НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
Інститут прикладного системного аналізу

Кафедра математичних методів системного аналізу

«На правах рукопису»
УДК 519.226

«До захисту допущено»
Завідувач кафедри
_____ О.Л. Тимощук
«___»_____20__ р.

Магістерська дисертація

на здобуття ступеня магістра

зі спеціальності 124 Системний аналіз

на тему: «Система оцінки кредитоспроможності фізичних осіб з використанням методів регресійного аналізу»

Виконав:
Студент (ка) ІІ курсу, групи КА-62м
Бакун Сабіна Антонівна _____

Керівник:
к.т.н., ст.викладач каф.ММСА
Терентьєв О.М. _____

Рецензент:
д.т.н., проф.
Теленик С.Ф. _____

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студент _____

Київ
2018

РЕФЕРАТ

Магістерська дисертація: 107 с., 32 рис., 32 табл., 5 додатків, 19 джерел.

Актуальність теми: в Україні бурхливо зростає ринок споживчого кредитування. Проте, разом з цим, зростає і кількість неповернених кредитів, що наносить досить великі збитки банківським установам. Таким чином, розробка та застосування систем оцінки кредитоспроможності фізичних осіб у процесі прийняття рішення щодо видачі кредиту є актуальними на сьогоднішній день.

Мета даної роботи полягає у дослідженні та вдосконаленні існуючих методик побудови скорингових моделей та розробці системи підтримки прийняття рішень для оцінювання кредитоспроможності фізичних осіб з використанням методу логістичної регресії.

Об’єктом дослідження є набір статистичних даних щодо наданих банком споживчих кредитів фізичним особам.

Методи дослідження: метод логістичної регресії, метод максимальної правдоподібності, метод градієнтного спуску, операції над матрицями.

Програмний продукт реалізований за допомогою мови програмування C# у середовищі розробки Microsoft Visual Studio 2012. Для порівняльного аналізу отриманих результатів були побудовані моделі у вигляді дерев рішень і скорингової карти в системі SAS Enterprise Miner.

Отримані результати: розроблено систему підтримки прийняття рішень для прогнозування кредитоспроможності фізичних осіб з використанням методу логістичної регресії та методу максимальної правдоподібності. Запропоновано спосіб використання категоріальних даних в регресійних моделях.

КРЕДИТОСПРОМОЖНІСТЬ, КРЕДИТНИЙ СКОРИНГ, ЛОГІСТИЧНА РЕГРЕСІЯ, МЕТОД МАКСИМАЛЬНОЇ ПРАВДОПОДІБНОСТІ, ЗАГАЛЬНА ТОЧНІСТЬ МОДЕЛІ, ІНДЕКС GINI

ABSTRACT

Theme: “System for evaluating the solvency of individuals using regression analysis methods”.

Master's thesis explanatory note: 107 p., 32 fig., 32 tab., 5 appendices, 19 sources.

Actuality: the consumer lending market is growing rapidly in Ukraine. However, along with this, the number of unreturned loans is increasing, which causes quite large losses to banking institutions. Thus, the development and application of systems for assessing the creditworthiness of individuals in the process of making a decision on the issuance of a loan are actual for today.

The purpose of this work is to study and improve existing methods of constructing scoring models and to develop a decision support system for assessing the creditworthiness of individuals using the method of logistic regression.

The object of the study is a set of statistical data on consumer loans provided by the bank to individuals.

Methods of research: logistic regression method, maximum likelihood method, gradient descent method, operations on matrices.

The software product was implemented using the C# programming language in the Microsoft Visual Studio 2012 development environment. For a comparative analysis of the results were built models as decision trees and scorecard in the SAS Enterprise Miner system.

Obtained results: a decision support system was developed for predicting the creditworthiness of individuals using the logistic regression method and the maximum likelihood method. The method of using categorical data in regression models is proposed.

CREDITWORTHINESS, CREDIT SCORING, LOGISTIC REGRESSION, METHOD OF MAXIMUM LIKELIHOOD, COMMON ACCURACY OF THE MODEL, INDEX GINI

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

CA (англ. Common Accuracy) – загальна точність моделі

IV (англ. Information Value) – інформаційне значення

MAE (англ. Mean Absolute Error) – середня абсолютна похибка

MSE (англ. Mean Squared Error) – середньоквадратична похибка

WOE (англ. Weight of evidence) – зважена сукупність

БД – база даних

ЛІМ – лінійна імовірнісна модель

ММП – метод максимальної правдоподібності

ОПР – особа, яка приймає рішення

ПЕОМ – персональна електронно-обчислювальна машина

ПП – програмний продукт

СППР – система підтримки прийняття рішень

ВСТУП

У період стабільного розвитку в Україні спостерігалось збільшення банківського споживчого кредитування, що спричинило розвиток ще не дуже відомого напрямку прикладного математичного моделювання для нашої країни – кредитного скорингу. В умовах політичної й економічної кризи коректне вирішення задач виявлення кредитоспроможних клієнтів перетворюється для фінансових установ в завдання прямого виживання на ринку кредитних послуг [1]. Незважаючи на те, що кредитування фізичних осіб є одним з найбільш прибуткових видів банківської діяльності, воно водночас є найбільш ризикованим, оскільки багато банків зіштовхуються з проблемою неповернення виданих фізичним особам кредитів [2]. У зв'язку з цим, є критичними застосування та розробка нових більш досконалих методів оцінювання кредитних ризиків і кредитоспроможності осіб за умови нинішніх кризових явищ у банківській сфері.

Ефективним заходом, що зменшує ризик фінансових установ та дозволяє оптимально вирішувати завдання оцінювання кредитоспроможності клієнтів, є кредитний скоринг. Кредитний скоринг – математична або статистична модель, яка допомагає банківським установам визначити ймовірність повернення кредиту потенційним клієнтом у встановлений строк [3]. Скорингові моделі можуть бути отримані методами лінійної регресії, логістичної регресії, дискримінантного аналізу, дерев рішень, нейронних мереж та іншими.

Кредитний скоринг допомагає оптимізувати процес прийняття рішень щодо надання споживчих кредитів, що й визначає актуальність даного дослідження.

Таким чином, ціллю даної роботи є дослідження методів побудови скорингових моделей та їх використання для оцінювання кредитних ризиків фінансових установ.

Для досягнення даної мети були вирішені наступні задачі:

- проаналізовано існуючі рішення для бізнес аналітики у сфері кредитних ризиків;
- проведено огляд та аналіз існуючих математичних методів моделювання і прогнозування кредитних ризиків;
- розроблено архітектуру системи підтримки прийняття рішень для аналізу, моделювання та прогнозування ймовірності повернення наданого споживчого кредиту;
- розроблено програмний продукт, в якому реалізовано алгоритм логістичної регресії.

Практичним результатом роботи є розроблений програмний продукт, який дозволяє оцінювати кредитоспроможність фізичних осіб на основі методів інтелектуального аналізу даних.

