івановська К.А.* _____

(прізвище, ініціали)

Керівник *проф. Овезгельдисв А.О.*

(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри системотехніки _____

(підпис)

_____ *проф. Гребеннік І.В.* _____

(прізвище, ініціали)

2021 р.

Харківський національний університет радіоелектроніки

Факультет _____ *Комп'ютерних наук* _____

Кафедра _____ *Системотехніки* _____

Рівень вищої освіти _____ *другий (магістерський)* _____

Спеціальність _____ *122 Комп'ютерні науки* _____
(код і повна назва)

Освітня програма _____ *Інформаційні технології проектування* _____

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

« ____ » _____ 20 ____ р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові _____ *Івановській Ксенії Анатоліївни* _____
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження методів формування проектних команд в ІТ-індустрії затверджена наказом по університету від _____ *08.11* 2021 р. № *1663 Ст*
2. Термін подання студентом роботи до екзаменаційної комісії _____ *15.12.2021р.*
3. Вихідні дані до роботи: підручники, довідники, електронні ресурси за обраною предметною областю, методи які будуть досліджуватися, план дослідження, вхідні тестові дані до задачі, завчасно сформовані ідеальні результати на тестовій вибірці, методи аналізу даних, алгоритми розв'язання задач про призначення, алгоритми розв'язання багатокритеріальних задач.
4. Перелік питань, що потрібно опрацювати в роботі 4.1 Вступ. 4.2 Аналіз предметної області. 4.2.1 Управління персоналом. 4.2.2 Процедура формування проектних команд та ролі в командах. 4.2.3 Роль соціометрії у процесі формування проектних команд. 4.3 Огляд методів технологій, що застосовуються в предметній області. 4.3.1 Найпоширеніші методи прийняття рішень. 4.3.2 Огляд СППР, що використовують у ІТ. 4.3.3 Методи Data Mining 4.4 Постановка задачі дослідження. 4.5 Розробка інформаційної технології вирішення задачі. 4.5.1 Розгляд задачі багатокритеріальної оптимізації. 4.5.2 Розгляд задачі про призначення. 4.5.3 Огляд методів числового оцінювання альтернатив. 4.5.4 Огляд методів, ефективність яких буде оцінюватися 4.6 Дослідження та аналіз отриманих результатів. 4.6.1 Проведення дослідження для методу опорних векторів. 4.6.2 Проведення дослідження для методу k-найближчих сусідів. 4.6.3 Підготовка тестових даних. 4.6.4 Аналіз отриманих результатів.

5.Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) 5.1 Загальна схема алгоритму k-ближніх сусідів. 5.2 Загальна схема алгоритму методу опорних векторів. 5.3 Контекстна діаграма. 5.4 Діаграма декомпозиції основного бізнес-процесу контекстної діаграми. 5.5 Діаграма декомпозиції бізнес-процесу «Accounting for employees». 5.6 Діаграма декомпозиції бізнес-процесу «Authentication and registration». 5.7 Діаграма декомпозиції бізнес-процесу «Defining employee criteria for a new project». 5.8 Діаграма варіантів використання системи. 5.9 Схема логічної моделі даних для БД. 5.10 Схема фізичної моделі даних для БД. 5.11 Діаграма послідовностей для прецеденту «Формування команд за певними критеріями». 5.12 Діаграма послідовностей для прецеденту «Додавання інформації про працівника до бази даних». 5.13 Діаграма станів системи під час роботи проектного менеджера. 5.14 Діаграма класів підсистеми аналізу даних.

6.Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів	Примітка
1	<i>Отримання завдання кваліфікаційної роботи</i>	<i>01.09.2021</i>	
2	<i>Аналіз завдання, літератури та аналогів з теми кваліфікаційної роботи</i>	<i>06.09.2021-15.09.2021</i>	
3	<i>Огляд методів та технологій, що застосовуються у предметній області</i>	<i>16.09.2021-30.09.2021</i>	
4	<i>Розробка інформаційної технології вирішення задачі</i>	<i>01.10.2021-18.10.2021</i>	
5	<i>Проведення математичних досліджень</i>	<i>19.10.2021-08.11.2021</i>	
6	<i>Аналіз результатів досліджень</i>	<i>09.11.2021-23.11.2021</i>	
7	<i>Оформлення пояснювальної записки</i>	<i>24.11.2021-02.12.2021</i>	
8	<i>Оформлення графічної частини та презентаційних матеріалів комп'ютерного захисту</i>	<i>03.12.2021-07.12.2021</i>	
9	<i>Представлення роботи на рецензування</i>	<i>09.12.2021</i>	
10	<i>Попередній захист</i>	<i>13.12.2021</i>	
11	<i>Представлення кваліфікаційної роботи</i>	<i>15.12.2021</i>	

Дата видачі завдання 01.09.2021 р.

Студент _____

(підпис)

Керівник роботи _____

(підпис)

Івановська К.А.

проф. Овезгельдисв А.О.

(посада, прізвище, ініціали)

РЕФЕРАТ

Кваліфікаційна робота містить: 73 сторінки, 13 формул, 16 рисунків, 6 таблиць, 3 додатки, 29 джерел.

Об'єктом досліджень є методи, які використовуються у системах підтримки прийняття рішень та допомагають обрати найкращий варіант проектної команди для виконання визначеного обсягу робіт.

Предметом досліджень є рівень знань спеціалістів у певних галузях, соціальні відносини між працівниками, сукупність теоретико-методичних та практичних підходів, взаємозв'язки, рольовий розподіл та результати спільної праці, що виникають при формуванні ефективних проектних команд, а також ефективність методів, які використовуються у СППР.

Мета дослідження – визначення ефективності методів, що використовуються у СППР для аналізу процесів формування команд та оцінки працездатності кожного з її членів, на підставі яких й відбувається визначення ефективності їх сумісної роботи.

Методи дослідження – метод статистичного аналізу, системний та процесний підходи для аналізу процесу створення команди з урахуванням соціальних відносин, методи Data Mining, методи вирішення багатокритеріальних задач, задачі про призначення, методи числового оцінювання альтернатив та приведення їх до загального вигляду, метод опорних векторів, метод k-ближчих сусідів.

У роботі проаналізовано результати математичних досліджень для вибору найкращої проектної команди задля автоматизації роботи проектних менеджерів, надання чіткого структурування необхідної інформації та мінімізації фактору людської помилки.

Галузь застосування – ІТ-компанії будь-якого масштабу з довільною кількістю працівників.

ABSTRACT

Qualifying work contains: 73 pages, 13 formulas, 16 figures, 6 table, 3 appendices, 29 sources.

The object of research is the methods used in decision support systems and help to choose the best project team to perform a certain amount of work.

The subject of research is the level of knowledge of specialists in certain fields, social relations between employees, a set of theoretical and methodological and practical approaches, relationships, role distribution and results of joint work that arise in the formation of effective project teams, and the effectiveness of methods used in DSS.

The purpose of the study is to determine the effectiveness of the methods used in DSS to analyze the processes of team building and assess the performance of each of its members, on the basis of which is to determine the effectiveness of their joint work.

Research methods – statistical analysis method, system and process approaches for analysis of team building process taking into account social relations, Data Mining methods, methods for solving multicriteria problems, assignment problems, methods of numerical evaluation of alternatives and bringing them to general form, reference vectors method, method k-nearest neighbors.

The paper analyzes the results of mathematical research to select the best project team to automate the work of project managers, provide a clear structuring of the necessary information and minimize the human error factor.

Scope – IT companies of any scale with any number of employees.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ ТА ТЕРМІНІВ.....	6
ВСТУП.....	7
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	9
1.1 Управління персоналом.....	12
1.2 Процедура формування проектних команд та ролі в командах.....	15
1.3 Роль соціометрії у процесі формування проектних команд.....	19
2 ОГЛЯД МЕТОДІВ ТА ТЕХНОЛОГІЙ, ЯКІ ЗАСТОСОВУЮТЬСЯ В ПРЕДМЕТНІЙ ОБЛАСТІ.....	26
2.1 Найпоширеніші методи прийняття рішень.....	27
2.2 Огляд СППР, що використовують у ІТ.....	31
2.3 Методи Data Mining.....	34
3 ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ.....	36
4 РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ВИРШЕННЯ ЗАДАЧІ.....	43
4.1 Розгляд задачі багатокритеріальної оптимізації.....	43
4.2 Розгляд задачі про призначення.....	44
4.3 Огляд методів числового оцінювання альтернатив.....	47
4.4 Огляд методів, ефективність яких буде оцінюватися.....	49
5 ДОСЛІДЖЕННЯ ТА АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ.....	56
5.1 Проведення дослідження для методу опорних векторів.....	56
5.2 Проведення дослідження для методу k-найближчих сусідів.....	59
5.3 Підготовка тестових даних.....	61
5.4 Аналіз отриманих результатів.....	67
ВИСНОВКИ.....	69
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....	70
ДОДАТОК А Графічний матеріал кваліфікаційної роботи.....	73
ДОДАТОК Б Текст програми.....	90
ДОДАТОК В Відомість атестаційної роботи.....	103

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ ТА
ТЕРМІНІВ

DSS – Decision Support System.

HRM – Human Resource Management.

MCDM – Multiple-Criteria Decision-Making.

MCDA – Multiple-Criteria Decision Analysis.

OLAP – OnLine Analytical Processing.

БД – база даних.

ЕК – електронна комерція.

ІС – інформаційна система.

ІСППСПР – інтелектуальна система підтримки прийняття рішень.

ІТ – інформаційні технології.

КІС – корпоративна інформаційна система.

ОПР – особа, що приймає рішення.

СППСПР – система підтримки прийняття рішень.

SVM – support vector machine.

ВСТУП

У наш час важко уявити життя та роботу людей без комп'ютеру, інтернету та різних програмних засобів. З точки зору звичайного користувача людина лише використовує певні програми та не має замислюватися над реалізацією та виконанням цих програм. Але процес створення будь-яких додатків – це насправді складна багаторівнева система, що використовує бізнес-логіку, містить в собі безліч правил, модулів, функцій, розрахунків тощо. Деякі з програм створюються, змінюються та підтримуються роками й навіть десятиліттями, тоді як найпростішу мобільну гру можна створити за декілька місяців щоденної роботи.

У сучасній ІТ-спільноті не часто можна зустріти працівників, що власноруч працюють над усім проектом (так звані стартапи з невеликою кількістю виконавців). Це доволі рідкісне явище, коли одна людина в змозі ефективно, повноцінно та за виділений обмежений проміжок часу розробити весь проект (бекенд, фронтенд, якісно протестувати весь додаток, створити базу даних та документацію тощо). Тому всі ІТ-проекти та компанії стикаються з необхідністю наявності людей різної спеціалізації та різного рівня знань, які будуть об'єднані в окремі команди для роботи над спільним проектом, де кожен з них виконуватиме окрему, виділену саме для нього, частину задач.

Проектна команда, в цілому, це сукупність окремих груп та осіб, що залучені до виконання задач проекту та які несуть відповідальність перед керівником або менеджером проекту за виконання цих задач. Для великих і середніх проектів, які нерідко можуть налічувати сотні або навіть тисячі учасників, під командою проекту розуміють керівництво проекту, що складається з менеджера проекту і менеджерів, які несуть відповідальність за окремі напрямки діяльності проекту або його функції. Для малих проектів, де чисельність учасників, безпосередньо залучених до виконання завдань проекту, не часто перевищує 10 чоловік, всі з них відносяться до команди проекту.

В умовах сьогодення, доволі гострою є проблема того, що при наявності нових перспективних проектів (частіш за все, дуже добре оплачуваних) не вистачає працівників на IT-підприємствах для виконання задач. Не завжди у компанії є можливість найняти ще декількох спеціалістів (а часом є неможливим знаходження людини, необхідної кваліфікації, яка б поєднувала у собі усі необхідні знання та якості), щоб вони працювали над новими задачами. Саме у такі моменти проектні менеджери стикаються з тим, що необхідно переводити співробітників з проекту на проект або розподіляти робочий час працівника у якомусь співвідношенні (наприклад, 30 відсотків робочого часу на старому проекті та 70 – на новому) задля більш ефективної роботи усіх команд в цілому.

Але неможливо здійснити переведення працівника з його проекту на новий лише за бажанням проект-менеджера або іншої особи – спочатку необхідно провести якісний аналіз того, наскільки це буде ефективним та яку вигоду може надати компанії.

Метою професійної практики є оцінка ефективності різних методів при підтримці прийняття рішень у процесі формування проектних команд.

Тема порівняння ефективності методів при підтримці прийняття рішень при формуванні проектних команд пройшла апробацію на наукових конференціях. По результатам конференцій були видані збірники [1, 2].

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

Основною предметною областю даної роботи є системи підтримки прийняття рішень (СППР) та робота з ними. СППР – це автоматизована система, що була створена з метою допомогти людині прийняти рішення в складних умовах, ця система забезпечує повний та об'єктивний аналіз вхідних даних та предметної діяльності. Іншими словами – це система, яка отримує вхідні дані, запит щодо роботи з ними та після обробки цього запиту повертає інформацію користувачеві одним зі способів: друкованим документом, виведення результату на монітор комп'ютера, звуковим сигналом тощо. Відповідь системи на запит може складатися з речення, тексту, таблиці, будевого значення та допомагає користувачеві швидко і точно оцінити ситуацію та прийняти рішення. СППР виникли як результат об'єднання інформаційних управлінських систем (ІУС) і систем управління базами даних (СУБД) [5].

Для аналізу інформації та формування відповіді такі системи використовують різні методи та різні набори даних (дані можуть бути структуровані, не структуровані або слабко структуровані).

Методами для аналізу та обробки вхідної інформації є інформаційний пошук, інтелектуальний аналіз даних, пошук знань в існуючих базах даних або формування нових баз знань на основі отриманих результатів, оцінювання вхідної інформації на основі прецедентів, нейронні мережі, ситуаційний аналіз, імітаційне моделювання, еволюційні обчислення та генетичні алгоритми тощо. Деякі з цих методів розроблялися в рамках штучного інтелекту, отже якщо в основі роботи СППР лежить метод штучного інтелекту, то вважається, що така СППР є інтелектуалізованою (ІСППР). Найближчими за змістом роботи до СППР є експертні системи та автоматизовані системи управління [17].

Процес прийняття рішення – це доволі складний та відповідальний вибір найоптимальнішого (найкращого) варіанту серед певної множини альтернативних рішень. Існує безліч методів, які дозволяють спростити та пришвидшити процес

прийняття оптимального рішення шляхом скорочення множини варіантів та/або виділення оптимальних критеріїв серед множини усіх параметрів, але слід пам'ятати, що системи підтримки прийняття рішень лише надають можливі варіанти, оптимальні для роботи їх алгоритму, а ось остаточне рішення завжди належить людині – особі, що уповноважена здійснювати вибір та здатна логічного та критично мислити. Рішенням може вважатися чітко обґрунтований набір дій з боку особи, що приймає рішення, яких спрямований на об'єкт чи систему управління та надає можливість перевести об'єкт/систему з поточного стану в необхідний задля досягнення поставленої мети [4]. Сам термін «рішення» може бути інтерпретований у багатьох сенсах: він може відповідати вибору «найкращої» альтернативи з набору доступних альтернатив (де «найкраща» = «найбільш бажана» альтернатива для особи, що приймає рішення), а також сенс терміну «рішення» може полягати у виборі невеликого набору задовільних альтернатив або у групуванні альтернатив в різномантні набори переваг[17].

Важливим є те, що будь-яке рішення має сенс та право на існування лише тоді, коли воно ефективне та поліпшує роботу об'єкту чи системи, а не погіршує її. Ефективність будь-якого рішення – числове значення, яке може бути виражене за допомогою доволі простої формули:

$$E = Q * A, \quad (1.1)$$

де Q – фактор якості рішення,

A – фактор прийняття рішення людиною.

Багатокритеріальне прийняття рішень (MCDM) – є важливою додатковою дисципліною досліджень операцій, яка явно оцінює множинні конфліктуючі критерії при прийнятті будь-яких рішень (в щоденному житті, в умовах бізнесу, політики, навчання, медицини тощо). Конфліктуючі критерії майже у всіх ситуаціях є умовно типовими: зазвичай. грошові витрати або отримання є одним з основних критеріїв, а

показники якості є допоміжними критеріями, які можуть значно суперечити грошовому параметру.

Наприклад, при створенні будь-якого програмного продукту для кінцевого користувача вартість цього продукту, комфортне та зрозуміле використання, наявна технічна підтримка та захист даних є основними критеріями, але у сучасних реаліях покупцю такого продукту може здаватися незвичним та підозрілим, що найдешевша програма є найзручнішою і найбезпечнішою, а отже, для ефективної реалізації кінцевого продукту має бути розумно оцінений баланс між послугами, що надаються та їх вартістю. У сфері ІТ-послуг задоволеність клієнтів, їх довіра до продукту та вартість надання послуг є фундаментальними суперечливими критеріями.

Щодня людина не замислюючись зважає безліч критеріїв для прийняття рішень та може бути як задоволена наслідками рішень, що були прийняті спираючися лише на інтуїцію, так і абсолютно розчарована. Якщо ставки при прийнятті рішення занадто високі важливою є задача правильно структурувати проблему, явно оцінити критерії, можливі варіанти рішення та обрати найкраще з них. Явне та усвідомлене структурування складних задач з великою кількістю критеріїв призводить до прийняття більш ефективних рішень, ніж ті, що приймаються на основі так званого «шостого чуття». MCDM займається структуруванням даних і розв'язанням задач з рішень і планувань, що пов'язані з вибором на основі декількох критеріїв. Мета MCDM полягає в підтримці осіб, які мають приймати рішення, але стикаються з певними проблемами. Як правило, для задач не існує єдиного оптимального рішення, бо кожна задача є у своєму роді унікальною. Для того, щоб якісно визначити всі доступні варіанти перед прийняттям рішення необхідно використовувати чіткий математичний апарат в купі з вподобаннями людини, що приймає рішення. Перед безпосереднім початком визначення доступних варіантів для прийняття рішення слід звертати увагу на додаткові субдисципліни, процеси та процедури.

1.1 Управління персоналом

Управління персоналом (менеджмент персоналу, HRM) – систематично-організаційний процес відтворення та ефективного використання персоналу.

Управління персоналом можна визначити як галузь знань або практичну діяльність, що спрямована на постійну неперервну підтримку найефективнішого використання працівників для досягнення цілей підприємства та особистісних цілей. Перші, традиційно, пов'язуються з забезпеченням ефективності підприємства. Причому ефективність іноді розуміється у вузькому значенні – як отримання максимального прибутку. Однак дедалі частіше ефективність розглядається не тільки в економічному плані – як економічність, якість, продуктивність, нововведення, прибуток, але і в більш широкому контексті й пов'язується з такими поняттями особистісного та психологічного плану, як задоволеність співробітників своєю працею, участю у трудовому колективі підприємства, високий рівень самооцінки колективу, мотивація персоналу до ефективної праці, бажання вдосконалювати свої навички [3].

Управління персоналом розвивається в трьох напрямках:

- більше розвивається на практиці (як практична діяльність);
- є індивідуально зорієнтованим на працівника, як особистість і надає послуги по задоволенню потреби;
- зорієнтований на майбутнє (вписується у стратегію організації).

Як сфера практичної діяльності управління персоналом – це система взаємозв'язаних організаційно-економічних заходів по створенню умов для нормального функціонування, розвитку ефективного використання кадрового потенціалу організації.

Управління персоналом стоїть у центрі управління організації, у той час як управління організації займає місце в управлінні бізнесом. Предметом менеджменту персоналу є управління працівниками у процесі виробництва з точки зору найбільш повного та ефективного використання їх потенціалу. Основною метою менеджменту

персоналу є поєднання ефективного навчання персоналу, підвищення кваліфікації та стимулювання його до високоефективної праці.

Для досягнення цієї мети, насамперед, потрібно чітко з'ясувати особливості управлінської праці.

Управління персоналу має 4 рівні:

- оперативний (домінує кадрова робота);
- тактичний (домінує, власне, управління персоналом);
- стратегічний (домінує управління людськими ресурсами);
- політичний (домінує робота по розробці кадрової політики).

Відносно сфер діяльності управління персоналом охоплює такі важливі для компанії функції, як:

- пошук та початкова адаптація нового персоналу після входження працівника до нового колективу та ознайомлення зі специфікою майбутньої роботи;
- робота з існуючим персоналом, що полягає у підвищенні рівня знань персоналу, загальному розвитку, проведенні об'єктивної оцінки кар'єрного росту кожного з працівників, а також у підвищенні мотивованості та бажанні залишатися у компанії якнайдовше;
- стратегічна робота з персоналом.

Основними задачами, які вирішуються за допомогою управління персоналом можна вважати:

- створення спільної стратегії роботи;
- визначення необхідності компанії у нових працівниках, ураховуючи уснуючих та зміни у кадровому складі;
- здійснення дій з підбору персоналу та формування команд на поточних проектах;
- управління витратами на працівників (заробітні плати, страхування, премії, грошова допомога у надзвичайних випадках);
- проведення аналізу якості роботи та виявлення необхідності вдосконалення робочого процесу;

- виставлення пропозицій щодо професійної підготовки, перепідготовки, підвищення кваліфікаційного рівня;
- розробка системи для індивідуального управління кар'єрою кожного з працівників у рамках поточного проекту та компанії в цілому;
- здійснення заходів щодо мотивації працівників, як то підвищення заробітної плати, надання премій, соціальний пакет з пільгами тощо.

Можна вважати, що управління персоналом базується на наступних дев'яти принципах:

- принцип підбору працівника відносно його ділових та особистих якостей;
- принцип наслідування або наступності, що полягає у поєднанні на проекті молодих та досвідчених працівників;
- принцип відповідності, у змісті якого лежить те, що доручена задача має відповідати досвіду та рівню знань виконавця;
- принцип тимчасової заміни відсутнього працівника його колегами або іншою людиною (за необхідності);
- принцип забезпечення умов для постійного професійного зростання та вдосконалення знань;
- принцип підвищення кваліфікації (що тісно пов'язаний з попереднім принципом);
- принцип правової захищеності працівника, який досягається за допомогою прийняття лише тих управлінських та кадрових рішень, які базуються на діючому законі трудового законодавства;
- принцип відкритої та прозорої конкуренції між колегами, що претендують на керівну позицію;
- принцип поєднання довіри до працівників з контролем якості виконання людиною своїх прямих обов'язків.

Отже, як зрозуміло з написаного вище, головною задачею управління персоналом є максимально ефективне використання трудових ресурсів, що є у підприємства та готовність безперервно вдосконалювати роботу персоналу.

1.2 Процедура формування проектних команд та ролі в командах

Управління персоналом дуже пов'язане з управлінням проектами, адже не існує проектів без виконавців. У свій час одним із найважливіших завдань управління проектом є формування команди. Керівникам проекту і функціональних підрозділів, що приймають участь у створенні, реалізації та підтримці проекту, на цій стадії необхідно вирішувати ряд доволі специфічних задач, які пов'язані з мотивацією праці, конфліктами, виконанням, контролем, відповідальністю, комунікаціями, лідерством та інше. Це створює сприятливі умови для роботи, допомагає перебороти величезні психологічні навантаження, що виникають у процесі пошуку, узгодження і реалізації проектних рішень, дозволяє уникнути конфліктів і стресів, що в кінцевому рахунку позначаються на науково-технічному рівні і якості проекту. Багато дослідників підтверджують, що близько 80 % опитаних, ставлять фактор людських відносин на перше місце з усіх факторів, що впливають на успішне здійснення проекту, тому пріоритетність цієї сфери діяльності не викликає сумнівів. Створення професійної команди для нового проекту – один із основних обов'язків проект-менеджера на першому етапі його роботи. Цей процес вимагає ряд навиків управління у визначенні, відборі і об'єднанні в команду спеціалістів із різних відділів і організацій [4].

У IT-сфері є більше 20 ролей на проектах, які можна розділити на 3 узагальнені блоки. Важливо розуміти, що ці ролі не залежать напряму від кількості людей у командах, так як не завжди у команді є 15 та більше осіб, але чим менше команда, тим більше ролей має (свідомо чи ні) виконувати одна людина. Виконання усіх ролей допомагає підтримувати ефективну працездатність проектів, а втрата якісного виконання навіть однієї з ролей може призвести до того, що ніби-то ідеальна картинка почне розпадатися з надзвичайною швидкістю.

Розглянемо ті три блоки, про які було сказано раніше:

- "Локомотив", ролі, які всіх просувають всіх до досягнення мети;

- галузева/предметна експертиза в продукті/бізнес-моделі;
- виробнича експертиза в реалізації/розробці.

У блок «Локомотив» входять такі ролі:

- Замовник/Owner;
- Product Owner;
- Проджект менеджер.

Блок галузевої експертизи включає:

- Тім-лід;
- Product Owner;
- Бізнес-Аналітик.

А блок виробничої експертизи складається з таких ролей:

- Архітектор;
- Тех-лід/Системний аналітик;
- Розробники(front-end, back-end, інтеграції, Full-stack, Mobile);
- DevOps;
- UX-дизайнер;
- UI-дизайнер;
- QA (тестувальники – manual, automation, load testing);
- Technical writer (або тех-райтер).

За рівнями спеціалізації виділяють стажерів (intern/trainee), молодших спеціалістів (junior), спеціалістів (middle), старших спеціалістів (senior) та керівних спеціалістів (lead).

Тобто, узагальнено, проектна команда – це сукупність людей різних ролей та різних рівнів спеціалізації, що підпорядковані проектному менеджеру та працюють разом над виконанням певного проекту. Проектна команда створюється на період реалізації проекту з метою досягнення синергічного ефекту групової роботи, що виникає внаслідок колективного опрацювання проектних рішень, надання конструктивної допомоги одним членам команди (менш досвідчених) з боку інших (більш досвідчених), створення колективної бази даних (досвіду, інформації, правил). Основними характеристиками проектних команд є: склад – сукупність характеристик

членів команди (чисельність, вік, стать, освіта тощо); структура – співвідношення кількості працівників за функціями, які вони виконують (професійний склад); групові процеси – показники динаміки (характеристика процесів розвитку, спрацьованість членів групи, вироблення рішень, групового тиску).

Отже, формуючи команду, проект-менеджер збирає разом групу людей, намагаючись об'єднати їх загальною метою та задачами, результат виконання яких призведе до створення загального продукту. Новизна, унікальність, ризик і швидкоплинність – всі ці риси можуть бути притаманні новому проекту, саме вони і визначають труднощі при формуванні команди. Створення команди для нового проекту ускладнено ще й тим, що, можливо, ці люди ніколи не працювали разом, не мають загальних цінностей і норм, але повинні працювати ефективно і синхронно. Зазвичай, необхідний тривалий час, щоб всередині групи сформувалося командне почуття, встановились загальні норми, стандарти і цінності. Щоб проект був успішним, згрупування людей повинно відбутися максимально швидко і до того часу, коли команда почне працювати “на повну потужність”.

За своєю формою команда проекту відображає певну організаційну структуру управління проектом, розділення функцій, обов'язків і відповідальності за рішення, що приймаються в процесі його реалізації. На верхньому рівні структури знаходиться менеджер проекту, а на нижніх – виконавці, відділи і фахівці, що відповідають за окремі функціональні сфери.

За змістом команда проекту є групою фахівців високої кваліфікації та різної спеціалізації, що володіють знаннями і навичками, необхідними для ефективного досягнення цілей проекту у найкоротший термін з максимальною вигодою для своєї фірми.

Основним чинником створення та подальшої діяльності команди виступає стратегічна мета – реалізація проекту. У процесі досягнення цілей проекту команда набуває певних навичок роботи разом, використовує організаційні можливості учасників і ресурси проекту. Команда проекту виступає як окремий соціальний організм, що має свій початок, здійснює процес життєдіяльності (управління

проектом) і завершує своє існування після закінчення проекту розформуванням або трансформацією в іншу команду для роботи над новою задачею.

Таким чином можна зробити висновки, що процес формування команди (тобто підбору людей з орієнтуванням на конкретні критерії, що дозволяють їм не тільки ефективно виконувати трудову діяльність і досягати бажаних результатів, але і робити це спільно, злагоджено та швидко, за необхідності допомагаючи один одному) – є однією з найважливіших проблем в темі групового суб'єкту праці, що диктується насамперед реальними прикладними запитами.

Сам процес формування команди проекту (або ж командування) звичайно розглядають як утворення єдиного, цілісного колективу управлінців, здатного ефективно досягати мети проекту. Значення командної роботи по реалізації проекту укладається в можливості синергетичного ефекту від об'єднання групових зусиль, знань і вироблення групових управлінських рішень, тобто в досягненні “стану, при якому ціле більше, ніж сума його складових частин”. Аналогічно життєвому циклу проекту команда проекту має свій життєвий цикл, в якому можна виділити п'ять основних стадій: формування, спрацювання, функціонування, реорганізація, розформування [5, 7].

Однією з важливих проблем, від вирішення якої залежить ефективність проектного управління в цілому, є проблема оцінки діяльності функціонального виконавця в системі проектування організаційної структури управління. Функціональний виконавець підпорядкований, щонайменше, функціональному керівнику та керівнику проекту. Якщо функціональному керівнику виконавець підпорядкований постійно, то керівнику проекту – тимчасово, на період виконання робіт, пов'язаних із реалізацією даного проекту. В багатьох випадках виконавець одночасно приймає участь у декількох проектах, в зв'язку із чим підпорядковується декільком керівникам. Якщо система орієнтована на кінцеву ціль – виконання проекту, то вона сприяє скороченню термінів виконання проекту, підвищенню оперативності вирішення поточних проблем, пов'язаних з ходом виконання проекту, більш збалансованій узгодженості програми робіт із ресурсними можливостями, економії ресурсів, а також більш об'єктивній оцінці діяльності окремих спеціалістів.

Існують два підходи до формування команд:

- а) функціональний – фахівці однієї спеціальності або професії об'єднуються у функціональні підрозділи;
- б) цільовий – об'єднуються виконавці різних спеціальностей або функцій і працюють разом над певною частиною проекту або завданням [10].

1.3 Роль соціометрії у процесі формування проектних команд

При формуванні будь-яких команд з людей окрім професійних якостей необхідно використовувати й соціометричні знання, щоб вивчити соціальну поведінку кожного члена команди в умовах спільної роботи та щоб була можливість оцінювати соціально-психологічну сумісність членів різних команд.

Соціометрія – це наука, що вивчає міжособові відносини або ж методика вивчення міжособових зв'язків, відносин та ієрархії в невеликих групах осіб. В перекладі з латинської «соціометрія» – це вимірювання відносин між особами в певній групі або кластері. Соціометричні дослідження принципово відрізняються від усіх інших соціологічних досліджень тим, що тут не може бути повної анонімності, бо тоді буде втрачено ефективність досліджень, а також тим, що результатами опитувань є не характеристика конкретного респондента, а опис взаємини між усіма особами, що опитувалися. Також характерним є те, що об'єднані відповіді усіх респондентів формують структуру відносин. Ця структура може бути представлена у вигляді соціограми (спеціального малюнку-схеми) або соціоматриці (таблиці).

Використання соціометрії може допомогти при визначенні лідерів груп, переформуванню команд з метою зниження психологічного напруження, антипатії між колегами тощо. Саме дослідження зазвичай проводиться не особисто для кожного члена групи, а для усіх одразу та займає не більше 15-20 хвилин, його результати дуже корисні при роботі по вдосконаленню відносин в колективі, а отже й ефективності роботи [6].

Перед початком соціометричного опитування необхідно визначити постановку задачі, вибору об'єктів, сформувані на цій основі основні гіпотези, що можуть стосуватися критеріїв опитування, після цього критерії заносяться у спеціальну таблицю для опитіваних або питання ставляться усно під час так званого інтерв'ю. Кожен з учасників опитування має відповідати на запитання, обираючи того чи іншого члена групи в залежності від типу питання (симпатія, антипатія, довіра, недовіра, бажання працювати під керівництвом особи або навпаки керувати особою, тощо).

Опитування може відбуватися у 2 формах :

- непараметрична процедура означає, що при відповіді на питання немає обмеження числа вибору, тобто в групі з N осіб максимальне число відповідей дорівнює $(N-1)$, тобто обрати не можна лише самого себе;
- параметрична процедура має на увазі те, що опитуваним дають строго фіксоване число осіб, які мають бути обрані при відповіді на питання.

Перевагою першої процедури є те, що вона дозволяє виявити емоційну експансивність кожного з опитуваних осіб, недоліками є те, що при великих розмірах групи опитуваних (від 20 осіб) важко проаналізувати результати без обчислювальної техніки, а також те, що особа може обрати варіант «за всіх» у випадку якщо вона дійсно байдужа до всього, що відбувається навколо неї або ж свідомо прикриває свою фальшиву відповідь формальною лояльністю до колег.

Недоліком другої процедури є те, що неможливо розкрити різноманіття взаємовідносин в групі.

Також на ряду з соціометрією слід звертати увагу на групову динаміку – систему поведінкових та психологічних процесів, які відбуваються в рамках групи з кількох людей (внутрішньогрупова динаміка), або між різними групами (міжгрупова динаміка). Іншими словами важливою є взаємодія як всередині команди, так і зовнішній вплив на неї.

Насправді, формування групи починається саме з психологічного зв'язку між індивідами. Важливим є те, що для формування групи необхідний певний рівень психологічної відмінності між людьми в ній. Саме через взаємодію людей з різними поглядами на ситуації та на вирішення проблем, починають розроблятися рішення (з

метою досягнення компромісу), які групи людей з однаковими поглядами ніколи б не розглянули.

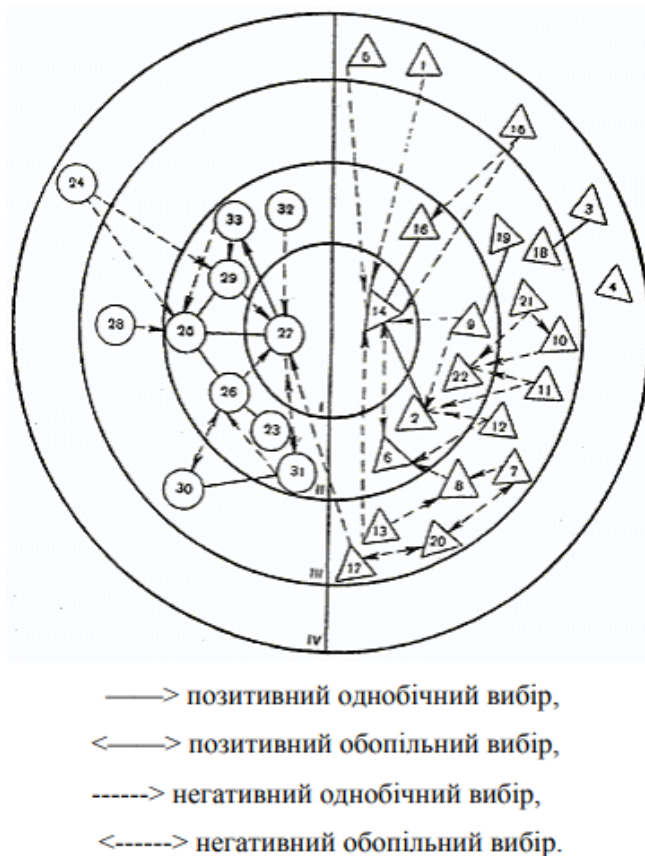


Рисунок 1.1 – Приклад соціограми (карти групової диференціації)

Динаміка малої за розміром групи відображає складність кожної особистості, що входить до її складу, і наявність такої групової властивості, як синергізм на основі взаємодії цих особистостей, внаслідок чого виникає цілісність, яка є чимось більшим, ніж просто сумою її окремих членів. Навіть якщо взяти саму елементарну групу, яка складається з 2 осіб, то в ній можна виділити вплив «Я-концепції» однієї особи на «Я-концепцію» іншої особи та взаємовідносини цих людей. Таким чином, кожна група характеризуватиметься неповторною своєрідністю своєї динаміки. Вивчення та дослідження групової динаміки може бути дуже корисним для розуміння поведінки прийняття рішень у колективі. Групова динаміка лежить в основі розуміння расизму,

сексизму та інших форм соціальних упереджень й дискримінації та створення груп з мінімально можливими проявами цих упереджень.