Робота складається з 4 розділів. В першому розділі розглядаються поняття кредитоспроможності та кредитного скорингу, проводиться огляд програмного забезпечення, призначеного для банківської клієнтської аналітики. У другому розділі вивчаються методи побудови скорингових систем та процес попереднього аналізу і обробки даних. У третьому розділі дисертації описується розроблена СППР, надається інструкція з експлуатації програмного продукту, а також проводиться порівняння результатів роботи системи з іншими методами. Четвертий розділ присвячено розробленню стартап-проекту на основі створеного програмного продукту.

РОЗДІЛ 1 ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

Кредити складають найбільшу долю від прибуткових банківських активів, що є основною частиною доходів банку. За рахунок цього джерела формується левова частина чистого прибутку, яка йде в резервні фонди і з якої виплачуються дивіденди акціонерам. Однак кредитування окрім високих доходів несе в собі і високі ризики. Оскільки кредитна діяльність супроводжується підвищеним ризиком, кредитні операції залишаються найбільш ризикованим компонентом активів банківських установ. Якщо комерційні банки будуть вести надто ризиковану кредитну політику, це може призвести до їх банкрутства [4].

На сьогоднішній день банки потребують безперервного вдосконалення методології ведення своєї кредитної діяльності. У зв’язку з цим, аналіз і управління кредитним ризиком є досить актуальним за сучасних тенденцій розвитку банківського сектору з метою зменшення рівня ризику кредитних операцій.

Основною технікою мінімізації ризику кредитної діяльності банків є оцінювання кредитоспроможності клієнтів. Аналіз їх кредитоспроможності дозволяє позбутися невиправданих ризиків при прийнятті рішень щодо видачі кредитів. Точність оцінювання є не менш важливою для позичальника, оскільки від цього залежать умови кредитування та розмір позики.

1.1 Поняття кредитоспроможності та скорингу

Основним критерієм, що формує кредитні відносини між банківською установою та потенційним клієнтом, є кредитоспроможність позичальника. Саме визначена банком кредитоспроможність позичальника є необхідною умовою для укладення кредитного договору і дає можливість визначити фактори, які впливатимуть на невиплату кредиту.

Кредитоспроможність – наявність у позичальника передумов для видачі кредиту і його здатність повернути борг в обумовлені договором строки та у повному обсязі [5].

Кредитоспроможність позичальника визначається за такими його характеристиками як:

а) здатність своєчасно розраховуватися за раніше одержаними кредитами;

б) поточне фінансове становище;

в) спроможність мобілізації коштів з інших джерел.

Кредитний скоринг – це метод оцінки кредитоспроможності людини, який на основі статистичних даних по кредитній історії банку оцінює ймовірність неповернення коштів потенційним позичальником, виходячи з його соціально-демографічних ознак, таких як вік, стать, освіта, посада, трудовий стаж, термін проживання в регіоні тощо. Маючи базу даних виплачених (гарних) і не виплачених (поганих) кредитів, банківська установа за допомогою статистичного аналізу має змогу виявити основні фактори, що впливають на можливість позичальника повернути борг. Загально прийнято, що існує кореляція між певними соціальними даними і надійністю позичальника.

З огляду на специфіку наявної інформації існують чотири види кредитного скорингу (рис. 1.1):

а) application-scoring (аплікаційний скоринг) – оцінка кредитної спроможності клієнта, для прийняття рішення щодо можливості видачі йому кредиту;

б) behavioral-scoring (поведінковий скоринг) – оцінка ймовірності повернення вже наданих кредитів. Здійснюється в межах кредитного періоду з метою виявлення ризиків дефолту та прийняття запобіжних заходів щодо зниження цих ризиків. Скорингові моделі дозволяють спрогнозувати зміну кредитоспроможності клієнта, завдяки чому можна контролювати оптимальний ліміт по кредитній карті;

в) collection-scoring (колекторський скоринг) – оцінка можливості повернення кредиту (повного або часткового) позичальником при порушенні ним термінів погашення заборгованості. Здійснюється після закінчення кредитного періоду з метою прийняття певних заходів щодо повернення кредиту. За статистикою 40% не погашених вчасно платежів припадають на позичальників, що просто забули внести платіж по кредиту;

г) fraud-scoring (скоринг проти шахраїв) – статистична оцінка ймовірності шахрайських дій з боку потенційного позичальника. При цьому вважається, що до 10% неповернень по кредитах пов’язані з відвертим шахрайством і цей показник зростає.

Рисунок 1.1 – Види кредитного скорингу

Найбільш поширеним серед українських банків є лише аплікаційний скоринг, зазвичай це пояснюється застарілістю вітчизняних систем скорингу та зависокими цінами розробників аналітичних скорингових моделей.

В залежності від методу прийняття рішення та якості доступної інформації про потенційного позичальника скоринг можна поділити на емпіричний (empirical credit scoring) та дедуктивний (deductive credit scoring) [6].

Основою для дедуктивних систем є висновки та оцінки експертів, відповідно до критичних значень і ваг оцінок окремих критеріїв. Порівняно з емпіричним скорингом, ці системи вважаються менш адекватними з огляду впливу людського фактору і ризику суб'єктивності оцінювання експертом. У свою чергу, основними елементами емпіричних систем скорингу є статистичні моделі та методи ранжирування неоднорідних багатовимірних даних. При цьому вибір методу класифікації значною мірою залежить від категорії позичальника і виду кредиту. Це зумовлено значними відмінностями статистичних моделей з огляду на відповідні види кредитування: іпотечне, споживче, на придбання авто тощо [6].

Скорингові системи оцінюють кредитоспроможність позичальника більш ефективно та адекватно, особливо в умовах кризи і посткризової рецесії. Виходячи з досвіду західних країн, після введення в роботу скорингових моделей рівень «поганих» кредитів зменшився на 15-20% у порівнянні з ручним опрацюванням кредитних заявок.

Основними перевагами скорингових систем перш за все є: зниження рівня неповернення кредиту, швидкість і неупередженість в ухваленні рішень, ефективне управління кредитними портфелями, а також відсутність необхідності тривалого навчання персоналу.

Скорингова система аналізу кредитоспроможності фізичних осіб повинна бути статистично вірною і потребує постійного оновлення інформації та вдосконалення моделей [7].

1.2 Історія розвитку скорингу

У загальному значенні скоринг є методом класифікації об’єктів на різні групи. При цьому саме значення характеристики, на основі котрої об’єкти розподіляються на групи, є невідомим, але відомі інші фактори, які пов’язані з цією характеристикою [8].

У статистиці ідея класифікації популяції на групи на прикладі рослин (шкідливі і корисні) була розроблена в 1936 році німецьким біохіміком і лауреатом Нобелівської премії Хансом Фішером (рис. 1.2).

Рисунок 1.2 – Ханс Фішер

У тридцяті роки, під час «Великої депресії», в США були зроблені перші спроби створення суб’єктивних систем кредитного скорингу. Ситуація в економіці була не дуже стабільною, а банкам, з одного боку, хотілося видавати якомога більше позик, а з іншого – мінімізувати частку «поганих» кредитів в своїх кредитних портфелях. Ці системи будувалися на візуальній оцінці потенційних позичальників кредитними менеджерами [9].