Групова динаміка має бути проведена з потенційними сформованими групами та складається з 4 стадій розвитку:

а) знайомство, під час якого відбувається знайомство учасників, емоційна оцінка оточуючих та ситуації загалом. Учасник відповідає для себе на запитання: Чи мені тут комфортно? Як поведуться оточуючі відносно мене? Як краще показати себе? Кому можна довіряти? та на багато інших;

б) розподіл потенційних ролей у команді – у цей період в групі з'являються лідери, які роблять взаємодію групи більш емоційною. Саме лідери активізують групу, організують взаємодію у міні-групах, але вони ж можуть дезорганізувати роботу;

в) стійка робота групи – період максимальної працездатності, коли вся група зосереджено працює над досягненням спільної мети (якась виокремлена задача, що може бути на реальному проекті). Учасники довіряють один одному, готові до плідної співпраці та обговорення під час прийняття рішень. Мікроклімат у команді має бути рівний, спокійний, комфортний для усіх учасників;

г) завершення взаємодії групи – етап на якому відбувається завершення роботи, учасники підбивають підсумки взаємодії, резюмують собі найцінніше. Усі учасники в емоційно піднесеному настрої активні, доброзичливі, негатив по відношенню до інших членів команди має бути відсутній.

Якщо всі 4 стадії пройшли за планом – можна робити висновки, що ця команда має потенціал до плідної взаємодії на реальному проекті [8].

Для спрощення процесу відбору співробітників до проектних команд вводять інтегральний показник, який характеризує кожного члена команди одночасно як суб'єкт вибору і як його об'єкт. Цей індекс характеризує внесок співробітника в зміцнення групової згуртованості, а, отже, в підвищення ефективності роботи команди. Назвемо його персональний внесок i -го співробітника в групову взаємодію V_i . Даний показник можна визначити як середнє арифметичне індексів соціометричного статусу і емоційної експансивності індивіда

$$V_i = \frac{C_i + E_i}{2}. \quad (1.2)$$

Чим вище значення даного індексу, тим більший внесок співробітник вносить в підвищення ефективності групової взаємодії. Отже, при формуванні нової команди перевагу слід віддавати не тільки «зіркам» – співробітникам з найвищим значенням соціометричного статусу, а й співробітникам з найвищим значенням персонального вкладу в групову взаємодію.

Соціометричні дослідження із застосуванням дихотомічного критерію оцінки учасників, які проводилися у випадкових групах розмірністю від 6 до 12 людей, показали, що зі збільшенням розмірності групи кількість співробітників з співпадаючими значеннями індексу персонального вкладу (V) збільшується. Графік залежності числа співробітників зі співпадаючими значеннями індексу V від розмірності групи наведено на рисунку 1.2.

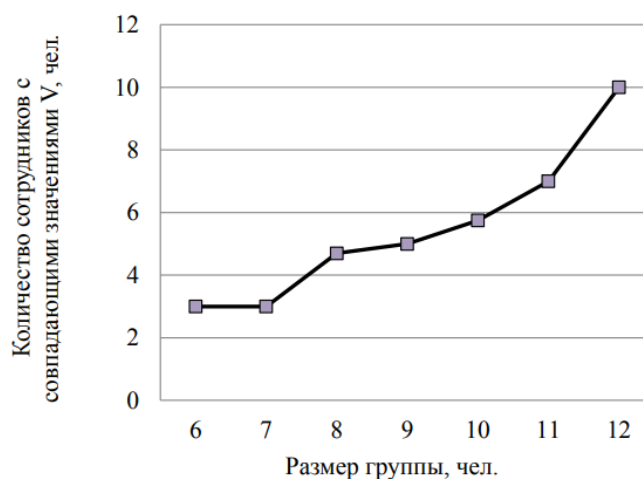


Рисунок 1.2 – Залежність кількості співробітників зі співпадаючими значеннями V від розміру групи

Команда має працювати ефективно, але що саме маєтсья на увазі під «ефективністю»? Ефективною можна називати команду, в якій:

- члени команди мають єдине уявлення про мету роботи та її очукваний результат, розуміють завдання та перспективи розвитку, реально оцінюють час, який потрібен на виконання цих цілей та час, який закладено на виконання певної задачі;
- існує єдина система цінностей, перелік внутришніх правил;
- кожен з членів команди має особисту зацікавленість в досягненні командних цілей, робота відбувається за принципом «результат проекту – результат роботи кожного працівника»;
- існує лідер, однозначно визнаний командою (лід проекту). Це може бути як окрема посада для людини, яка контролює роботу, так і можливе поєднання керування командою з роботою у ній (розробка, тестування, аналіз тощо);
- панує неформальна і відкрита атмосфера, але підтримується субординація;
- усі члени команди прислухаються до порад та/або з легкістю можуть прохати про допомогу;
- в обговорення приймають участь усі члени команди та заохочується як висловлювання ідей, так і вираження емоцій (у культурній формі);
- рішення ґрунтується на приходженні до спільної компромісної думки, а не на голосуванні більшості;
- конфлікти і розбіжності між членами групи центруються навколо ідей і методів, а не навколо особистих стосунків.

Також, оцінка ефективності діяльності команди здійснюється за такими показниками:

- чітке розуміння цілі та спрямованість на кінцевий результат;
- чіткий розподіл функцій і відповідальності;
- наявність плану розвитку команди
- командна солідарність;
- взаєморозуміння і безконфліктність;
- відвідуваність робочих мітингів та активна участь усіх членів команди у вирішенні проблем.

Говорячи про пріоритети проектів, логічно, що першочергова перевага віддається проектам з більшим значенням чистої приведеної вартості (Net Present Value, NPV).

$$NVP = \sum_{t=0}^n \frac{CF_t}{(1+k)^t}, \quad (1.3)$$

де CF – сподівані чисті грошові потоки у визначений часовий період t ,

k – вартість капіталу проекту,

t – порядковий номер розрахунку,

n – кількість років роботи над проектом.

Доволі зрозумілим є те, що, не зважаючи на таку пріоритетність, робота над будь-яким проектом має виконуватися якісно, вчасно та ефективно. Адже не можна віддати перевагу кільком найбільш пріоритетним проектам, пустивши на самоплив проекти з мінімальною пріоритетністю.

2 ОГЛЯД МЕТОДІВ ТА ТЕХНОЛОГІЙ, ЯКІ ЗАСТОСОВУЮТЬСЯ В ПРЕДМЕТНІЙ ОБЛАСТІ

Процес управління в більшості випадків реалізується через певні рішення, підготовку та прийняття яких умовно можна назвати технологією розробки (прийняття) рішень. Вона являє собою сукупність послідовних дій, які повторюються та складаються з окремих етапів, процедур або операцій.

Існує багато схем розробки та реалізації управлінських рішень, що різняться між собою за мірою деталізації, але всі вони в певній мірі включають п'ять етапів, наведених на узагальненій традиційній схемі (рис. 2.1).

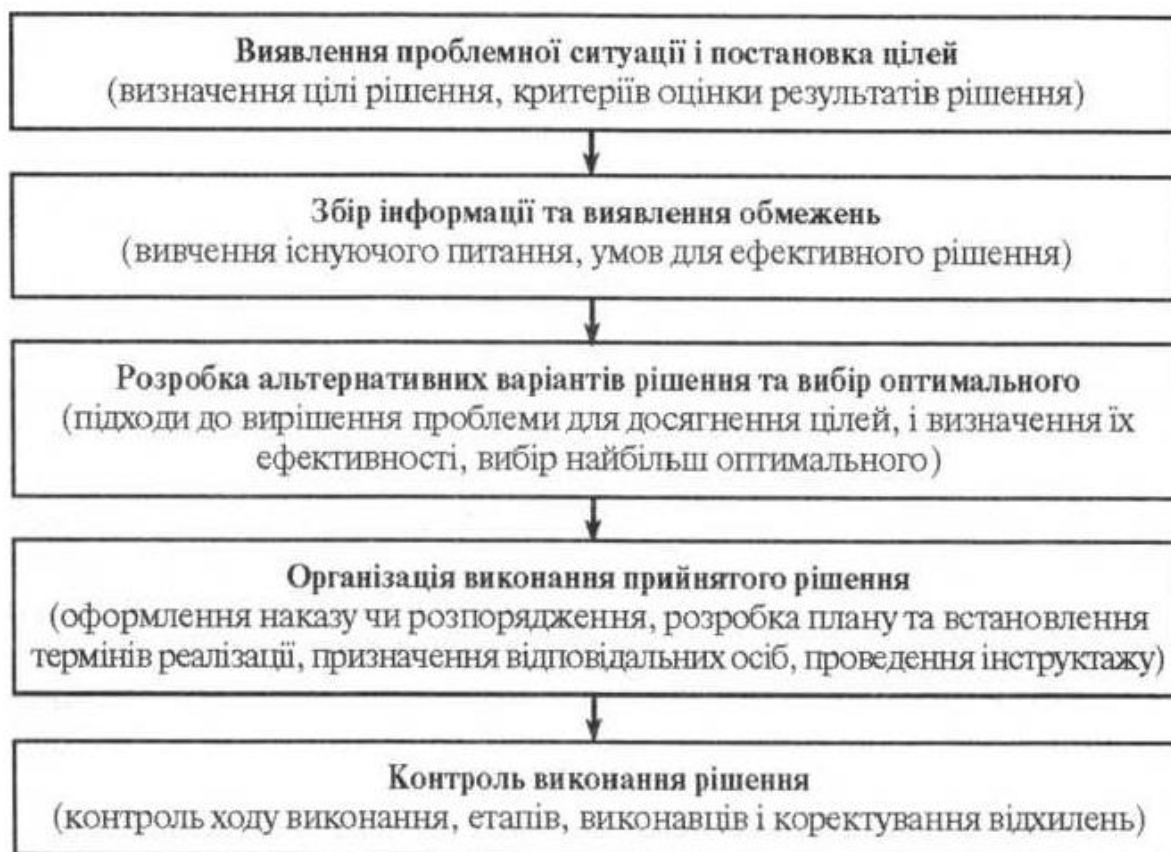


Рисунок 2.1 – Традиційна схема розробки і реалізації управлінських рішень

Вибір методу розв'язання проблеми впливає на перелік етапів і процедур технологічного процесу що будуть використовуватися для пошуку рішення. Так, з позиції системотехніки (напрямок науки і техніки, який охоплює проектування, створення, випробування та експлуатацію складних систем технічного і соціально-технічного характеру, а також є прикладним втіленням теорії систем) найчастіше етапи розв'язання проблеми включають:

- уточнення задачі і вибір цілей;
- перелік або розробку альтернатив;
- аналіз альтернатив;
- вибір найкращого рішення;
- надання результатів.

Узагальнений або типовий процес розробки рішень, в якому використовуються наукові методи, складається з послідовних етапів:

- попереднього базового формулювання задачі;
- вибіркритеріїв оцінки ефективності рішення;
- збір даних для уточнення поставленої задачі та визначення точної постановки задачі;
- розробка можливих варіантів рішення задачі;
- складання математичних моделей;
- зіставлення варіантів за критерієм ефективності та вибір альтернатив;
- власне, прийняття рішення [12].

2.1 Найпоширеніші методи прийняття рішень

Всі методи прийняття рішень умовно можна розділити на 3 різні групи: евристичні (неформальні), колективні та кількісні. Управлінська практика говорить, що при прийнятті рішень певна частина проект-менеджерів (або керівників) використовує евристичні методи, які засновані на аналітичних здібностях людей, які приймають ці рішення [13].

Евристичні методи – це певна сукупність логічних прийомів і методики вибору найкращих рішень керівником, аналіз альтернатив з урахуванням досвіду, які базуються в основному на інтуїції тої людини, що й приймає рішення. Перевага цих методів в тому, що вони приймаються швидко. Недоліком може стати те, що неформальні методи не гарантують захист від вибору неефективних рішень, оскільки не виключена помилка через людський фактор.

Головною відзнакою у роботі колективного методу для реалізації управлінських рішень є те, що коло осіб (учасників даної процедури) чітко визначене. У більшості випадків це спеціально створений колектив, до складу якого найчастіше входять і керівники, і виконавці.

Головними критеріями формування такої групи працівників є конструктивність мислення, компетентність, комунікабельність, здатність вирішувати задачі за допомогою декількох методів. Колективні форми групової роботи можуть мати вигляд мітингів, нарад, роботи в комісії тощо. Найбільш поширений такий метод колективного підготування управлінських рішень, як «мозковий штурм» (спільне генерування нових ідей і наступне за ним прийняття найкращих рішень).

Якщо треба вирішити складну проблему, збирається група людей, що пропонують будь-яке рішення певної проблеми. Загальна суть «мозкового штурму» – створення таких умов, що будуть максимально сприятливі для генерування ідей, де кожен з учасників не буде соромитися висловити свою думку. Щоб цього досягти, забороняється критикувати будь-яку ідею, якою би на перший погляд нереальною або важко реалізованою вона не здавалася б. Абсолютно всі ідеї записуються, а вже потім аналізуються фахівцями [15].

Прикладом колективного прийняття рішення може бути метод Дельфа, що названий на честь грецького міста Дельф, що славився мудрецьми – провісниками майбутнього. Цей метод – багатоетапна процедура анкетування фахівців, де після кожного етапу дані допрацьовуються, а отримані результати повідомляються експертам з указанням розташування обраних оцінок. Перший тур анкетування проводиться без додаткової інформації – лише вибір варіанту, а от на другому етапі кожна відповідь з першого туру має бути або аргументована, або експерт може

змінити свій вибір та аргументувати вже новий. Після стабілізації оцінок опитування припиняється та приймається взважене рішення.

Також існує кільцева(кількісна) система прийняття рішень, її суть в тому, що на обговорення готується проект-нововведення. Цей проект передається для обговорення особам за списком, який складе керівник. Кожний зі списку має розглянути проект та висловити свої зауваження в письмовому вигляді, лише після цього може проводитися нарада. Слід зазначити, що на нараду запрошуються ті фахівці, чия думка керівнику була не зовсім зрозуміла або кардинально відрізнялася від більшості думок інших фахівців. Експерти висловлюють своє рішення відповідно до тих переваг, які вони визначають для себе і якщо рішення не збігаються, то виникає вектор переваг, який визначається за допомогою одного з принципів:

- принцип більшості голосів – обирається одне рішення, яке має найбільше число людей, хто проголосував за нього;
- диктатура – за основу береться думка особи, яка вважається важливою (такий принцип характерний або для військових організацій, або для прийняття рішень у надзвичайних обставинах);
- принцип Курно використовується в тому випадку, коли немає груп, об'єднаних за прийнятими рішеннями, тобто пропонується число рішень, яке рівне числу експертів. У цьому випадку необхідно знайти те рішення, що відповідатиме поставленій задачі, але при цьому не буде обмежувати інтереси кожного з експертів;
- принцип Парето використовується при прийнятті рішень, коли всі експерти утворюють єдине ціле (коаліцію). У цьому випадку оптимальним для прийняття буде те рішення, яке не вигідно змінювати відразу всім членам групи;
- принцип Еджворта використовується якщо група складається з декількох коаліцій, кожній із яких не вигідно змінювати своє рішення. Знаючи переваги кожної з груп, можливим стає прийняття оптимального компромісного рішення, яке влаштує усіх.

В основі кількісних методів прийняття рішень лежить науково-практичний підхід, що припускає вибір оптимальних рішень за допомогою опрацювання великих масивів інформації.

В залежності від типу математичних функцій, покладених в основу моделей, можна виділити кілька методів рішень:

- лінійне моделювання;
- динамічне програмування (дозволяє вводити додаткові перемінні в процес рішення задач);
- ймовірні і статистичні моделі, реалізовані в методах теорії масового обслуговування;
- теорія ігор – моделювання ситуацій, ухвалення рішення в яких має враховувати різноманітність інтересів підрозділів;
- імітаційні моделі (дозволяють експериментально перевірити реалізацію рішень, змінити вихідні умови або уточнити вимоги до них).

Формалізація прийняття рішень підвищує ефективність керування в результаті зниження імовірності помилки та економії часу: нема необхідності щоразу розробляти нове рішення, коли виникає схожа ситуація. Тому керівництво організацій часто формалізує рішення для визначених, регулярно повторюваних ситуацій, розробляючи відповідні правила, інструкції або нормативи.