Однак, під час Другої світової війни більшість кредитних аналітиків було покликано в діючу армію, а їх місце зайняли нові люди. Перед звільненням керівництво більшості американських банків змусило кредитних

аналітиків написати набір правил, якими слід було керуватися при прийнятті рішення про видачу кредиту, щоб оцінку позичальника міг проводити неспеціаліст. Але незабаром з'ясувалося, що ці правила швидко втрачали актуальність, оскільки не враховували всіх змін в економіці і житті потенційних позичальників. Та й кредитних аналітиків потрібно було занадто багато.

У 1940 році Національне бюро економічних досліджень США опублікувало роботу Ральфа Янга «Personal Finance Companies and Their Credit Practices», де вперше були висловлені ідеї скорингу. Однак, зараз про цю публікацію знає лише вузьке коло фахівців, а «винахідником» скорингу вважається Девід Дюран (рис. 1.3).

Рисунок 1.3 – Девід Дюран

Девід Дюран адаптував розроблену Хансом Фішером методику класифікації рослин для класифікації кредитів на «погані» і «хороші». Національне бюро економічних досліджень США у 1941 р. виступило замовником дослідницького проекту, в якому Дюран на основі статистики комерційних банків і інших фінансових організацій проаналізував сотню позитивних і негативних кредитних історій, використав дискримінантний аналіз та розробив індивідуальну систему «ваг» [8].

Модель Д. Дюрана складається з групи чинників для визначення ступеня кредитного ризику та коефіцієнтів для різних факторів, що характеризують кредитоспроможність клієнта:

- стать: жіноча (0.4 бала), чоловіча (0 балів);
- вік: менше 20 років (0 балів), 21 рік (0.1 бала), 22 роки (0.2 бала), вище 23 років (0.3 бала);
- строк проживання в одному місці: по 0.042 бала за кожен рік, максимальна кількість – 0.42 бала;
- професія: з низьким ризиком (0.55 бала), з високим ризиком (0 балів), інші професії (0.16 бала);
- фінансові показники: якщо наявний банківського рахунку (0.45 бала), якщо наявна нерухомість (0.35 бала), якщо наявний страховий поліс (0.19 бала);
- робота: в суспільній сфері (0.21 бала), інші (0 балів);
- зайнятість: 0.059 бала за кожен рік перебування на останній роботі.

Дюран визначав суму в 1.25 бала як поріг кредитоспроможності (точка відсікання). Банки оперативно почали впроваджувати запропоновану Девідом Дюраном бальну систему оцінки позичальників, тим більше що вона не вимагала від персоналу високої кваліфікації. Потрібно було лише провести опитування клієнта, розставити бали, а потім підсумувати їх.

У 1956 році американці – інженер Біл Файр і математик Ерл Айзек, які працювали в Стенфордському дослідницькому інституті, винайшли першу кредитну скорингову модель. Партнери розробили математичний алгоритм, що обчислює рівень кредитоспроможності позичальника в чисельному значенні. Іншими словами, алгоритм дозволяв прораховувати кредитні ризики у вигляді трьохзначного числа, яке є кредитним рейтингом. Вони організували компанію Fair, Isaac and Company [9].

На початку 60-х років з появою кредитних карток, банки та інші кредитні організації стали розуміти корисність і роль кредитного скорингу. Великий потік клієнтів, які зверталися за кредитними картками, разом з

дефіцитом трудових ресурсів привів до автоматизації кредитного процесу. Використання кредитного скорингу в банках зменшило кількість дефолтів по позикам на 50% в порівнянні з використанням експертної думки (judgmental), яку використовували до цього. Більш детально про це можна прочитати в роботах Майерса і Форга 1963 року, а також Черчілля, Невіна і Ватсона 1977 року [8].

Супротивником даного підходу виступив Капон (1982 р.), який стверджував: «емпіричні висновки кредитного скорингу є груба сила, яка порушує традиції нашого суспільства». Він вказував на те, що має бути більше приділено уваги до кредитної історії позичальника, а також більше розуміння того, чому певні характеристики повинні бути включені в модель, а інші ні.

Закон про рівні кредитні можливості (Equal Credit Opportunity Acts, 1975 – 1976 рр.) вперше законодавчо закріпив застосування кредитного скорингу. Закон забороняв відмовляти у видачі кредиту на підставі наступних показників: раса, колір шкіри, інвалідність, національне походження, вік, стать, сімейний стан, релігія, отримання соціальної допомоги. Ця незаконна дискримінація в наданні кредиту була вирішена за допомогою кредитного скорингу [8].

Успіх застосування скорингу у видачі кредитних карт забезпечив на початку 80-х років перехід методу і на інші банківські продукти, такі як персональні кредити, іпотечні кредити та кредити для малого бізнесу. Досягнення в області обчислювальної техніки дозволили будувати скорингові моделі новими способами. Були впроваджені основні математичні методи, які використовуються і по сьогоднішній день: лінійне програмування, дерева рішень та логістична регресія. Пізніше з’явилися методи штучного інтелекту: експертні системи і нейронні мережі [10].

В даний час акцент робиться з одного боку на мінімізацію ймовірності дефолту позичальника за індивідуальним кредитним продуктом, з іншого – на максимізацію прибутку, який можна отримати від цього позичальника.

1.3 Огляд ринку програмного забезпечення призначеного для банківської клієнтської аналітики

1.3.1 Компанія SAS Institute та система SAS Enterprise Miner

Всесвітньо визнаним лідером в області інтелектуального аналізу даних та спеціалізованих рішень для вирішення задач скорингу є компанія SAS Institute, продукти якої використовуються в 50% банківських установ США, та покривають 30% світового ринку рішень для бізнес-аналітики. В Україні відповідні рішення використовуються в ТОП-30 найбільших банках.

Компанія SAS надає широкий діапазон послуг в сфері бізнес-аналітики: консалтинг, впровадження рішень, навчання персоналу, технічна підтримка. Набір пропонованих компанією рішень і послуг охоплює всі етапи роботи з інформацією – від збору і забезпечення якості даних до процесів їх аналізу та побудови аналітичної звітності [11].

SAS Enterprise Miner (рис. 1.4) – є спеціалізованим інструментом компанії SAS Institute, який коштує близько 100 000, і вирішує такі задачі як задачі прогнозного моделювання, виявлення структур даних та інші задач інтелектуального аналізу даних.

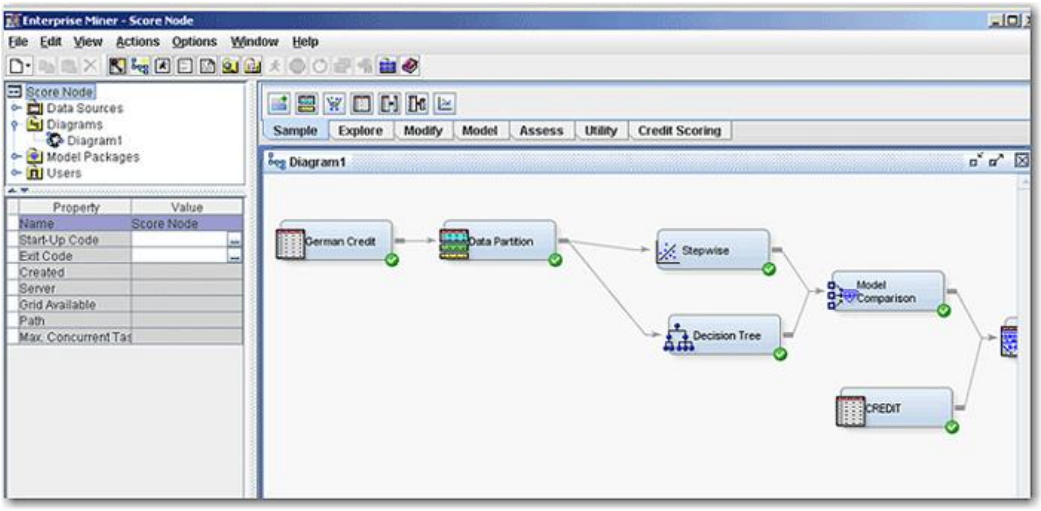


Рисунок 1.4 – Вигляд системи SAS Enterprise Miner

SAS Enterprise Miner використовується для виявлення в масивах даних інформації, яку використовують для прийняття рішень. Оскільки основною спеціалізацією продукту є пошук та аналіз прихованих закономірностей в даних (Data mining) Enterprise Miner складається з:

– методів статистичного аналізу;
– методології виконання проектів дослідження даних (SEMMA)
– графічного інтерфейсу користувача.

SEMMA – абревіатура від Sample (відбір даних), Explorer (дослідження відносин в даних), Modify (модифікація даних), Model (моделювання взаємозалежностей), Assess (оцінка отриманих моделей і результатів), що основними етапами аналітичного проекту.

Уся підготовка й аналіз даних здійснюються за допомогою вузлів (nodes) в Enterprise Miner. Для кожного типу завдань (відповідно до методології SEMMA) існує ряд відповідних вузлів (рис. 1.5) [12].

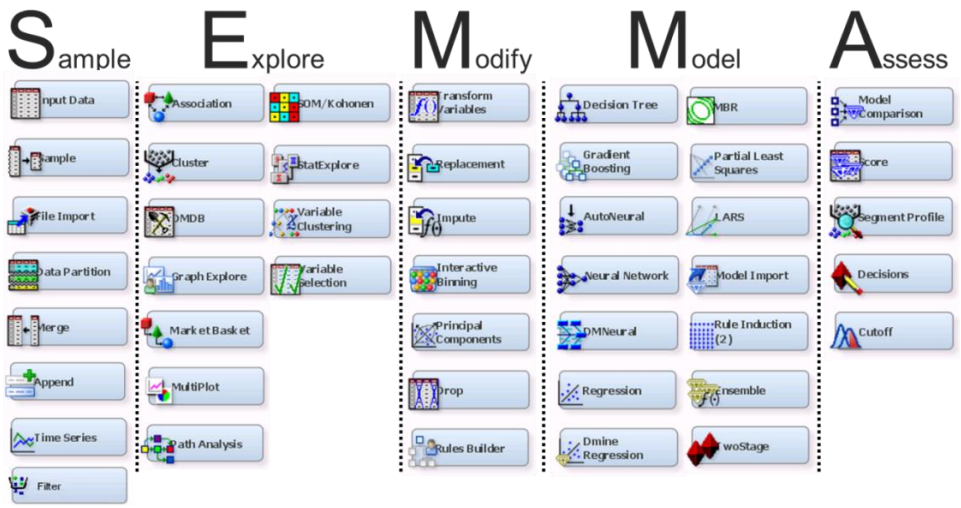


Рисунок 1.5 – Вузли SEMMA

Для вирішення задачі кредитного скорингу, що є певним піднапрямком задач прогнозного моделювання, компанією було розроблено спеціальний набір інструментів (компонентів), що інтегровані у групу під назвою SAS

Credit Scoring [12]. За допомогою цих вузлів розв’язуються задачі розробки і використання скорингових моделей:

- вузол Scorecard розраховує скорингові карти використовуючи результати логістичної регресії на наборі даних навчальної вибірки. Також, даний вузол формує ряд звітів оцінки якості (прогнозуючої здатності) побудованої скорингової карти базуючись на статистичних показниках для визначення балу відсікання;
- вузол Interactive Grouping призначений для автоматичного відбору найбільш значущих вхідних параметрів і формування характеристик для вхідних параметрів з неперервними значеннями. Для автоматичного вибору найбільш значущих вхідних змінних використовуються статистичні показники такі як: коефіцієнти Gini та Information Value, а для автоматичного формування груп використовуються коефіцієнти Weight of Evidence (WOE) в якості критеріїв розбиття діапазону значень на групи;
- вузол Reject Interface доповнює навчальну вибірку даними по позичальникам, яким відмовили, проводячи автоматичну сценарну розмітку прецедентів на позитивні - негативні (розмітка Good/Bad).

1.3.2 Програмне забезпечення IBMSPSS

IBM SPSS (Statistical Package for the Social Sciences – статистичний пакет для соціальних наук) – це програмне забезпечення, призначене для збору статистики і прогностичної аналітики. Воно дозволяє оптимізувати бізнес процеси, приймати обґрунтовані правильні рішення, підвищувати ефективність корпоративної роботи і збільшувати прибуток.

Функції продуктів IBM SPSS:

– аналіз наявної інформації і прогнозування корпоративних бізнес процесів для подальшої оптимізації;
– збір необхідних компанії даних, найбільш доречним способом;
– вплив на показники корпоративної діяльності шляхом застосування прогностичної аналітики IBM SPSS у бізнес процесах.

Склад лінійки продуктів IBM SPSS: IBM SPSS Statistics – платформа для аналізу статистичних даних; IBM SPSS Modeler – рішення для технології Data Mining; IBM SPSS DataCollection – рішення для збору даних і проведення опитувань.

IBM SPSS Statistics (рис. 1.6) – модульне програмне забезпечення, призначене для аналізу даних і охоплює всі рівні аналітичного процесу: від планування аналізу до формування звітів та подання його результатів. Пропонує базові процедури статистики, які необхідні бізнес-керівникам і аналітикам для вирішення важливих питань, пов’язаних з бізнес-процесами і дослідженнями. Ця програма надає засоби, що дозволяють користувачам швидко переглядати дані, формулювати гіпотези для додаткового тестування та виконувати процедури для виявлення зв’язків між змінними, створення кластерів, визначення тенденцій і складання прогнозів.