Важливим елементом при прийнятті рішень є кількість критеріїв вибору. Якщо вибір найкращої альтернативи визначається лише за одним критерієм (що характерно для формалізованих рішень), то прийняте рішення буде простим. І навпаки, коли обрана альтернатива повинна задовольняти одночасно декільком критеріям, рішення буде складним. На практиці використовується прийняття багатокритеріальних рішень, якщо для прийнятого рішення важливими є: обсяг прибутку, час отримання прибутку, рівень якості виконання проекту, рівень зайнятості працівників, термін реалізації тощо [16,17].

2.2 Огляд СППР, що використовують у ІТ

На даний момент не існує інформаційної системи для підтримки прийняття рішень при формуванні проектних команд, але є СППР для використання у ІТ-сфері, розглянемо деякі з них.

Система підтримки прийняття рішень "Експерт" – це інструмент для формалізації і рішення слабоструктурованих і неструктурованих завдань планування, прогнозування та управління.

Особливості системи "Експерт":

- система базується на сучасних методах підтримки прийняття рішень, що застосовувалися в США, Мексиці, Канаді та інших країнах для вирішення завдань аналітичного планування;
- підтримка як числових значень, так і суб'єктивних вербальних переваг користувача;
- можливість аналізу даних на предмет узгодженості та правдивості, виправлення неузгодженості;
- зручний графічний інтерфейс, інструменти для формалізації проблеми, аналізу результатів;
- можливість обробки будь-яких зовнішніх даних;
- обробка спільних суджень, досягнення консенсусу;
- докладні друковані звіти;
- наявність бібліотеки рішень типових задач в області фінансів, економіки, управління персоналом або підприємством і т.д.;
- низькі системні вимоги.

Power BI – комплексне програмне забезпечення бізнес-аналізу (Business Intelligence) компанії Microsoft, що об'єднує кілька програмних продуктів, які мають загальний технологічний і візуальний дизайн, з'єднувачі (шлюзи), а також web-сервіси. Power BI відноситься до класу self-service BI, дозволяє очищувати, консолідувати дані, перетворювати їх у зручний для аналізу формат, моделювати результат різних

варіантів дій та відстежувати результати прийняття рішень. Систем є частиною єдиної платформи Microsoft Power Platform.

Ключовий і найперший продукт лінійки – Power BI Desktop складається з трьох інтегрованих компонентів, що мають кожен свій інтерфейс:

- Power Query (редактор запитів) - виконує завантаження і очищення даних (ETL);
- PowerPivot (набори даних і моделі даних) - інтерфейс роботи з табличними даними в оперативній пам'яті де виконуються запити до даних, агрегація, розрахунки і т.д.;
- Power View - підсистема візуалізації і побудови звітів (Reporting).

Tableau – система інтерактивної аналітики, що дозволяє в найкоротші терміни проводити глибокий і різнобічний аналіз великих масивів інформації і не вимагає навчання бізнес-користувачів і дорогого впровадження.

Переваги Tableau:

- обробка даних будь-якого формату – від Excel до Oracle;
- швидка установка програмного продукту(90 секунд);
- не потребує тривалого впровадження;
- наявність готових галузевих рішень;
- висока швидкість отримання результату;
- низька вартість;
- інтуїтивно зрозумілий інтерфейс;
- створення будь-якого звіту усього за кілька кроків;
- скорочення часу аналізу даних;
- широкі можливості візуалізації інформації.

Загалом, усі СППР призначені для підтримки багатокритеріальних рішень у складному інформаційному середовищі. При цьому під багатокритеріальністю розуміється те, що результати прийнятих рішень оцінюються не по одному, а по сукупності багатьох показників (критеріїв), що розглядаються одночасно та є рівнозначно важливими.

Інформаційна складність систем визначається необхідністю врахування великого обсягу даних, обробка яких без допомоги сучасної обчислювальної техніки практично нездійсненна. У цих умовах число можливих рішень, як правило, дуже велике, і вибір найкращого з них «на око» без всебічного аналізу може призводити до дуже грубих помилок.

Система підтримки прийняття рішень має вирішувати дві основні задачі:

- вибір найкращого рішення з безлічі можливих (оптимізація);
- впорядкування можливих рішень по перевагу (ранжування).

В обох задачах принциповим моментом є вибір сукупності критеріїв, на основі яких в подальшому будуть оцінюватися і зіставлятися альтернативні рішення. СППР якраз й допомагає користувачеві зробити такий вибір.

Система управління персоналом – одна з найважливіших складових частин сучасного менеджменту. Основна мета таких систем – залучення та утримання цінних для підприємства кадрових фахівців. HRM (Human resource management)-системи вирішують дві головні задачі: упорядкування всіх облікових та розрахункових процесів, пов'язаних з персоналом, і зниження відсотка звільнення співробітників. Таким чином, HRM-системи в певному сенсі можна назвати «CRM-системами навпаки», що привертають і утримують не покупців, а власних співробітників компаній. Зрозуміло, методи тут застосовуються зовсім інші, але загальні підходи схожі.

Функції HRM-систем:

- пошук персоналу;
- підбір і відбір персоналу;
- оцінка персоналу;
- навчання і розвиток персоналу;
- управління корпоративною культурою;
- мотивація персоналу;
- організація праці.

2.3 Методи Data Mining

Якщо розглянути методи Data Mining, то вони поділяються на 2 великі групи за принципом роботи з вихідними даними – для такого поділу важливим є те, чи зберігаються дані після роботи методу у будь-якому вигляді (як звіти, шаблони, первні результати або чек-листи у бази знань, до пам'яті самого методу з метою самонавчання тощо).

Перша група – це безпосереднє використання даних, або їх збереження. Дані зберігаються в явному детальному вигляді і безпосередньо використовуються на стадіях прогностичного моделювання.

Друга група – це виявлення і використання формальних закономірностей. З вихідних даних витягується один зразок (шаблон) інформації, тому зберігаються не всі дані, а лише цей шаблон. У системі є можливим додавання критеріїв, тобто зміна розмірності набору даних. Через це, при зміні кількості критеріїв, необхідно буде перебудувувати модель даних, для чого є необхідним використання усіх даних, тому необхідно зберігати їх у базі [23].

Можна зробити висновок, що друга група не підходить для вирішення поставленої задачі, отже будемо розглядати першу.

Методами, які зберігають дані після опрацювання масивів інформації є:

– класифікація за допомогою дерев рішень. Перевагами даного методу є: інтуїтивність, можливість опису природною мовою, відсутність вибору користувачем вхідних параметрів. Недоліками можна вважати те, що чим більшою є кількість окремих випадків, описана в дереві рішень, тим менша кількість об'єктів потраплятиме в кожен окремий випадок. В результаті такого «переповнення» дерев їх здатність до узагальнення зменшується, тобто побудовані моделі можуть давати помилкові відповіді. У розглядаємії системі доволі багато параметрів, тому ризик отримати «переповнення» є цілком імовірним.

– класифікація методом опорних векторів. Перевагою цього методу є те, що для класифікації достатньо невеликої кількості даних. Однією з проблем є те, що не

завжди можна легко знайти лінійний кордон між двома класами. Рішенням проблеми є збільшення розмірності. Складність побудови SVM-моделі полягає в тому, що чим вище розмірність, тим складніше працювати з моделлю [22];

– класифікація за допомогою штучних нейронних мереж. Нейронна мережа працює тільки з числовими вхідними даними. Навчальна вибірка не повинна містити протиріч, бо нейронна мережа однозначно зіставляє вихідні значення вхідним. Так само слід враховувати фактор, що чим більша розмірність даних, тим більше часу буде потрібно на навчання мережі. Через то, що в роботі досить велика розмірність, цей метод також може давати не зовсім достовірні результати;

– класифікація методом k-найближчого сусіда. Метод «найближчого сусіда» (англійською «nearest neighbour») відноситься до класу методів, які зберігають дані для порівняння з новими елементами. При появі нового запису для прогнозування знаходяться відхилення між цим записом і подібними наборами даних, найбільш подібний елемент (або найближчий сусід) ідентифікується. Даний метод не створює ніяких моделей, узагальнюючих попередній досвід. Перевагою даного методу є простота використання, бо для роботи обирається лише один параметр k.

3 ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

Як і будь-яка фізична одиниця, проектна команда має свій життєвий цикл, що складається з певних кроків та етапів. Теоретично (при розгляданні ідеальної ситуації) кожна проектна команда переживає п'ять стадій по відношенню до проекту, а саме:

Стадія 1. Формування. Результативність команди на цьому етапі низька, тому що її члени ще не знайомі й не впевнені в роботі та професійних навичках один одного. Основні труднощі й перші "підводні камені" на етапі формування команди: особисті відчуття працівників, пов'язані з визначенням їхнього місця в команді та місця тимчасової команди всередині фірми; освідомлення людиною того, що він, колишній керівник, тепер є підлеглим, або навпаки – у керівника тимчасової команди раніше не було жодного досвіду керування. На цьому етапі проект-менеджер має докласти зусиль для подолання "підводних каменів" і об'єднання команди з орієнтацією на основну мету проекту.

Стадія 2. "Притирання" учасників. Під час того, як члени команди починають працювати разом, вони можуть стикатися з тим, що всі застосовують різні підходи, методи та технології в роботі над проектом. Такі розбіжності можуть спричинити суперечки та суттєві конфлікти, що аж ніяк не сприятиме підвищенню ефективності команди. Загальні проблеми на цьому етапі можна визначити так:

- "борсання на місці" без прогресу або навіть регрес;
- перекладання повноважень і відповідальності один на одного;
- зіткнення характерів (так як у команді можуть зустрітися найбільш розповсюджені характери, такі як: владолюбні учасники, неформальні, невизнані лідери, учасники-"мільні бульбашки", ледарі, занадто відповідальні люди та ті, хто боїться відповідальності);
- суперечки у команді з будь-якого приводу або, ще гірше, прийняття будь-яких думок більш досвідчених колег (лідів) без заперечень та обговорень інших існуючих варіантів.

Проте поступово, в разі вмілого та досвідченого керівництва, на основі загальних цінностей і норм у команді формуються ділові та дружні відносини, а також конструктивні неформальні групи, що визначають поведінські ролі членів команди, психологічний клімату групи.

Стадія 3. Нормальне функціонування. Час, коли всі суперечки й конфлікти залишилися позаду, кожен з членів команди зрозумів свою роль/місце в колективі, де він працюватиме протягом життєвого циклу проекту або протягом найближчих місяців. На двох перших етапах у групі мають сформуватися командне почуття та почуття загальної відповідальності, що вкрай необхідно для якісного досягнення мети. Це дає міцну основу, на якій команда може продуктивно працювати. Третій етап завжди триває найдовше з усіх етапів та є найрезультативнішим для проекту, отже кваліфікований проект-менеджер повинен використовувати його на максимум для того щоб встигнути зробити левову частку роботи.

Стадія 4. Реорганізація. Етап, коли приймається рішення кількісний або якісний склад команди з певних причин: необхідно менше/більше працівників певної спеціалізації для виконання конкретної задачі, зміни внаслідок корегування обсягів і видів робіт, необхідні заміни деяких працівників через їхню непридатність виконати нову роботу (з технічних або особистих причин) або їх неефективність, залучення нових спеціалістів з необхідними знаннями, запрошення тимчасових експертів, посилення команди тестувальників тощо.

Стадія 5. Розформування команди. Після завершення проекту команду розформовують, бо більше роботи над цим проектом не передбачується. При цьому можливі дві ситуації:

- у разі ефективності проекту та відповідної, «розумної» мотивації усі члени команди відчувають задоволення від своєї роботи, колектив вже сформований та усі його члени мають бажання працювати разом надалі на інших проектах. Як правило, якщо мова йде про компанії, де багато тимчасових проектів, проект-менеджер на новий проект запрошує до складу команди тих людей, з якими ефективно реалізував попередній проект;

– у разі провалу проекту члени команди залишають його з відчуттям моральної незадоволеності собою та командою в цілому, вони не хочуть продовжувати роботу у такому складі, бо не вірять у свою ефективність.

Головне завдання проект-менеджера під час усього життєвого циклу команд будь-якою ціною намагатися досягти першої ситуації, тобто створити ефективну команду та успішно завершити поточний проект.

У сучасних реаліях життєвий цикл команди дещо відрізняється від описаного, та в більшості випадків будується по відношенню до компанії, у якій працюють робітники, а не по відношенню до проекту. В такому випадку життєвий цикл виглядає приблизно так:

Стадія 1. Формування початкових проектних команд для N проектів.

Стадія 2. Робота над проектами.

Стадія 3. Перерозподіл учасників між проектами та формування нових команд.

Стадія 4. Робота над проектами.

Стадія 5. Перерозподіл учасників між проектами та формування нових команд.

Тобто, стадія 2 та 3 циклічно повторюються безліч разів, за умови що люди наймаються саме до компанії, а не на тимчасове виконання конкретного проекту. В такому випадку для кожного працівника може існувати статус “on the bench” – це означає “бути на лавці запасних” – знаходитися в очікуванні підходячого проекту, при цьому перебуваючи в штаті компанії та отримуючи заробітню плату (середню для своєї вакансії або таку, яка була в цій компанії на останньому проекті).

Питання формування команд все-таки залишається актуальним у будь-якому випадку, тож, перш за все, перед формуванням команди вирішуються питання по типу:

– для якої платформи створюють проект (для десктопу, мобайлу, вебу, буде це гра, сайт, додаток тощо) та що саме необхідно буде створити – лише серверну частину; лише клієнтську частину; і серверну, і клієнтську; чи необхідно буде тестувати продукт або це буде робити інше підприємство; або взагалі треба лише тестувати вже існуючий продукт тощо;

– як буде розроблятися проект;

– які механізми будуть використовуватися для підтримки проекту;

- чи має використовуватися певна мова програмування, патерни тощо (якщо так – які саме) ;
- яка відносна важливість вартісних витрат, часу і сил;
- як часто має бути реліз;
- яким чином буде оцінюватися прогрес роботи;
- хто буде оцінювати роботу, чи є конкретна аудиторія для продукту, що розробляється (якщо так – хто є цільовою аудиторією) ;
- як мають відбуватися мітинги, чи можуть вони бути «віддалено» за допомогою застосунків, що забезпечують голосовий або відеозв'язок.

Наступним кроком важливо розуміти, з якою іт-компанією матимемо справу – там де один офіс або кілька офісів, всі вони в одному місті, у одній країні або ж у різних країнах. Від того, чи можуть мітинги відбуватися «віддалено», приймається рішення чи можна формувати команду з працівників, що належать до різних офісів, міст або навіть країн.

Далі необхідно враховувати:

- яка кількість спеціалістів певної спеціалізації необхідна(розробники, тестувальники, девопси і т.д.),
- які мінімально припустимі фахові навички кандидатів,
- чи є мінімальний рівень працівників за ранжуванням кваліфікації працівника (джуніор, мідл, сіньор),
- чи є відгуки про кандидатів після участі у тренінгах або заходах, що передбачають роботу в команді (уміння кандидата адаптуватися до умов роботи у новій команді, взаємодія з командою і т.д.),
- визначити пріоритет та прибуток від вже існуючих проектів кандидатів, які підходять нам для роботи на новому проекті,
- розрахувати збитки або наслідки для фірми при знятті кандидата з проекту, на якому він працює.

Також визначимо, що важливими факторами є :

– фахові навички кандидата, які можуть мати рівень «Дуже добре», «Посередньо (або задовільно)», «Незадовільно». Найлогічніше, що компанію цікавлять кандидати з рівнем необхідних навичок не менш ніж «Посередньо»;

– можливість поєднувати в роботі однієї людини 2 посади на новому проекті (наприклад, QA+BA) з метою зменшення переведення співробітників з проекту на проект;

– достатньо важливий фактор – soft skills кандидата, бо процес адаптації для роботи нової команди має бути мінімальним, а сама робота – максимально ефективною. Можливо, якісь з обраних кандидатів уже мають досвід роботи в одній команді – тоді використовуваний метод має враховувати ефективність роботи цієї команди при форміванні нової;

– пріоритет проектів – виставляється в залежності від дедлайнів самого проекту – довготерміновий або тимчасовий (є певний дедлайн задачі проекту, наприклад, задача проекту на прод та подальша робота не буде продовжена). Пріоритет може бути високий, середній, низький. Теоретично, найвигідніше знімати людей з проектів з низьким пріоритетом, але це необхідно підтвердити або спростувати математично;

– необхідно розрахувати усі можливі наслідки для фірми при знятті людей з проектів – які збитки ми понесемо у грошовому еквіваленті, у вигляді людино-годин. Можливо, що грошовий збиток на одному проекті буде повністю перекритий заробітком на новому. Або є можливим беззбиткове збільшення часу виконання проекту (навіть з високим пріоритетом).

Потрібно зазначити, що на етапі підбору членів команди дуже важливим є усвідомлення необхідних для членів команди якостей особистості. До характерних рис бажаних членів команди відносять:

- професіоналізм;
- мотивацію;
- комунікабельність;
- відповідальність;

- вміння швидко та ефективно вирішувати існуючі проблеми та мінізувати появу нових з їх вини;
- неконфліктність та стресостійкість.

Слід зауважити, що розглядається формування проектних команд за умови наявності N_i вакансій, де $i = 1..m$, кожна з вакансій характеризується своїми частковими критеріями.

Визначимо часткові критерії для вакансій, за якими буде відбуватися підбір працівників в проектну команду:

- мінімально допустимий кваліфікаційний рівень кандидата;
- максимально допустимий узагальнений час виконання кандидатом задачі середньої складності;
- рівень володіння кандидатом t кількістю технологій (діапазон – t критеріїв, по 1 на кожну технологію);
- рівень часу роботи кандидата з t кількістю технологій (діапазон – t критеріїв, по 1 на кожну технологію);
- загальний досвід роботи на даній вакансії;
- рівень знання іноземної мови;
- оцінка стресостійкості кандидата;
- час роботи у поточній команді.