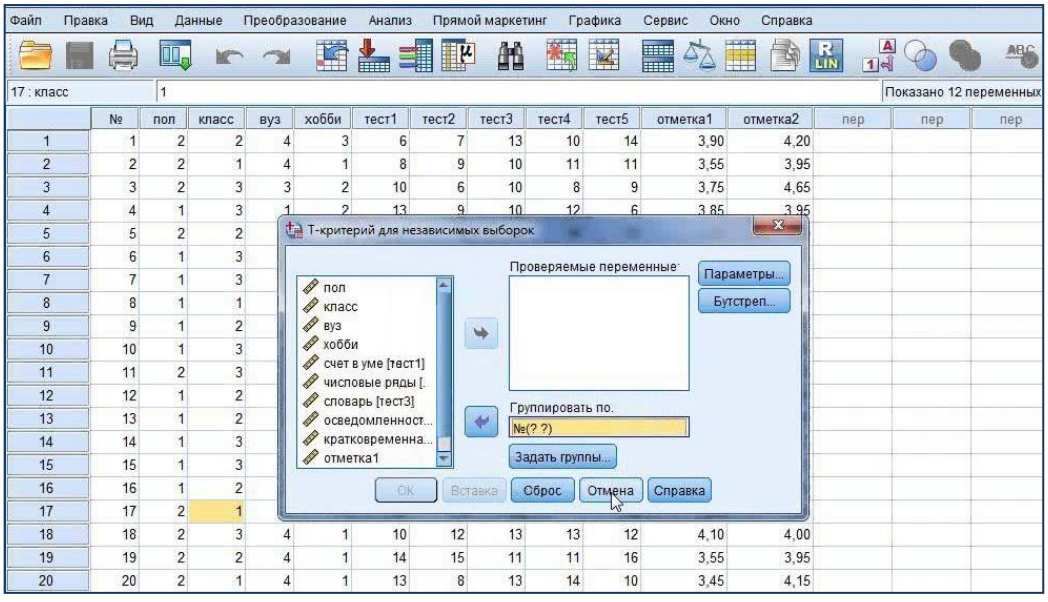


Рисунок 1.6 – Вигляд системи IBM SPSS Statistics

IBM SPSS Statistics широко використовується в таких галузях, як:

– соціальні і маркетингові дослідження;
– управління персоналом та кадрова політика;
– CRM-аналітика – аналіз клієнтських баз даних для оптимізації співпраці в подальшому;
– збір і обробка даних статистики;
– прогнозування;
– освіта та наукова діяльність.

Технологія аналізу інформації Data Mining дозволяє виявляти в архівній інформації корисні і раніше невідомі, нетривіальні знання, необхідні для прийняття оптимальних рішень в бізнес процесах. Ключовою метою Data Mining є знаходження закономірностей в даних, вивчення складних систем і їх моделювання, яке ґрунтується на історії їх поведінки.

IBM SPSS Modeler (рис. 1.7) є програмним продуктом для Data Mining, який володіє всіма необхідними інструментами для аналітичної роботи з даними, розробки та реалізації ефективних прогностичних моделей, як фахівцями, так і звичайними користувачами.

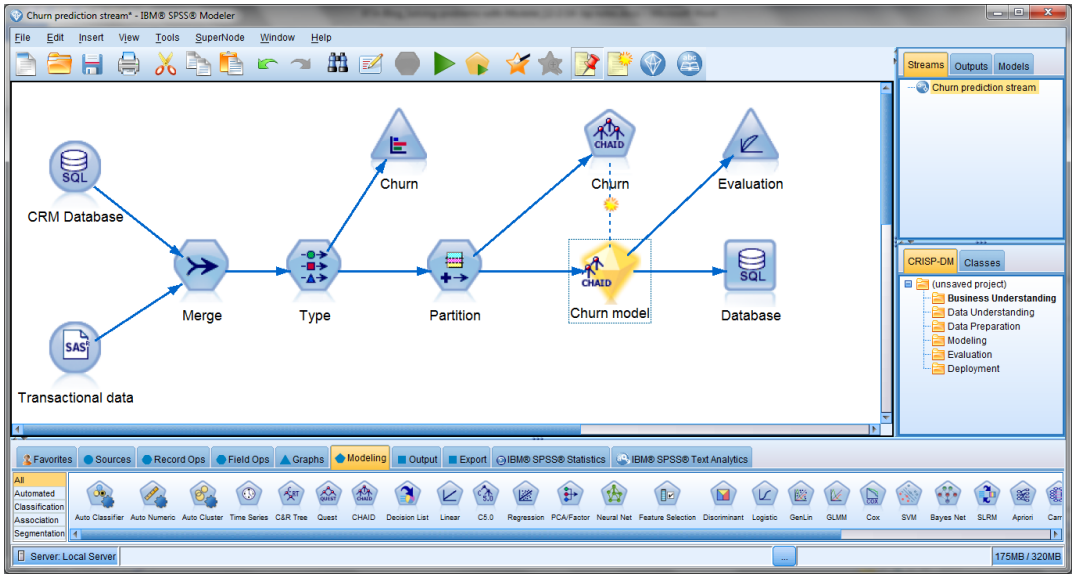


Рисунок 1.7 – Вигляд системи IBM SPSS Modeler

Використання візуального інтерфейсу у поєднанні з сучасними аналітичними засобами дозволяє виявляти залежності і тенденції в структурованих або неструктурованих даних. Платформа надає набір удосконалених алгоритмів і технологій, у тому числі аналіз тексту, аналіз сутностей, управління рішеннями і їх оптимізацію.

1.3.3 Аналітична платформа Deductor Studio

Deductor Studio – аналітична платформа компанії Base Group Labs, що дає змогу аналітику автоматизувати шаблонні операції по обробці даних і зосередитися на інтелектуальній роботі: формалізація логіки прийняття рішень, побудова моделей, прогнозування. Інші співробітники компанії можуть легко скористатися готовими результатами, не вникаючи в складності аналізу (рис. 1.8).

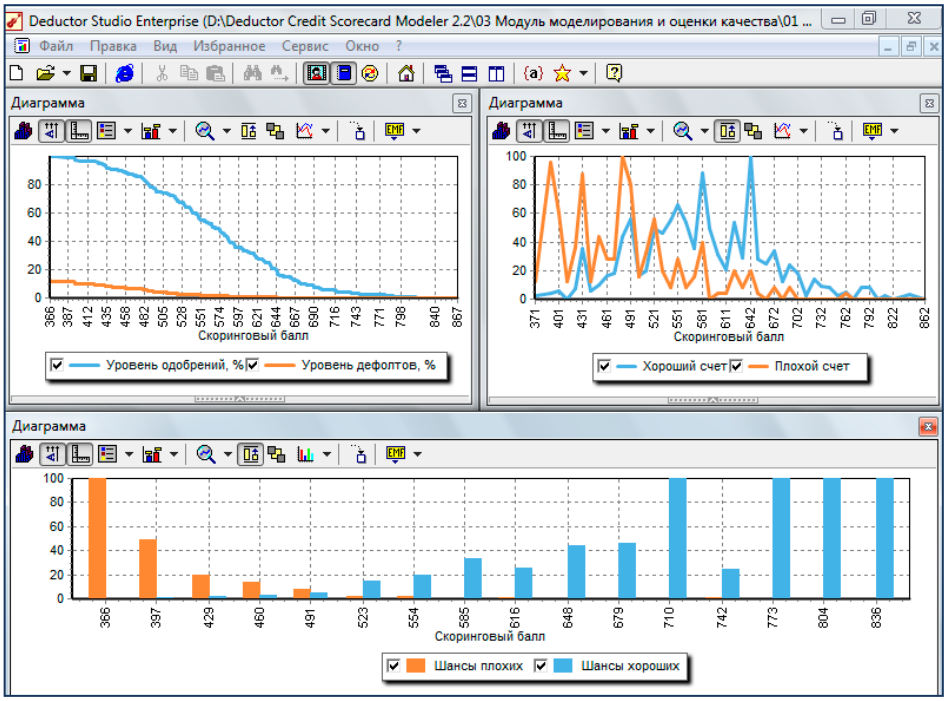


Рисунок 1.8 – Вигляд системи Deductor Studio

Deductor Credit Scorecard Modeler – комплексне рішення, що автоматизує процес побудови скорингових карт. Застосування даного рішення дозволяє на підставі великої кількості характеристик позичальника кількісно оцінити пов’язані з клієнтом ризики і передбачати ймовірність повернення ним кредиту.

Deductor Credit Scorecard Modeler дозволяє виконати всі етапи побудови скорингової карти в рамках однієї платформи і рішення:

- імпорт даних з різнорідних джерел з подальшим завантаженням в спеціалізовану вітрину даних;
- побудова матриці міграції;
- визначення статусу рахунку;
- формування вибірки для побудови карти, семплінг;
- балансування класів;
- аналіз прогнозуючої сили атрибутів і вибір значущих атрибутів;
- побудова скорингової моделі, калібрування і побудова скорингової карти;
- аналіз відхилених заявок;
- оцінка якості скорингової карти;
- вибір порогового балу;
- тестування і моніторинг карти.

Використання даного програмного продукту надає такі переваги:

- скорочення кількості необґрунтованих відмов;
- підвищення точності оцінки позичальника;
- збільшення швидкості оцінки позичальника;
- мінімізація людського фактору в процесі прийняття рішення;
- отримання готових звітів на всіх етапах розробки скорингової карти.

1.4 Постановка задачі дослідження

Метою магістерської дисертації є дослідження та вдосконалення існуючих методів аналізу і прогнозування кредитних ризиків, розробка програмного забезпечення для попередньої обробки даних і побудови скорингової моделі, та перевірка побудованої моделі на адекватність.

У рамках дисертації необхідно:

- розробити архітектуру системи підтримки прийняття рішень для аналізу, моделювання та прогнозування ймовірності повернення наданого споживчого кредиту;
- розробити програму для побудови скорингової моделі на основі алгоритму логістичної регресії з використанням методу максимальної правдоподібності;
- протестувати комп’ютерну програму на реальних даних та провести порівняльний аналіз з іншими методами.

Для рішення цих задач необхідно дослідити вже існуючі інтелектуальні рішення для вирішення задач керування кредитними ризиками.

Об’єкт дослідження – статистичні дані щодо виданих фінансовими установами споживчих кредитів, які потребують ефективної аналітичної обробки та є необхідними для побудови скорингових моделей та прийняття рішень при визначені кредитоспроможності клієнтів банку.

Предмет дослідження – математичні методи побудови скорингових моделей, а саме: лінійна регресія, логістична регресія, дерева рішень.

Висновки до розділу 1

Кредитна діяльність банківських установ є однією з основних ознак, що вирізняє банки серед інших установ. Кредитні операції, як правило, є найбільш прибутковою часткою банківського бізнесу. Проте, неповернення кредитів може привести до банкрутства банківську установу. Ось чому кредитні ризики являють собою серйозну проблему для банку, і управління ними є критично необхідним.

У даному розділі розглянуто основний інструмент з мінімізації ризику в кредитній діяльності банку, а саме: оцінку кредитоспроможності позичальників. Було дано визначення поняттям кредитоспроможності і кредитного скорингу, а також досліджено основні етапи розвитку скорингу. Розглянуто чотири види кредитного скорингу: application-scoring (аплікаційний скоринг), behavioral-scoring (поведінковий скоринг), collection-scoring (колекторський скоринг) та fraud-scoring (скоринг проти шахраїв).

Проведено огляд програмних продуктів компаній, які є основними лідерами на ринку програмного забезпечення призначеного для банківської клієнтської аналітики, а саме SAS Enterprise Miner, IBM SPSS Statistics, IBM SPSS Modeler, Deductor Studio.

Показано актуальність та перспективність дослідження, на основі чого сформульовано постановку задачі магістерської дисертації, та виділено етапи її розв’язку.

РОЗДІЛ 2 МЕТОДИКА ПРОГНОЗУВАННЯ

Скорингові моделі традиційно використовуються в банківській сфері для оцінки кредитоспроможності позичальників на стадії розгляду заявок на кредит. Скоринг дозволяє отримати математико-статистичну модель класифікацій спостережень за різними групами відповідно до характеристик цих спостережень.

Для побудови скорингових систем можуть використовуватися різні математико-статистичні моделі. Вибір конкретного методу залежить від передумов його застосування і від шкал вимірювання наявних статистичних даних. У зв’язку з цим постає задача вибору оптимальної моделі, яка б давала адекватні прогнози для процесів чи системи, які досліджуються.

На практиці найбільш часто використовують такі методи: лінійний регресійний аналіз, логістичну регресію, дискримінантний аналіз, дерева рішень та нейронні мережі.

Окрім того, дуже важливим є етап попередньої підготовки і дослідження даних, який може займати до 90% витраченого часу на створення скорингової моделі. При побудові скорингових моделей для оцінювання кредитних ризиків найчастіше виникають такі особливості реальних статистичних даних: різний формат зібраних даних, наявність пропущених або некоректних значень в статистичних вибірках, наявність аномальних спостережень.

У даному розділі розглядаються підходи та методи побудови скорингових систем, етапи процесу розробки скорингової моделі та способи подолання вище зазначених проблем аналізу реальних статистичних даних.

2.1 Методи і моделі для вирішення задачі оцінювання кредитних ризиків

2.1.1 Дерева рішень

Популярним підходом, щодо вирішення завдання оцінки кредитоспроможності фізичних осіб є застосування алгоритмів, що вирішують задачу класифікації, а саме віднесення будь-якого об’єкта (потенційного позичальника) до одного із заздалегідь відомих класів («хороший»/«поганий»).

Такого роду завдання можуть вирішуватися за допомогою дерев рішень. Метод дерев рішень дозволяє автоматично аналізувати величезні масиви даних. Перші згадки про дерева рішень можна знайти в роботах Е.Ханта і П.Ховленда, що датуються 50-ми роками XX століття.

Отримана в результаті використання цього методу модель – це зручний засіб представлення правил у вигляді ієрархічної, послідовної структури, де кожний об’єкт потрапляє в єдиний для нього кінцевий вузол. Під правилом розуміється логічна конструкція, представлена у формі «якщо..., то...».

Менеджери найчастіше дають перевагу даному підходу за його наочність та інтуїтивну зрозумілість процесу прийняття рішення у вигляді простих та зрозумілих правил (рис. 2.1).