Отже, задачею кваліфікаційної роботи є оцінка ефективності двох методів при підтримці прийняття багатокритеріальних рішень у процесі формування проектних команд на K проектів, з урахуванням пріоритетності проектів, за умови, що є N_i вакансій, де $i = 1..m$, m – ціле число. Кожна з вакансій характеризується своїми частковими критеріями, які важливі та мають бути ураховані. При цьому необхідно максимізувати загальний прибуток від усіх проектів.

Актуальність задачі полягає в тому, що визначення найефективнішого методу має допомогти проект-менеджерам за мінімальний відрізок часу підібрати працездатну проектну групу, при цьому максимізувавши загальний прибуток від усіх проектів.

Критерієм оптимального формування проектних команд для всіх проектів є максимум сумарного прибутку від усіх проектів.

$$F = \sum_{i=1}^k E_i \rightarrow \max, \quad (3.1)$$

де k – загальна кількість проектів;

E_i – показник ефективності i -ого проекту.

$$E_i = \sum_{i=1}^k p_i - \sum_{i=1}^k c_i \rightarrow \max, \quad (3.2)$$

де p_i – прибуток від i -ого проекту;

c_i – матеріальні затрати на i -ому проекті.

4 РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ВИРІШЕННЯ ЗАДАЧІ

4.1 Розгляд задачі багатокритеріальної оптимізації

Задачу формування проектних команд будемо вважати задачею багатокритеріальної оптимізації, оскільки відбирати кандидатів на вакансії необхідно за великою кількістю критеріїв.

У загальному вигляді задача багатокритеріальної оптимізації формулюється таким чином: $\min_{\vec{x}} \{f_1(\vec{x}), f_2(\vec{x}), \dots, f_k(\vec{x})\}$, $\vec{x} \in S$, де $f_i: R^n \rightarrow R$ це k ($k \geq 2$) цільових функцій. Векторами розв'язку є $\vec{x} = (x_1, x_2, \dots, x_n)^T$, які належать до не порожньої області визначення S . Задача багатокритеріальної оптимізації полягає у пошуку вектора цільових змінних, який задовільняє накладеним обмеженням та оптимізує векторну функцію, елементи якої відповідають цільовим функціям. Ці функції утворюють математичне описання критерію задовільності та, зазвичай, взаємно конфліктують. Звідси, «оптимізувати» означає знайти такий розв'язок, за якого значення цільових функцій були б прийнятними для постановника задачі [11].

Розв'язування багатокритеріальної задачі ефективно при використанні методу послідовних поступок. Процедура розв'язування при виборі такого методу полягає у тому, що часткові критерії мають бути пронумеровані у порядку їхньої відносної важливості. Перший, а отже – найважливіший критерій максимізується, потім призначають величину припустимого зниження значення цього критерію і максимізують другий за важливістю частковий критерій за умови, що значення першого критерію не повинно відрізнятися від максимального допустимого значення більш ніж на величину встановленого зниження (поступки). Після цього знову призначають величину поступки, але вже другому критерію і знаходять максимум третього за важливістю критерію за умови, щоб значення у перших двох критеріїв не відрізнялися від раніше знайдених максимальних значень більш як на величини

відповідних поступок. На наступних кроках аналогічно по черзі використовуються інші часткові критерії.

Оптимальною зазвичай вважають будь-яку стратегію, що буде отримана під час розв'язування задачі пошуку умовного максимуму останнього за важливістю критерію. Отже, під час використання методу послідовних поступок багатокритеріальну задачу зводять до почергової максимізації часткових критеріїв і вибору величин поступок. Величини поступок характеризують відхилення пріоритету перших приватних критеріїв над іншими за такою умовою: чим менші поступки, тим жорсткіший пріоритет.

4.2 Розгляд задачі про призначення

Також, важливим є те, що такі задачі, як підбор кадрів або призначення кандидатів на вакантні позиції проекту найчастіше розглядаються як задача про призначення, яку можна сформулювати таким чином: необхідно виконати N різних робіт (у випадку, що розглядається – закрити N вакансій). Для їхнього виконання можна залучити не більше ніж N робітників. Кожний робітник за певну заробітню плату готовий зайняти одну вакансію, а деякі з них можуть зайняти 2 одночасно. Виконання будь-якої роботи (задач вакансії) варто доручити не більше ніж одному робітникові. Потрібно так розподілити роботи (вакансії) між робітниками, щоб загальні витрати на виконання всіх робіт були мінімальними, а прибуток від усіх проектів – максимальним.

Прийmemo кількість вакансій за N . Спочатку сформуємо задачу про призначення в стандартній формі – при цій формі передбачається, що кількість працівників дорівнює кількості вакансій, тобто працівників теж N .

Позначення: c_{ij} – показник ефективності призначення i -го працівника на j -й вакансії, наприклад витрати виконання i -м працівником j -ї роботи, що передбачає та чи інша вакансія; x_{ij} – змінна моделі ($x_{ij} = 1$, якщо i -й робітник використовується на j -й роботі, і $x_{ij} = 0$ – у протилежному випадку).

Модель задачі про призначення:

$$\sum_{j=1}^m \sum_{i=1}^m c_{ij} x_{ij} \rightarrow \min, \quad (4.1)$$

$$\sum_{i=1}^m x_{ij} = 1, j = 1..m, \quad (4.2)$$

$$\sum_{j=1}^m x_{ij} = 1, i = 1..m, \quad (4.3)$$

$$x_{ij} \geq 0, i = 1..m; j = 1..m. \quad (4.4)$$

- де (4.1) – це цільова функція зведення до мінімуму всіх витрат на проектах,
 (4.2) - (4.4) – система обмежень, що відображає наступне:
 (4.2) – умова, що кожна вакансія повинна бути закрита одним робітником;
 (4.3) – умова, що кожний робітник може займати лише одну вакансію;
 (4.4) – умови незаперечності змінних.

При рішенні задачі про призначення вихідною інформацією є таблиця задач про призначення $c = \{c_{ij}\}$, елементами якої служать показники ефективності призначень. Для задачі про призначення, записаної в стандартній формі, кількість рядків цієї таблиці збігається з кількістю стовпців (табл. 4.1).

Таблиця 4.1 – Матриця загального рішення для задачі про призначення

Вакансія	1	2	...	j	m
Працівник						
1	C_{11}	C_{21}	...	C_{1j}	...	C_{1m}
2	C_{21}	C_{22}	...	C_{2j}	...	C_{2m}
...
i	C_{i1}	C_{i2}	...	C_{ij}	...	C_{im}
...
m	C_{m1}	C_{m2}	...	C_{mj}	...	C_{mm}

Результатом рішення задачі про призначення (4.1) – (4.4) є вектор $x^* = \{x_{ij}^*\}$, компоненти якого – цілі числа.

Оптимальний план задачі про призначення (4.1) – (4.4) можна представити у вигляді квадратної матриці призначень, у кожному рядку й у кожному стовпці якої перебуває рівно одна одиниця. Таку матрицю іноді називають матрицею перестановок. Значення цільової функції (4.1), що відповідає оптимальному плану, називають ефективністю призначень.

Для ефективної роботи з даними (значеннями критеріїв) вони мають бути приведені до певного загального вигляду, тобто – нормалізовані.

Процес нормалізації даних – це приведення різномасштабних числових даних до єдиного діапазону значень. Загальноприйнятими діапазонами лінійної нормалізації прийнято вважати $[0; 1]$ або $[-1; 1]$. Нормалізація забезпечує однаковий вплив кожного значення параметра на вихідне значення розрахунків.

Приведення числових параметрів до цих діапазонів може бути здійснене за однією з наступних формул:

$$k_{Ni} = \frac{k_i - k_{min}}{k_{max} - k_{min}}, \quad (4.5)$$

$$k_{Ni} = 2 \cdot \frac{k_i - k_{min}}{k_{max} - k_{min}} - 1, \quad (4.6)$$

де k_i – поточне значення параметру;

k_{min} – мінімальне значення параметру;

k_{max} – максимальне значення параметру;

k_{Ni} – значення параметру, що відповідає заданому діапазону.

При лінійній нормалізації найчастіше значення параметрів приводяться до діапазону $[0; 1]$, таким чином вважається, що значення параметру описує показник – якою мірою в об'єкті проявляється (є важливим) деякий параметр. Таким чином найменшому (найменш важливому) значенню вибірки даних відповідатиме значення 0, а найбільшому (найважливішому) – 1. Приведення значень параметрів до діапазону $[-1; 1]$ застосовується у тих ситуаціях, коли числове значення параметру кількісно

описує прояв в одному об'єкті двох протилежних параметрів. Таким чином 1 або -1 використовується, якщо в об'єкті проявляється лише один з параметрів, а 0 – якщо обидва параметри проявляються однаково [20].

Формально, постановку задачі формування команди можна представити у вигляді:

$$|N| = n \leq n_0, \quad (4.7)$$

де N – склад команди,

N_0 – множина кандидатів на зайняття ролі в команді.

За умови, що $|N_0|=n_0$.

Функціональність команди та її ефективність відображається її складом, який має мати найбільший з можливих потенціалів для найпродуктивнішої реалізації мети[9].

$F(N)$ – функціонал ефективності, що дорівнює кожному з можливих складів команди. Реальний склад команди знаходять за формулою:

$$N^* = \arg \max F(N). \quad (4.8)$$

4.3 Огляд методів числового оцінювання альтернатив

Для отримання експертних оцінок переваг можуть бути застосовані різні методи, в залежності від характеру ситуації, що має бути проаналізована та яка служить основою для прийняття рішень. Якщо треба одержати кількісні оцінки відносної переваги між критеріями – застосовуються методи числового оцінювання від безпосередніх числових оцінок до найдосконаліших.

Ці оцінки можна одержати за допомогою числового оцінювання альтернатив: методів Черчмена-Акоффа та Терстоуна, теореми Фон Неймана-Моргенштерна.

Для числового оцінювання якості альтернатив достатньо часто застосовують метод Черчмена-Акоффа. Спочатку альтернативи ранжуються за важливістю (альтернативи $x_1..x_n$ є відсортованими у певному порядку, тобто $x_1 > x_2 > \dots > x_n$). Експерт вказує попередні числові оцінки $f(x_1), \dots, f(x_n)$, при цьому найкраща з альтернатив отримує оцінку 1, а оцінки інших коливаються в межах від 0 до 1. Надалі експерт порівнює x_1 та сумарну дію альтернатив $x_2..x_n$. Якщо x_1 виявиться важливішею за цю суму, експерт має скоригувати оцінки таким чином, щоб виконувалось співвідношення $f(x_1) > \sum_{i=2}^n f(x_i)$. Якщо станеться ситуація, що альтернатива x_1 менш важлива за суму всіх інших альтернатив, то для уточнення її послідовно порівнюють із сумами, що обчислюються за допомогою $\sum_{i=2}^k f(x_i), k \geq 2$ відповідно до порядку зменшення значення k з $n - 1$ до 2. Як тільки буде отримано перше k , для якого виконується умова $f(x_1) \geq \sum_{i=2}^k f(x_i)$ альтернатива x_1 має бути виключена з розгляду. Аналогічно корегуються числові оцінки для усіх альтернатив. У випадку, якщо альтернатив надто багато, для зменшення трудомісткості виконання методу вони можуть бути розбиті на групи, при цьому одна з альтернатив має бути включеною до складу кожної з груп, щоб була можливість отримати числові оцінки для всіх альтернатив за допомогою оцінювання всередині кожної групи.

У методі Терстоуна результати попарних порівнянь є первинною інформацією для визначення числових оцінок, а оцінка кожної з альтернативи є випадковою величиною, яка розподілена за нормальним законом із математичним сподіванням M_i та дисперсією D_i . Різниця оцінок – випадкових величин $f(x_i)$ та $f(x_j)$ розподіляється за нормальним законом, при цьому математичне сподівання приймає значення $M_{ij} = M_i - M_j$, а дисперсія приймає значення $D_{ij} = D_i + D_j + 2r_{ij}\sqrt{D_i D_j}$, де r_{ij} – коефіцієнт кореляції між $f(x_i)$ та $f(x_j)$. Результатом виконання цього методу є значення $M_i \in \{1, 2, \dots, n\}$, що обираються як числові оцінки альтернатив за значеннями частот S_{ij} , де S_{ij} – частота вибору альтернативи x_i як пріоритетнішої, відносно альтернативи x_j .

Метод Фон Неймана-Моргентшерна виходить зі стохастичного характеру значень оцінок, у якому для визначення оцінок використано суміші ймовірностей.

Прийнятим є те, що експерт може вказати для довільної альтернативи x_j , що є гіршою за x_i але кращою, ніж x_1 , таке число p з умовою ($0 \leq p \leq 1$), що альтернатива x_j буде еквівалентна «змішаній» альтернативі $px_i, (1 - p)x_1$. Таку альтернативу можна реалізувати за допомогою стохастичного вибору, коли альтернатива x_i обирається з певною імовірністю p , в той час, коли для альтернативи x_1 імовірність становить $(1 - p)$. Якщо ймовірність p більш наближена до 1, приймається, що має виконуватися умова $x_j < [px_i, (1 - p)x_1]$. У випадку, коли ймовірність прямує до 0 – має виконуватися умова $x_j > [px_i, (1 - p)x_1]$. Оцінювати альтернативи у вигляді одного числа слід з підвищеною обережністю, бо для достатньо складних альтернатив таке оцінювання є суттєво недостатнім. Під час практичного застосування методу Фон Неймана-Моргентшерна необхідно враховувати ті обмеження, що накладаються методами отримання експертної оцінки. При цьому, для попередньої орієнтовної оцінки альтернатив доцільно застосовувати будь-які методи числового оцінювання.

4.4 Огляд методів, ефективність яких буде оцінюватися

Метод опорних векторів – це метод аналізу даних, що використовується для класифікації та аналізу за допомогою моделей, пов'язаних з алгоритмами навчання, що носять назву опорно-векторних машин (ОВМ або ж SVM – від англ. Support Vector Networks), який набув популярності у використанні лише кілька років тому. Модель ОВМ є уявленням зразків(критеріїв) як точок у просторі, що відображені таким чином, що зразки(критерії), що відносяться до окремих категорій розділено широким функціональним зазором. Нові зразки відображуються у цьому ж просторі та робиться прогнозування щодо їх належності до однієї з категорій на основі того, на який бік зазору вони потраплять.

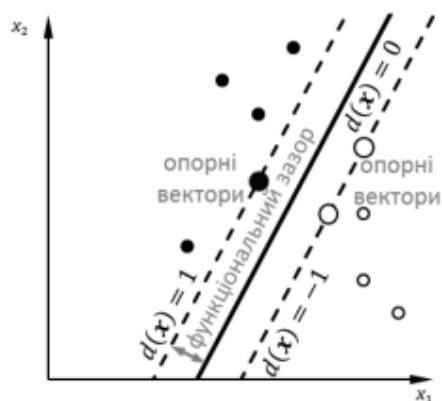


Рисунок 4.1 – Роздільна функція $d(x) = 0$, яка сформована за допомогою методу опорних векторів

SVM можуть ефективно виконувати як лінійну, так і нелінійну класифікацію при застосуванні так званого «ядрового трюку», неявно відображуючи свої входи до просторів ознак високої вимірності.

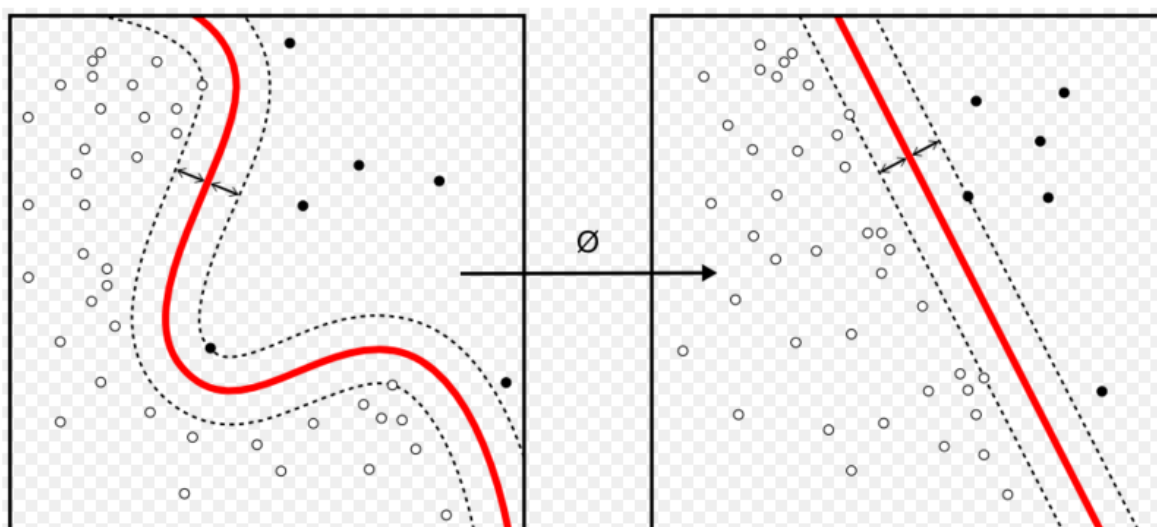


Рисунок 4.2 – Розподіл за допомогою «ядрового трюку»

Формально, SVM будує певну гіперплощину або набір гіперплощин у просторі високої (або навіть нескінченної) вимірності, які можна використовувати для класифікації, регресії та інших задач. Теоретично, найкращий розподіл досягається гіперплощиною, що має найбільшу відстань до найближчих точок

тестових(навчальних, тєнувальних) даних будь-якого з класів, оскільки в загальному випадку чим більшим є розділення, тим нижче похибка узагальнення класифікатора.