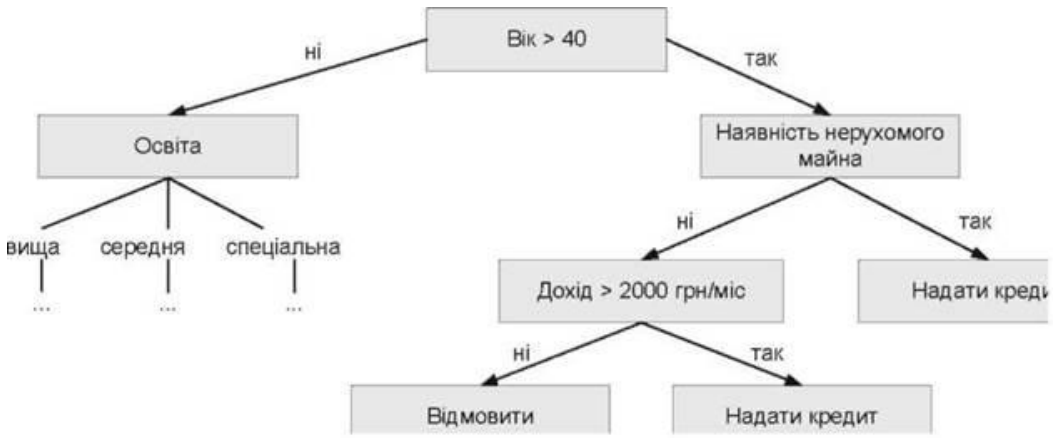


Рисунок 2.1 – Приклад моделі у вигляді дерева рішень

Розглянемо сутність даного підходу. Дерево будується на основі даних з минулих періодів, завдяки чому заздалегідь відомо до якого класу відноситься кожна з ситуацій. Тобто має бути відомо, чи був погашений кредит.

Спочатку всі ситуації з навчальної вибірки потрапляють в верхній (перший) вузол, а потім вони розподіляються по нижніх вузлах, що теж можуть бути розподілені на дочірні вузли. Критерієм розбиття є певні значення одного з вхідних параметрів. Для визначення поля, по якому буде відбуватися розбиття, використовується ентропія (міра невизначеності). Обирається те поле, при розбитті по якому вдасться найбільше позбавитися невизначеності. Чим більше об’єктів, що відносяться до різних класів (домішок) знаходиться в одному вузлі, тим більшою є невизначеність. Якщо у вузлі знаходяться об’єкти, що відносяться до одного класу, то ентропія повинна дорівнювати нулю.

Після побудови моделі на навчальній вибірці, отриману модель можна використовувати для визначення класу («хороший»/«поганий») нових ситуацій, тобто коли потенційний клієнт хоче отримати кредит.

Якщо на ринку істотно змінюється ситуація, то модель дерева можна перебудувати, адаптувавши до існуючої обстановки.

На сьогоднішній день відомо багато алгоритмів, що дозволяють побудувати дерева рішень: QUEST, CART, CHAID, C4.5, CN2, C&R тощо.

Метод CHAID (Chi-Square Automatic Interaction Detection – автоматичний детектор взаємозв'язків на основі критерію χ^2) розроблений в 1980 р. Є найбільш вживаним і швидкодіючим багатовимірним статистичним методом побудови дерева рішень, що базується на використанні критерію зв’язку χ^2 для пошуку оптимального розбиття між категоріальними змінними. При необхідності кожна вершина дерева може бути поділена більше ніж на дві вершини наступного рівня.

У випадку інтервальної залежної змінної в якості критерія оптимізації використовується F-критерій Фішера. Якщо незалежні змінні є

інтервальними, то вони автоматично перетворюються в категоріальні, окрім того кількість категорій можна змінювати.

В 1991р. був розроблений метод Вичерпний (Exhaustive) CHAID, що є модифікацією методу CHAID. Перевага даного методу полягає в тому, що при побудові дерева відбувається попередній аналіз всіх можливих розбиттів на наступних етапах алгоритму. Звичайно таки аналіз потребує більше ресурсів та часу.

Зовсім інший підхід було використано при розробці методу CART (Classification And Regression Trees – дерева класифікації і регресії) в 1984 р. На відміну від двох попередніх методів, метод CART базується не на статистичних критеріях відмінностей, а на мінімізації неоднорідності в групах об'єктів кінцевих вершин дерева. При цьому поділ «батьківської» вершини дерева відбувається лише на дві «дочірні» вершини наступного рівня.

Як і в вже розглянутих методах, в методі CART використовується як кількісні, так і категоріальні цільові і незалежні змінні. Оскільки в даному методі відбувається повний перебір, метод завжди знаходить оптимальний варіант обраного критерію. Але це рішення має невелику специфіку: оптимальним результатом найчастіше виявляється використання багатокатегоріальних незалежних змінних. Крім того, при великій кількості багатокатегоріальних незалежних змінних, алгоритм може досить довго опрацьовувати кожну таку змінну.

Метод QUEST (Quick, Unbiased, Efficient Statistical Trees – швидкі, незсунені, результативні статистичні дерева) був розроблений для усунення недоліків попереднього методу. Цей статистичний метод призначений для швидкої і ефективної побудови бінарних (як і в випадку CART) дерев був розроблений в 1997 р. Однак, цей метод застосовується, тільки в тому випадку якщо цільова змінна є номінальною. В залежності від типу тої чи іншої незалежної змінною використовуються різні статистичні критерії, а тому незалежні змінні можуть бути будь-якими.

2.1.2 Лінійна імовірнісна модель

Лінійна імовірнісна модель (ЛІМ) – це модель у формі лінійної регресії, залежна змінна якої набуває значення 0 або 1 залежно від того, яким є результат повернення особою наданого споживчого кредиту [13]. Формально така модель має вигляд:

$$y(k) = b_0 + b_1 x_1(k) + b_2 x_2(k) + \ldots + b_m x_m(k) + \varepsilon(k), \qquad (2.1)$$

де y – залежна змінна, значення якої відповідає результату повернення кредиту;

$b_0,\ldots,b_m$ – коефіцієнти (параметри) рівняння регресії, які оцінюються за даними, які характеризують клієнтів;

$x_1,\ldots,x_m$ – пояснювальні змінні (характеристики клієнта);

$\varepsilon(k)$– випадковий процес, зумовлений наявністю не вимірюваних збурень, а також помилок оцінювання структури і параметрів моделі;

k – ідентифікатор клієнта.

Застосування ЛІМ пов’язане з такими недоліками:

а) залежна змінна може набувати значення, які перебувають поза інтервалом [0, 1];

б) ЛІМ працюють виключно із змінними, які приймають неперервні значення;

в) процеси кредитування частіше характеризуються нелінійними залежностями, що потребує застосування моделей інших структур.

Можливі способи вирішення зазначених проблем:

а) трансформація результуючих значень залежної змінної, за допомогою обраного порогу відсікання (наприклад, 0.5):

$$\begin{cases} y = 0, \text{якщо } y < 0.5; \\ y = 1, \text{якщо } y \geq 0.5. \end{cases} \quad (2.2)$$

б) Кодування категоріальних змінних за допомогою:

1) порядкової нумерації категорій;

2) значень коефіцієнту зваженої сукупності (WOE);

в) Використання замість ЛІМ логіт- та пробіт-моделі.