Первинну тестову(навчальну) задачу можна сформулювати у скінченновимірному просторі, але на практиці часто трапляється так, що множини, які треба розрізняти, не можуть бути лінійно розділеними (тобто такими, які можна розділити на відокремлені групи за допомогою проведення прямих) в цьому просторі. Саме з цієї причини було запропоновано відображувати первинний скінченновимірний простір відносно простору набагато вищої вимірності, тобто роблячи розділення простішим у останньому. Для збереження рівного обчислювального навантаження, відображення, які використовуються методом опорних векторів, розробляються таким чином, щоб вони забезпечували можливість простого обчислення скалярних добутків у рамках змінних первинного простору, визначаючи їх у рамках ядрових функцій $k(x, y)$, які обираються окремо до кожної задачі [27].

Гіперплощини в просторі вищої вимірності визначаються як геометричне місце точок, скалярні добутки яких з вектором в обраному просторі є сталими. Вектори, що визначають кожен з гіперплощин можуть бути обрані як лінійні комбінації з параметрами a_i відображень векторів ознак x_i , які можна зустріти в базі даних. За умови такого вибору гіперплощини, точки x простору ознак, що мають проєкцію на гіперплощині, визначаються відношенням $\sum_i a_i k(x_i, x) = const$. Слід зауважити, що якщо $k(x, y)$ зменшується з послідовним віддаленням y від x , то кожен член цієї суми вимірює ступінь близькості пробної точки x до відповідних основних точок даних x_i . Таким чином, наведена вище сума ядер може використовуватися для вимірювання відносної близькості кожної пробної точки до точок даних, які походять з однієї або іншої з множин, які треба розрізняти. Важливим є те, що множина точок x , що може бути відображена на будь-яку гіперплощину, може бути в результаті доволі вигнутою, створюючи ситуацію, що буде набагато важче проводити розрізнення між множинами, які не були опуклими в первинному просторі.

Ядровий трюк дозволяє функціям діяти в неявному просторі ознак високої вимірності навіть без обчислення координат даних у цьому просторі, натомість просто обчислюючи скалярний добуток зображень всіх пар даних у цьому просторі ознак.

Переваги методу опорних векторів:

- а) один з найшвидших методів знаходження вирішальних функцій;
- б) метод можна звести до вирішення задачі квадратичного програмування у випуклій області, яка завжди матиме одне рішення;
- в) роздільна смуга, знайдена методом, матиме значення максимальної ширини, що дозволить надалі зробити кращу класифікацію;
- г) великий клас нейронних мереж може бути представлений у вигляді методу опорних векторів з визначеними ядрами;
- г) теоретичне обґрунтування: кінцеве правило методу обирається не за допомогою евристик, а відповідно до оптимізації деякої функції.

Недоліки методу опорних векторів:

- а) доволі мала кількість параметрів для налаштування: після того як ядро зафіксували, єдиним варіативним параметром буде лише коефіцієнт помилки C ;
- б) метод чутливий до шумів та до стандартизації даних;
- в) не існує загального підходу до автоматичного вибору ядра у випадку лінійної нероздільності класів;
- г) відносно повільне навчання методу.

Метод k -найближчих сусідів (від англ. k -nearest neighbor method) – це простий непараметричний класифікаційний метод, де для класифікації об'єктів у рамках простору властивостей використовуються евклідові відстані, порашовані по відношенню до усіх інших об'єктів. Обираються усі об'єкти, до яких відстань є найменшою – ці об'єкти створюють новий окремий клас.

У класичному розгляді методу має бути задано параметр k – кількість сусідів, що розглядатимуться при класифікації нового об'єкту. Під час фази класифікації обирається 1 новий об'єкт, відповідність якого до будь-якого з існуючих класів ще не задана. Для нього визначають k найближчих (у сенсі обраної метрики), вже

класифікованих спостережень. Після цього обирається клас, якому належить більшість з k найближчих прикладів-сусідів – саме до цього класу відноситиметься новий об'єкт, що класифікується.

Розглянемо найпростішу реалізацію роботи алгоритму за допомогою рис.4.3. Кружечком зі знаком питання зображено новий об'єкт, що має бути класифікований та віднесений до одного з двох класів – А або В. Якщо оберемо $k = 3$, то з трьох найближчих об'єктів два належатимуть класу А та один – класу В, тобто об'єкт з позначкою '?' буде привласнений до класу А. Якщо ж розглянути ситуацію, де $k = 5$, то з п'яти найближчих сусідів два будуть об'єктами класу А, та три – об'єктами класу В, отже і новий об'єкт буде ідентифіковано як складова класу В.

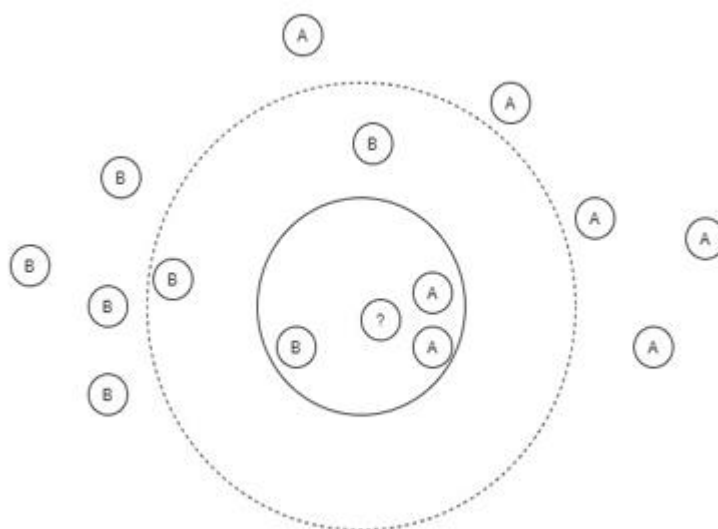


Рисунок 4.3 – Тестові дані для наочного пояснення роботи алгоритму k -найближчого сусіда

У найідеальнішому та найпростішому випадку клас нового об'єкту визначається за допомогою простого вибору класу, який найчастіше зустрічається серед з k об'єктів-сусідів. Але, як вже зрозуміло, на практиці це не завжди вдале рішення, бо частота появи для двох або більше класів може бути однаковою або кількість класів

може бути занадто великою. В такому випадку використовується деяка функція, за допомогою якої визначається клас, що називається функцією поєднання.

У звичайному випадку використовують просте невважене голосування, при якому передбачається, що всі k прикладів мають однакове право «голосу» незалежно від відстані до нового об'єкта, що класифікується. Однак, максимально логічним є припущення, що чим далі приклад розташований від класифікованого об'єкту в просторі ознак, тим нижче його важливість для класифікації. Для покращення результатів класифікації вводять зважування прикладів залежно від їхньої віддаленості – тобто використовують зважене голосування, в основі якого знаходиться введення «штрафу» для класу, залежно від того, наскільки приклади, які відносяться до нього, віддалені від класифікованого об'єкту. Такий «штраф» представляється як сума величин, обернених до квадрату відстаней від прикладу j -го класу до об'єкта, що має бути класифікований (у деяких джерелах це значення може назватися показником близькості): $Q_i = \sum_{j=1}^{n_j} \frac{1}{D^2(x, a_{ij})}$, де D – оператор обчислювання відстані, x – вектор ознак об'єкта, що класифікується, a_{ij} – i -тий об'єкт j -ого класу.

При такому розрахунку перевага надається тому класові, величина Q_i для якого буде найбільшою, при цьому вірогідність того, що деякі з класів отримають однакову оцінку значно зменшується.

Обрання параметру k є важливим для отримання коректних результатів при класифікації. Якщо значення параметра занадто мале, може виникати ефект перенавчання, коли рішення щодо класифікації приймається з урахуванням малого числа прикладів та має низьку значимість. Якщо встановити $k = 1$, то алгоритм буде ідентифікувати будь-який новий об'єкт як об'єкт класу найближчого з сусідів.

Крім того, слід враховувати, що взагалі використання досить малих значень k збільшує вплив шумів на результати класифікації, коли навіть невеликі зміни у даних призводять до великих змін у результатах класифікації. Але при цьому межі між класами є більш вираженими (клас при голосуванні перемагає зі значно більшим рахунком).

Та навпаки, якщо значення параметра k буде занадто великим, то під час класифікації буде брати участь багато об'єктів, що потенційно відносяться до різних класів. Така класифікація виявляється дуже грубою і погано відбиває локальні особливості набору даних. Таким чином, вибір параметра k є компромісом між точністю та узагальнюючою здатністю моделі. Також при великих значеннях параметра k зменшується шумування результатів класифікації, але знижується вираженість меж між класами. У більшості завдань бінарної класифікації є доцільним обирати k як непарне число, оскільки це дозволить уникнути рівності «голосів» відносно того, до якого класу віднести новий об'єкт.

Переваги методу k -найближчих сусідів:

а) алгоритм стійкий до аномальних викидів, тому що імовірність влучення нетипового запису до числа k -найближчих сусідів нікчемно мала. Якщо ж така ситуація все-таки відбулася, то вплив на голосування (особливо зважене або при $k > 2$), швидше за все, буде незначним – отже, незначним буде й вплив на загальний підсумок класифікації;

б) програмна реалізація цього алгоритму доволі проста;

в) результат роботи алгоритму легко інтерпретований, отже експертам з різних областей зрозуміла логіка роботи алгоритму, заснована на пошуку схожих об'єктів;

г) алгоритм можна модифікувати за допомогою використання найбільш придатних функцій сполучення і метрик, що дозволяє змінювати алгоритм під конкретну задачу.

Недоліки методу k -найближчих сусідів:

а) набір даних, що використовується для алгоритму, має бути репрезентативним;

б) модель існує лише разом з даними: для класифікації нового прикладу необхідно використовувати всі існуючі приклади – саме ця особливість значно обмежує використання алгоритму.

5 ДОСЛІДЖЕННЯ ТА АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ

Для того щоб була можливість оцінити ефективність роботи двох методів слід взяти до уваги 2 пункти:

а) час виконання методу має бути мінімально можливим (а отже й кількість кроків методу має бути мінімізована);

б) порівнювати результати роботи алгоритмів слід з завчасно визначеною тестовою вибіркою, щоб розуміти, наскільки наближеним до ідеального є результат.

5.1 Проведення дослідження для методу опорних векторів

Почнемо з розгляду роботи алгоритму методу опорних векторів на лінійно розділеній вибірці за допомогою рис.5.1 та виборки ірисів. Великі чорні точки є опорними для вектору, саме вони допомагають побудувати площину, за допомогою якої можна поділити всі спостереження на 2 класи. В залежності від розташування значення відносно площини воно буде віднесено до одного з класів.

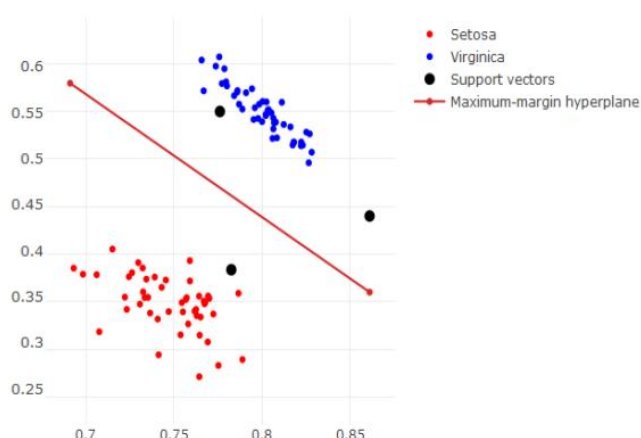


Рисунок 5.1 – Приклад роботи методу опорних векторів на лінійно роздільній вибірці

У реальному житті дуже рідко зустрічається лінійно розділена виборка, частіше можна зіткнутися з мультикласовою вибіркою, у якій одне спостереження може належати до декількох класів одночасно. Алгоритм методу опорних векторів допомагає вирішувати задачі бінарної класифікації, а отже його можна адаптувати до мультикласових задач.

У 1992 році Бернхардом Босером, Ізабелою Гійон та Володимиром Вапником було запропоновано спосіб створення нелінійного класифікатора, в основі якого знаходиться перехід від скалярних значень до довільних ядер. Результуючий алгоритм має багато збігів з алгоритмом лінійної класифікації, але кожне зі скалярних значень у формулах лінійної класифікації замінюється нелінійною функцією ядра (скалярним значенням у просторі з більшою розмірністю). Саме у цьому просторі зможе існувати оптимальна роздільна гіперплощина. Так як розмірність одержуваного простору може бути значно більшою за розмірність вихідного, то перетворення, що зіставляє скалярні значення, буде нелінійним – отже функція, яка відповідає у вихідному просторі оптимальній роздільній гіперплощині, також буде нелінійною. У випадку, коли вихідний простір має доволі високу розмірність, то вибірка може бути роздільною лінійно. Найбільш поширені ядра та результат роботи з їх допомогою зображена на рис 5.2.

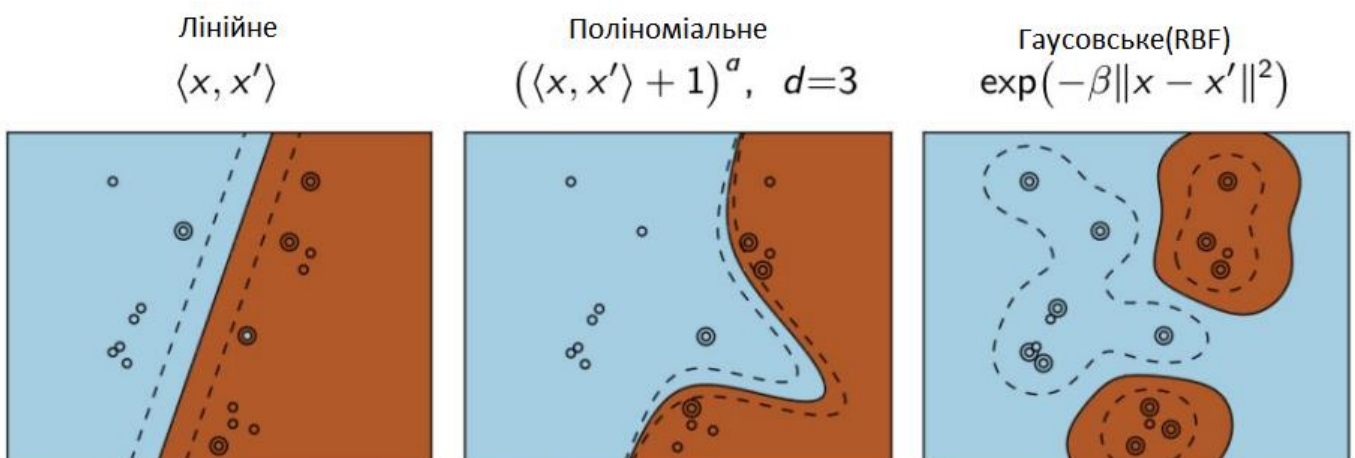


Рисунок 5.2 – Приклад роботи методу опорних векторів за допомогою лінійного, поліноміального та гаусовського ядер

Розглянемо рішення задачі за допомогою гаусівського ядра. Слід зазначити, що при такому ядровому трюці, SVM допомагає отримати схожий результат, як у випадку якби додавалася б велика кількість ознак близькості без їх фактичного додавання. Модель даних представлена у верхньому лівому кутку на рис. 5.3. На інших графіках на цьому рис. зображені моделі, навчені з різними значеннями параметрів γ і C . Збільшення значення γ призводить до звуження дзвоноподібної кривої, внаслідок чого область впливу кожного зі зразків зменшується, тобто межа рішень стає більш нерівномірною та заломлюється поблизу індивідуальних значень. Та навпаки, зменшення значення γ робить дзвоноподібну криву ширше, тому зразки мають велику область впливу, а межа рішень виявляється більш гладкою. Можна зробити висновок, що γ діє аналогічно гіперпараметру регуляризації: якщо модель перенавчається, тоді необхідно зменшити значення параметру γ , а якщо недонавчається – то збільшити його (подібно до гіперпараметру C).

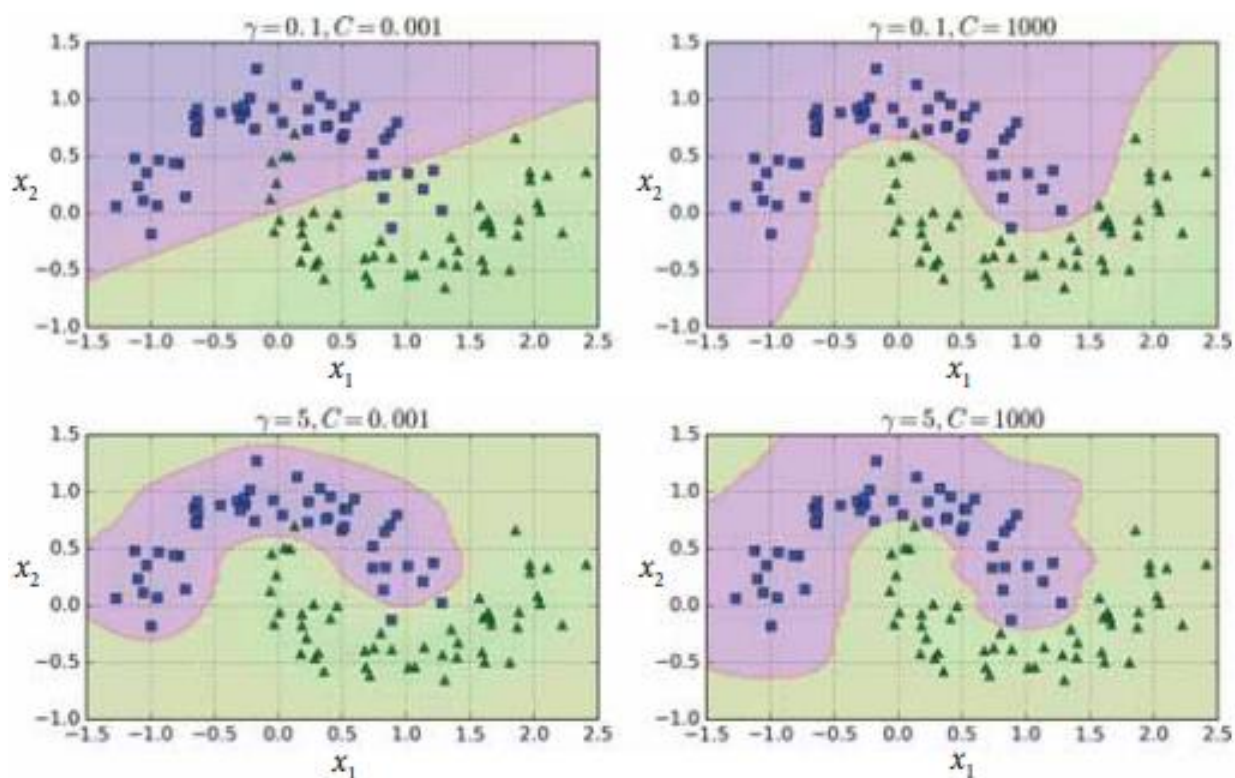


Рисунок 5.3 – Класифікатори машини опорних векторів, що використовують гаусівське ядро, в залежності від значень параметрів γ і C

5.2 Проведення дослідження для методу k-найближчих сусідів

Перевірка коректності роботи алгоритму методу k-найближчих сусідів полягає у тому, що необхідно провести тестування на стандартній вибірці, наприклад на Ірисах Фішера, яка доволі часто використовується для демонстрації роботи алгоритмів класифікації. Якщо досліджувати характер статистичної варіації ознак вибірки Ірисів Фішера з використанням діаграм, де дані категоризовані відповідно до окремих видів бачимо (з рис.5.4), що більш вираженою класифікаційною ознакою є розміри пелюстки *Petal.Width*, а не ширина чашолистка.

З рис.5.5 візуалізуються дані, що аналізуються, у просторі згідно до двох перших головуючих компонент. Алгоритм передбачення для методу k-NN (k-ближчих сусідів) будується за принципом "більшості голосів", тобто результатом роботи конкретного кроку є мітка класу-переможця. Об'єкт "зірочка", що має бути класифікований, потрапляє в осередок об'єктів класу "трикутник" (*versicolor*) та при $k = 1$ буде відноситися до цього класу. Однак варто лише змінити значення параметру k з 1 на 3 – по голосах двох найближчих сусідів з трьох тестованих новий об'єкт буде віднесено до класу "круг" (*setosa*). Імовірнісний варіант методу k-NN використовує для ранжирування передбачуваних класів суму "голосів" сусідів з урахуванням їх ваг, зокрема, важливими є евклідове значення відстані між об'єктом, що тестується, та його сусідами.

Варіант 1-NN забезпечує 100% коректне розпізнавання прикладів навчальної вибірки (адже найближчим сусідом до об'єкту є він сам), але нерідко помиляється на невідомих йому даних. При збільшенні значення параметру k до деяких меж якість розпізнавання на контрольній вибірці зростатиме, влесе, як і варіативність алгоритму. Оптимальне значення k для коректного прогнозування може бути знайдено за допомогою використання перехресної перевірки, для цього за фіксованим значенням k будується модель k-ближчих сусідів і оцінюється CV-помилка класифікації. Ці дії повторюються циклічно багато разів для різних значень k , при цьому значення, що

відповідатиме розміру найменшої помилки розпізнавання, буде прийняте за оптимальне.

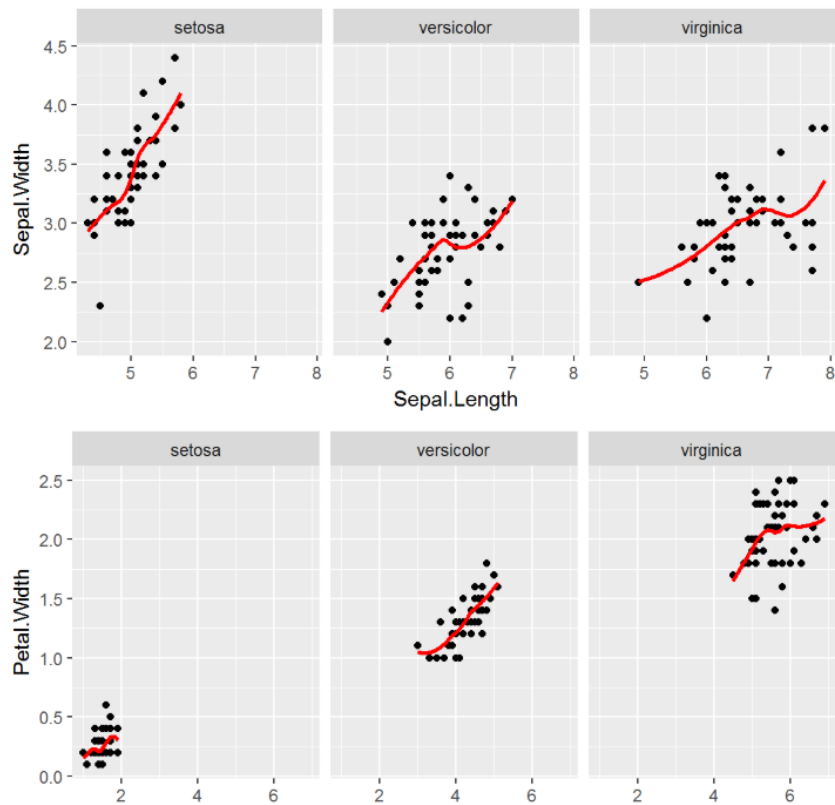


Рисунок 5.4 – Категоризована діаграма вибірки Ірисів Фішера

На рис.5.5 зображено результат кластеризації вибірки Ірисів Фішера.

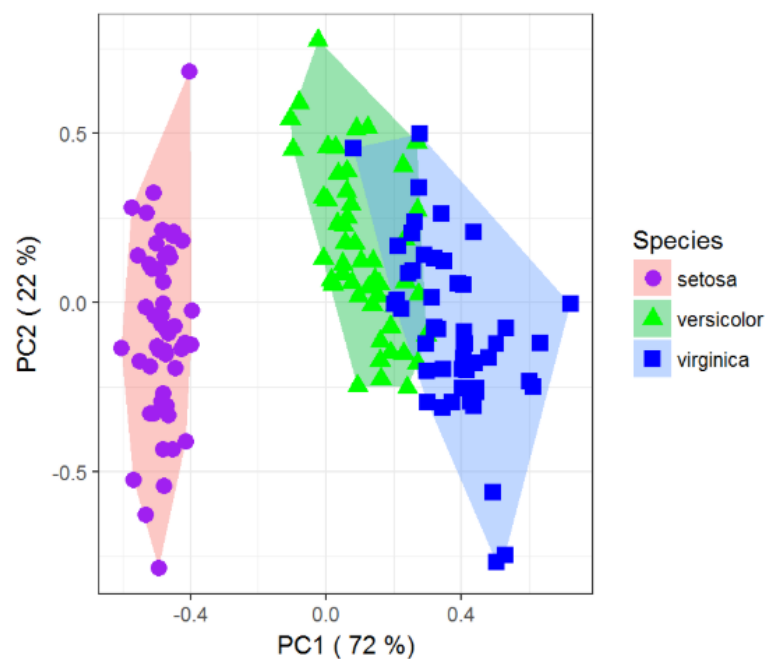


Рисунок 5.5 – Ординаційна діаграма даних по Ірисам Фішера в осях двох головуючих компонент

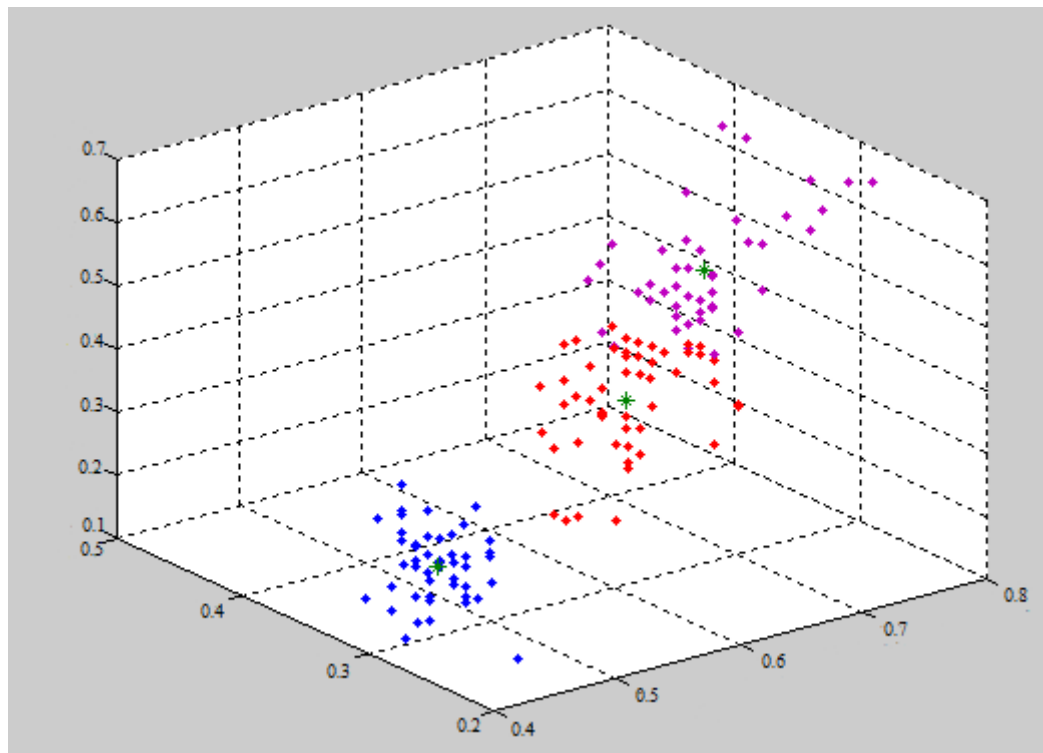


Рисунок 5.6 – Результат кластеризації Ірисів Фішера за допомогою методу kNN

5.3 Підготовка тестових даних

При проведенні досліджень щодо формування проектних команд двома методами була використана тестова вибірка даних, що складається з 20 кандидатів, по 4 кожної основної спеціалізації, з них 4 кандидати мають додаткову спеціалізацію (рис. 5.6) та є завчасно визначені найкращі варіанти проектних команд, сформовані за допомогою наближення до мінімізації витрат на заробітну платню працівників, але в той же час наближення до максимізації загальної ефективності команди (рис. 5.7).

Команди мають бути сформовані так, щоб у кожній з них були присутні кандидати кожної з основних спеціалізацій, а також, кандидат що має додаткову спеціалізацію. Загальна ефективність команди має бути максимально можливою, але слід пам'ятати про мінімізацію витрат на заробітну плату усієї команди.

Так як у тестовій вибірці не дуже велика кількість критеріїв то нехтувати деякими з них, приводячи значення до 0 не є логічним, виключенням є критерій другої (не основної) спеціалізації (бізнес-аналітик) та наявність вищої освіти.

Отже, нормалізація проводитиметься відносно найбільшого з критеріїв у кожній з основних спеціалізацій (за 1 визначено найбільше значення, а інші пораховані у процентному співвідношенні до цього значення), а не відносно різниці найбільшого та найменшого з критеріїв, як зазначено у формулі (4.5).

Для коректного проведення розрахунків відсортуємо тестову вибірку за основними спеціалізаціями та заробітною платою в рамках цих спеціалізацій для зручності огляду та нормалізуємо дані, тобто приведемо значення всіх критеріїв (будь-якого типу даних), що використовуються, до числового значення у діапазоні від 0 до 1. Для параметрів «Досвід роботи» та «Мінімальна заробітна плата» нормалізація проводиться окремо в рамках кожної зі спеціалізацій.

Для .Net Developer максимальним за часом є досвід роботи 1 рік та 6 місяців, отже приймемо його за 1, а усі інші параметри будемо порівнювати з ним (для спрощення переведемо значення у кількість місяців, отримаємо 18). Для спеціалізації Automation QA максимальним досвідом є термін 2 роки та 8 місяців (або ж 32 місяці), для Java Developer – 5 років та 4 місяці (64 місяці), для JS Developer – 3 роки та 4 місяці (40 місяців), для Manual QA – 3 роки та 8 місяців (а4 місяці), а для BA – 1 рік та 7 місяців (19 місяців). Результат нормалізації значень для досвіду роботи наведено у табл.5.1.

При нормалізації заробітної плати найбільш вигідним з точки зору роботодавця є працівник, який згоден працювати за меншу суму грошей, отже нормалізація проводиться від зворотнього – 1 присвоюється мінімальному грошовому значенню в рамках основної спеціалізації, а інші значення розраховуються як співвідношення мінімального значення до поточного, результати наведені у табл.5.2.

Відносно параметрів «Рівень знання іноземної мови», «Стресостійкість», «Темп роботи» основні спеціалізації не є важливими, так як ці параметри однаково важливі в рамках усієї команди. Те ж саме стосується параметру «Вища освіта» – вона або є, або її нема, нормалізований параметр приймає значення 0 або 1.

Рівень володіння мовою у тестовій вибірці приймає одне з чотирьох значень: В1, В2, С1, С2. При цьому 0 не буде присвоєно жодному з рівнів, крок при розподілі нормалізованого значення приймемо за 0,25 (табл.5.3).

Таблиця 5.1 – Нормалізовані значення для досвіду роботи (за спеціалізаціями)

ПІБ	Основна спеціалізація	Нормалізований параметр	Додаткова спеціалізація	Нормалізований параметр
Matveeva A.G.	.Net Developer	0,5	–	0
Andrukhovich M.F.	.Net Developer	0,61	–	0
Bragin A.A.	.Net Developer	0,94	–	0
Antonov A.P.	.Net Developer	1	–	0
Kudryavtseva M.F.	Automation QA	0,156	–	0
Savenko V.F.	Automation QA	0,28	BA	1
Sharov T.P.	Automation QA	0,5	BA	0,16
Baranova V.M.	Automation QA	1	BA	0,26
Belyaeva T.B.	Java Developer	0,39	–	0
Bobylev R.G.	Java Developer	0,31	–	0
Medyanik F.P.	Java Developer	1	–	0
Kalashnikova S.V.	Java Developer	0,95	–	0
Semenov A.D.	JS Developer	0,1	–	0
Lobanova O.A.	JS Developer	0,525	–	0
Pakhomov Y.G.	JS Developer	0,775	–	0
Ermakov G.D.	JS Developer	1	–	0
Andruschenko I.D.	Manual QA	0,07	–	0
Sokolov A.V.	Manual QA	0,14	–	0
Borisenko V.V.	Manual QA	0,59	–	0
Likhachev T.K.	Manual QA	1	BA	0,21

Стресостійкість та Темп роботи мають лише 3 значення: для стресостійкості це може бути High, Average або Low, для темпу роботи – Quickly, Medium або Slow. Найвище (або найшвидше) значення прирівнюється до 1, 0 у цих категоріях не присвоюється жодному зі значень.

Таблиця 5.2 – Нормалізовані значення для заробітної плати робітників (за спеціалізаціями)

ПІБ	Основна спеціалізація	Мінімальна заробітня плата (\$)	Нормалізований параметр
Matveeva A.G.	.Net Developer	650	1
Andrukhovich M.F.	.Net Developer	700	0,92
Bragin A.A.	.Net Developer	1400	0,464
Antonov A.P.	.Net Developer	1600	0,406
Kudryavtseva M.F.	Automation QA	350	1
Savenko V.F.	Automation QA	1300	0,27
Sharov T.P.	Automation QA	1400	0,25
Baranova V.M.	Automation QA	2200	0,16
Belyaeva T.B.	Java Developer	1800	1
Bobylev R.G.	Java Developer	2000	0,9
Medyanik F.P.	Java Developer	3700	0,487
Kalashnikova S.V.	Java Developer	3700	0,487
Semenov A.D.	JS Developer	550	1
Lobanova O.A.	JS Developer	1700	0,32
Pakhomov Y.G.	JS Developer	2000	0,275
Ermakov G.D.	JS Developer	2100	0,26
Andruschenko I.D.	Manual QA	300	1
Sokolov A.V.	Manual QA	450	0,67
Borisenko V.V.	Manual QA	1700	0,177
Likhachev T.K.	Manual QA	2200	0,14

Таблиця 5.3 – Нормалізовані значення для рівнів володіння іноземною (англійською) мовою

Рівень володіння мовою	Нормалізований параметр
B1	0,25
B2	0,5
C1	0,75
C2	1

Таблиця 5.4 – Нормалізовані значення для рівнів стресостійкості

Рівень володіння мовою	Нормалізований параметр
High	1
Average	0,66
Low	0,33

Таблиця 5.5 – Нормалізовані значення для швидкості темпу роботи

Рівень володіння мовою	Нормалізований параметр
Quickly	1
Medium	0,66
Slow	0,33

Значення загальної нормалізації усіх критеріїв та їх суму (значення ефективності кожного з кандидатів) можна побачити на рис. 5.8.

FullName	Specialization	Experience		Training	English	Stress resistance	Min salary(\$)	Higher education	Pace of work
Matveeva Alla Gennadievna	.Net Developer		9m		C1	High	650	-	Medium
Antonov Arseny Pavlovich	.Net Developer		1y 6m		B1	Low	1600	+	Medium
Andrukhovich Mark Fedorovich	.Net Developer		11m	NewStar	B1	High	700	-	Slow
Bragin Anton Alekseevich	.Net Developer		1y 5m		C1	High	1400	-	Slow
Sharov Timur Platonovich	Automation QA	BA	1y 4m	3m	NewStar	B1	1400	-	Medium
Baranova Victoria Maksimovna	Automation QA	BA	2y 8m	5m		B2	2200	+	Quickly
Savenko Vladislav Fedorovich	Automation QA	BA	9m	1y 7m		B1	1300	+	Slow
Kudryavtseva Marina Filippovna	Automation QA		5m		B1	Low	350	-	Slow
Bobylev Roman Grigorievich	Java Developer		1y 8m		TryIT-2019	B2	2000	-	Medium
Belyaeva Tatyana Bronislavovna	Java Developer		2y 1m		B1	Average	1800	+	Quickly
Medyanik Fedor Petrovich	Java Developer		5y 4m		B2	Low	3700	+	Quickly
Kalashnikova Sofya Valerievna	Java Developer		5y 1m		B2	High	3700	+	Quickly
Ermakov German Danilovich	JS Developer		3y 4m		B2	High	2100	+	Quickly
Pakhomov Yuri Grigoryevich	JS Developer		2y 7m		C1	Average	2000	+	Medium
Lobanova Olga Anatolievna	JS Developer		1y 9m		TryIT-2019	C1	1700	+	Medium
Semenov Andrii Denysovich	JS Developer		4m		B1	Average	550	-	Slow
Borisenko Veronika Valerievna	Manual QA		2y 2m		TryIT-2019	B2	1700	+	Medium
Andruschenko Inna Dmytrivna	Manual QA		3m		B2	Low	300	+	Medium
Sokolov Artemy Vitalievich	Manual QA		6m		C1	High	450	+	Medium
Likhachev Timur Konstantinovich	Manual QA	BA	3y 8m	4m		C2	2200	+	Quickly

Рисунок 5.6 – Тестова вибірка даних кандидатів

Varied selection of the best project teams:

Team 1:

Matveeva Alla Gennadievna
Baranova Victoria Maksimovna
Bobylev Roman Grigorievich
Semenov Andrii Denysovich
Sokolov Artemy Vitalievich

Payroll costs: 5850\$
Efficiency : 18.76

Team 3:

Andrukhovich Mark Fedorovich
Kudryavtseva Marina Filippovna
Belyaeva Tatyana Bronislavovna
Lobanova Olga Anatolievna
Likhachev Timur Konstantinovich

Payroll costs : 6750\$
Efficiency : 18.401

Team 2:

Bragin Anton Alekseevich
Savenko Vladislav Fedorovich
Kalashnikova Sofya Valerievna
Ermakov German Danilovich
Andruschenko Inna Dmytrivna

Payroll costs : 8800\$
Efficiency : 20.531

Team 4:

Antonov Arseny Pavlovich
Sharov Timur Platonovich
Medyanik Fedor Petrovich
Pakhomov Yuri Grigoryevich
Borisenko Veronika Valerievna

Payroll costs : 10400\$
Efficiency : 17.82

Рисунок 5.7 – Завчасно сформовані найкращі варіанти проектних команд на основі тестової вибірки даних

FullName	Specialization	Experience			Training	English		Stress resistance		Min salary(\$)		Higher education		Pace of work		Efficiency	
Matveeva Alla Gennadievna	.Net Developer	9m	0,5	0		C1	0,75	High	1	650	1	-	0	Medium	0,66	3,91	
Andrukhovich Mark Fedorovich	.Net Developer	11m	0,61	0	NewStar	B1	0,25	High	1	700	0,92	-	0	Slow	0,33	3,11	
Bragin Anton Alekseevich	.Net Developer	1y 5m	0,94	0		C1	0,75	High	1	1400	0,464	-	0	Slow	0,33	3,484	
Antonov Arseny Pavlovich	.Net Developer	1y 6m	1	0		B1	0,25	Low	0,33	1600	0,406	+	1	Medium	0,66	3,646	
Kudryavtseva Marina Filippovna	Automation QA	5m	0,156	0		B1	0,25	Low	0,33	350	1	-	0	Slow	0,33	2,066	
Savenko Vladislav Fedorovich	Automation QA	BA 9m	0,28	1y 7m	1	B1	0,25	Average	0,66	1300	0,27	+	1	Slow	0,33	3,79	
Sharov Timur Platonovich	Automation QA	BA 1y 4m	0,5	3m	0,16	NewStar	B1	0,25	Low	0,33	1400	0,25	-	0	Medium	0,66	2,15
Baranova Victoria Maksimovna	Automation QA	BA 2y 8m	1	5m	0,26	B2	0,5	High	1	2200	0,16	+	1	Quickly	1	4,92	
Belyaeva Tatyana Bronislavovna	Java Developer	2y 1m	0,39	0		B1	0,25	Average	0,66	1800	1	+	1	Quickly	1	4,3	
Bobylev Roman Grigorievich	Java Developer	1y 8m	0,31	0	TryIT-2019	B2	0,5	High	1	2000	0,9	-	0	Medium	0,66	3,37	
Medyanik Fedor Petrovich	Java Developer	5y 4m	1	0		B2	0,5	Low	0,33	3700	0,487	+	1	Quickly	1	4,317	
Kalashnikova Sofya Valerievna	Java Developer	5y 1m	0,95	0		B2	0,5	High	1	3700	0,487	+	1	Quickly	1	4,937	
Semenov Andrii Denysovich	JS Developer	4m	0,1	0		B1	0,25	Average	0,66	550	1	-	0	Slow	0,33	2,34	
Lobanova Olga Anatolievna	JS Developer	1y 9m	0,525	0	TryIT-2019	C1	0,75	Average	0,66	1700	0,32	+	1	Medium	0,66	3,915	
Pakhomov Yuri Grigoryevich	JS Developer	2y 7m	0,775	0		C1	0,75	Average	0,66	2000	0,275	+	1	Medium	0,66	4,12	
Ermakov German Danilovich	JS Developer	3y 4m	1	0		B2	0,5	High	1	2100	0,26	+	1	Quickly	1	4,76	
Andruschenko Inna Dmytrivna	Manual QA	3m	0,07	0		B2	0,5	Low	0,33	300	1	+	1	Medium	0,66	3,56	
Sokolov Artemy Vitalievich	Manual QA	6m	0,14	0		C1	0,75	High	1	450	0,67	+	1	Medium	0,66	4,22	
Borisenko Veronika Valerievna	Manual QA	2y 2m	0,59	0	TryIT-2019	B2	0,5	Average	0,66	1700	0,177	+	1	Medium	0,66	3,587	
Likhachev Timur Konstantinovich	Manual QA	BA 3y 8m	1	4m	0,21	C2	1	Average	0,66	2200	0,14	+	1	Quickly	1	5,01	

Рисунок 5.8 – Нормалізовані значення критеріїв для тестової вибірки даних, загальна ефективність кожного з кандидатів

5.4 Аналіз отриманих результатів

Після повної підготовки тестової вибірки було написано алгоритми для кожного з двох методів, що порівнюються, та пропущено тестові дані крізь них. У результаті для методу опорних векторів отримано такі результати команд, що зображені на рис. 5.9, а результати для методу k-ближчих сусідів – на рис.5.10. Можна зробити висновок, що найближчим за складом команд, загальною ефективністю та витратами на заробітну платню виявився метод k-ближчих сусідів, тоді як за методом опорних векторів на 1 місці опинилася команда з найбільшою сумарною ефективністю, але ця ж команда є майже найбільш витратною за загальною сумою мінімально допустимих заробітних плат.

Varied selection of the best project teams using support vectors method:

Team 1:

Matveeva Alla Gennadievna
Baranova Victoria Maksimovna
Kalashnikova Sofya Valerievna
Ermakov German Danilovich
Andruschenko Inna Dmytrivna

Payroll costs : 8950\$
Efficiency : 22.807

Team 2:

Bragin Anton Alekseevich
Kudryavtseva Marina Filippovna
Medyanik Fedor Petrovich
Lobanova Olga Anatolievna
Likhachev Timur Konstantinovich

Payroll costs : 9350\$
Efficiency : 18.792

Team 3:

Andrukhovich Mark Fedorovich
Sharov Timur Platonovich
Belyaeva Tatyana Bronislavovna
Pakhomov Yuri Grigoryevich
Sokolov Artemy Vitalievich

Payroll costs : 6350\$
Efficiency : 17.9

Team 4:

Antonov Arseny Pavlovich
Savenko Vladislav Fedorovich
Bobylev Roman Grigorievich
Semenov Andrii Denysovich
Borisenko Veronika Valerievna

Payroll costs : 7150\$
Efficiency : 16.733

Рисунок 5.9 – Найкращі варіанти проектних команд на основі тестової вибірки даних, сформовані за допомогою методу опорних векторів

Varied selection of the best project teams using k-nearest neighbors method:

Team 1:

Matveeva Alla Gennadievna
Baranova Victoria Maksimovna
Belyaeva Tatyana Bronislavovna
Semenov Andrii Denysovich
Sokolov Artemy Vitalievich

Payroll costs : 5650\$

Efficiency : 19.69

Team 3:

Andrukhovich Mark Fedorovich
Kudryavtseva Marina Filippovna
Bobylev Roman Grigorievich
Pakhomov Yuri Grigoryevich
Likhachev Timur Konstantinovich

Payroll costs : 7250\$

Efficiency : 17.676

Team 2:

Bragin Anton Alekseevich
Savenko Vladislav Fedorovich
Kalashnikova Sofya Valerievna
Ermakov German Danilovich
Andruschenko Inna Dmytrivna

Payroll costs : 8800\$

Efficiency : 20.531

Team 4:

Antonov Arseny Pavlovich
Sharov Timur Platonovich
Medyanik Fedor Petrovich
Lobanova Olga Anatolievna
Borisenko Veronika Valerievna

Payroll costs : 10100\$

Efficiency : 17.61

Рисунок 5.10 – Найкращі варіанти проектних команд на основі тестової вибірки даних, сформовані за допомогою методу k-ближчих сусідів

ВИСНОВКИ

У ході виконання кваліфікаційної роботи було поглиблено знання з предметної області, розглянуто способи та сфери застосування систем підтримки прийняття рішень, розглянуто деякі з існуючих СППР-аналогів для обраної предметної області, проаналізовано методи Data Mining та методи, що використовуються для рішення багатокритеріальних задач, розглянуто загальні рішення задач багатокритеріальної оптимізації, задач класифікації та задач про призначення.

Наступним кроком було визначено, ефективність яких саме методів буде розглянута у кваліфікаційній роботі, розглянуто загальну суть трьох методів для числового оцінювання альтернатив. Обрано способи нормалізації параметрів для тестової вибірки даних та для масштабних даних, які допомагають спростити процес прийняття рішень шляхом зведення параметрів будь-якого типу даних до числового діапазону у межах від 0 до 1, де 0 – найменш пріоритетний варіант, а 1, відповідно, найбільш пріоритетний.

Проведено аналіз ефективності методів опорних векторів та k -ближчих сусідів при формуванні проектних команд, як результат доведено що останній метод більш точний та ефективний для вирішення даної задачі. Основним недоліком методу опорних векторів є те, що при наявності великої кількості параметрів не завжди є можливість з легкістю знайти кордони між класами, отже похибка при класифікації збільшується прямо пропорційно до збільшення кількості параметрів. При роботі з великими моделями за допомогою цього методу можуть виникати суттєві похибки, а як наслідок – результати спотворюються і робота значно ускладнюється. Так як тестова вибірка має не таку велику кількість параметрів, а різниця між результатами методів уже значна та помітна – можна зробити висновок, що метод опорних векторів є менш ефективним ніж метод k -ближчих сусідів для вирішень задач класифікації та задач про призначення.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Івановська К.А. Порівняння ефективності методів при підтримці прийняття рішень у процесі формуванні проектних команд. *25-й Міжнародний молодіжний форум «Радіоелектроніка та молодь у XXI столітті»*. Зб. матеріалів форуму, т. 6. Харків: ХНУРЕ, 2021. С. 240-241.
2. Івановська К.А. Оцінка ефективності методів при підтримці прийняття рішень у процесі формуванні проектних команд. *XIV Міжнародна науково-практична конференція «Інформаційні технології та автоматизація - 2021»*. Зб. матеріалів форуму, Одеса, 21-22 жовтня 2021 р., Одеса, Видавництво ОНАХТ, 2021 р., С. 310-312.
3. Самойленко В. В. Менеджмент трудового потенціалу підприємства та його іноваційна перспектива. Економічний розвиток в сучасному просторі та часі, 2010, № 4, С. 440.
4. Бідюк П. І., Коршевнік Л.О. Проектування комп'ютерних інформаційних систем підтримки прийняття рішень: Навчальний посібник. Київ: ННК "ІПСА" НТУУ "КПІ", 2010. 350 с.
5. Ларичев О. И., Петровский А. В. Системы поддержки принятия решений. Современное состояние и перспективы их развития. *Итоги науки и техники. Сер. Техническая кибернетика*, 1987. №21. С. 131-164
6. Герасименко И. Я. Социометрия конфликта: монография / И.Я. Герасименко, Р.М. Юсупов; Рос. акад. наук. Санкт-Петербургский ин-т информатики и автоматизации, Санкт-Петербург: [б. и.], 1994, 76 с.
7. Чекинов Г.П. Применение ситуационного управления в информационной поддержке принятия решений при проектировании организационно-технических систем: Научный журнал. Москва, 2003. № 2. С. 14-19
8. Marappan S. Multi Resource Constrained Project Scheduling Problem by Adaptive Genetic & Decision Support System, 2016, Т. 6., Вып. 12, С. 520-534.

9. Золотовицкий Р. А. Организационная арттерапия и тренинг: социодрама и социометрия в работе с организациями / Роман Золотовицкий, Москва: Морено-Институт, 2003, 208 с.
10. Трахтенгерц Э.А. Компьютерная поддержка принятия решений. Москва: Наука, 2002. 365 с.
11. Вишнеков А.В., Курилова Н.С., Сафонова И.Е., Штейнберг В.И. Многоцелевые задачи принятия проектных решений: Учебное пособие. Москва: МГИЭМ, 2002. 101 с.
12. Вишнеков А.В. Методы принятия проектных решений в CAD/CAM/CAE системах электронной вычислительной техники: Учебное пособие. Москва: МГИЭМ, 1999. 78 с.
13. Катренко А.В., Пасічник В.В., Пасько В.П. Теорія прийняття рішень. Київ: Видавнича група ВНУ, 2009. 448 с.
14. Ромащенко В.Н. Принятие решений: ситуации и советы. Киев, 2012. 154 с.
15. Подиновский В.В. Введение в теорию важности критериев в многокритериальных задачах принятия решений: Учебное пособие. Москва: Физматлит, 2007. 64 с.
16. Черняховская Л.Р. Поддержка принятия решений при стратегическом управлении предприятием на основе инженерий знаний. Уфа: АН РБ, Гилем, 2010. 128 с.
17. Евланов А. Г. Теория и практика принятия решений. Москва: Экономика, 2010. 212 с.
18. Кривко О. Б. Информационные технологии. Москва: СОМИНТЭК. 2011. 179 с.
19. Мескон М. Основы менеджмента: Учебное пособие. Москва, 2012. 387 с.
20. Пирожков В. А. О реализации процессного подхода к управлению в виде системы поддержки принятия решений «Управление деятельностью организации»: Вестник Тамбовского ун-та. Сер.: Гуманитарные науки. 2008. Вып. 11. 489 с.

21. Румянцева З. П. Менеджмент организации: учебное пособие. Москва: ИНФРА-М, 2005. 432 с.
22. Holsapple C. W., Whinston A. B. Decision Support Systems: A Knowledge-based Approach. Minneapolis: West Publishing Co., 1996.
23. Коваленко И. И., Бидюк П. И., Баклан И. В. Системный анализ и информационные технологии в управлении проектами. К.: Экономика и право, 2001. 270 с.
24. Артюхов В.В. Общая теория систем: Самоорганизация, устойчивость, разнообразие, кризисы. Изд. 3-е. М.: Книжный дом «ЛИБРОКОМ», 2012. 224 с.
25. Бидюк П.І. Комп'ютерні системи підтримки прийняття рішень: Навчальний посібник. Миколаїв: Вид-во ЧДУ ім. Петра Могили, 2012. 380 с.
26. ДСТУ 3008-2015. Документація. Звіти в сфері науки і техніки. Структура і правила оформлювання. Чинний від 2017-07-01. Київ: ДП «УкрНДНЦ», 2016. 26 с.
27. Press, William H.; Teukolsky, Saul A.; Vetterling, William T.; Flannery, B. P. (2007). Section 16.5. Support Vector Machines. Numerical Recipes: The Art of Scientific Computing (вид. 3rd). New York: Cambridge University Press.
28. Вейстроффер, HR, та Li, Y, "Програмне забезпечення для аналізу кількох критеріїв", Ch 29 у: Greco, S, Ehrgott, M і Figueira, J, eds, Багатокритеріальний аналіз рішення: Серія опитувань, Спрінгер: Нью-Йорк, 2016.
29. Grebennik I., Ovezgeldyyev A., Hubarenko Y., Hubarenko M. (2020) Information Technology Reengineering of the Electricity Generation System in Post-disaster Recovery. In: Murayama Y., Velez D., Zlateva P. (eds) Information Technology in Disaster Risk Reduction. ITDRR 2019. IFIP Advances in Information and Communication Technology, vol 575. Springer, Cham.