2.1.3 Логістична регресія

Логістична регресія – є корисним класичним інструментом для вирішення завдань регресії та класифікації. В останні роки логістична регресія набула поширення в скорингу для розрахунку рейтингу позичальників і управління кредитними ризиками.

Логістична регресія є різновидом множинної регресії. Призначення логістичної регресії – аналіз зв’язку між цільовою змінної та незалежними змінними, або, як їх ще називають, регресорами чи предикторами. Якщо цільова змінна приймає тільки два значення (наприклад, 0 або 1), тобто є бінарною, то в такому випадку застосовується бінарна логістична регресія. Логістична регресія дозволяє оцінити ймовірність того, що відбудеться якась конкретна подія для певного випробування (наприклад, повернення кредиту).

Усі регресійні моделі можна записати у такому вигляді:

$$y = F(x_1, \dots, x_n). \quad (2.3)$$

Множинна регресія передбачає, що залежна (цільова) змінна є лінійною функцією від незалежних змінних (предикторів), тобто:

$$y = b_0 + b_1 x_1 + \cdots + b_n x_n. \quad (2.4)$$

Якщо розглядати модель множинної регресії для розв’язання задачі оцінювання ймовірності повернення кредиту, то необхідно задати змінну y зі значеннями 0 та 1, де 1 означає, що потенційний клієнт погасив кредит, а 0, що клієнт вчасно не розрахувався по кредиту. Однак тут з'являється проблема: множинна регресія не знає, що змінна відгуку бінарна за своєю природою. Це призведе до того, що модель буде прогнозувати значення більші за 1 і менші за 0. Проте, дані значення не є припустимими для поставленого завдання. Тому, множинна регресія буде ігнорувати обмеження на можливі значення для y.

Для того, щоб вирішити цю проблему, завдання регресії можна сформулювати по іншому: замість передбачення бінарної змінної, ми прогнозуємо неперервну змінну, яка може приймати значення з відрізку [0,1] при будь-яких значеннях предикторів. Цього можна досягти за допомогою застосування регресійного рівняння (логіт-перетворення):

$$p = \frac{1}{1 + e^{-y}}, \quad (2.5)$$

де p – ймовірність того, що відбудеться подія, що нас цікавить;

y – регресійне рівняння.

На рисунку 2.2 показана залежність між ймовірністю настання події і величиною y.

Логістична регресія фактично служить не для передбачення значень цільової змінної, а скоріше для оцінки ймовірності того, що залежна змінна прийме задане значення.

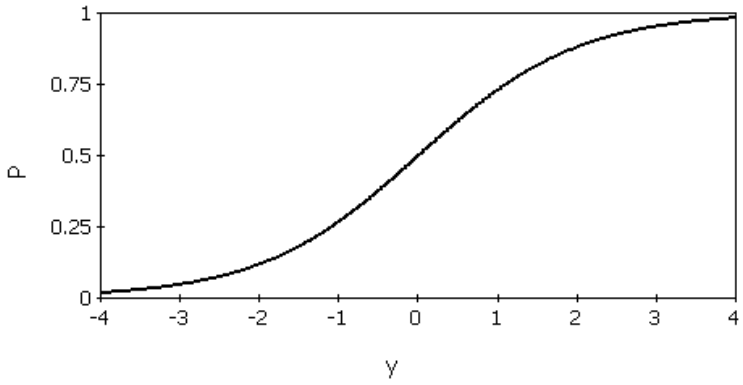


Рисунок 2.2 – Логістична крива

Існує кілька медотів знаходження коефіцієнтів логістичної регресії. Найчастіше використовують метод максимальної правдоподібності. Його використовують в статистиці, щоб одержати оцінки параметрів генеральної сукупності за даними вибірки. В основі методу знаходиться функція правдоподібності (likehood function), яка виражає щільність імовірності спільної появи результатів вибірки. Альтернативним методом оцінювання є метод Монте-Карло для марковських ланцюгів (МКМЛ), який ґрунтується на генеруванні псевдовипадкових послідовностей (ПВП) і відборі випадкових значень, які задовольняють певні вимоги. Цей метод широко використовують для оцінювання нелінійних моделей завдяки наявності альтернативних методів генерування ПВП.

2.1.3.1 Знаходження параметрів логістичної регресії з використанням методу максимальної правдоподібності

Метод максимальної правдоподібності дозволяє оцінити невідомі параметри за допомогою максимізації функції правдоподібності. Відповідно до цього методу вибираються такі параметри $b_0,...,b_m$ рівняння регресії (2.4), що значення функції правдоподібності є максимальним на навчальній вибірці [14].

$$\hat{b} = \arg\max_{b} L(b), \tag{2.6}$$

де $L(b)$ – функція правдоподібності;

b – вектор параметрів рівняння регресії;

$\hat{b}$– вектор оцінок параметрів рівняння регресії.

Випадкова величина y має розподіл Бернулі, оскільки приймає тільки два значення (0 і 1). Тоді, ймовірність настання події $y = 1$ рівна:

$$\Pr\{y = 1 | x\} = f(z), \tag{2.7}$$

$$f(z) = \frac{1}{1 + e^{-z}}, \tag{2.8}$$

$$z = b_0 + b_1 x_1 + \cdots + b_n x_n, \tag{2.9}$$

де $f(z)$ – логістична функція;

x_i – незалежні змінні;

b_i – параметри рівняння регресії.

Ймовірність настання другої можливої події $y = 0$ рівна:

$$\Pr\{y = 0 | x\} = 1 - f(z). \tag{2.10}$$

Отже, функцію розподілу y при заданому x можна записати у такому вигляді:

$$\Pr\{y | x\} = f(b^T x)^y (1 - f(b^T x))^{1-y}, y \in \{0, 1\}. \tag{2.11}$$

Тоді, функція правдоподібності на навчальній вибірці має вигляд:

$$L(b) = \prod_{i=1}^{n} \Pr\{y = y^{(i)} \big| x = x^{(i)}\}. \tag{2.12}$$

Замість максимізації функції правдоподібності можна максимізувати її логарифм:

$$\ln L(b) = \sum_{i=1}^{n} \ln \Pr\{y = y^{(i)} \big| x = x^{(i)}\} = \tag{2.13}$$

$$= \sum_{i=1}^{n} (y^{(i)} \ln f\left(b^T x^{(i)}\right) + \left(1 - y^{(i)}\right) \ln(1 - f\left(b^T x^{(i)}\right))).$$

Для того, щоб максимізувати цю функцію можна застосувати, наприклад, метод градієнтного спуску [15]. Даний метод полягає у виконання наступних ітерацій, починаючи з деякого початкового значення параметрів регресії b:

$$b \coloneqq b + \alpha * \nabla \ln L(b) = b + \alpha * \frac{d \ln L(b)}{db} = \tag{2.14}$$

$$= b + \alpha * \sum_{i=1}^{n} \left(y^{(i)} - f\left(b^T x^{(i)}\right)\right) * x^{(i)},$$

де $\alpha > 0$ – крок методу.

Також, на практиці використовують стохастичний градієнтний спуск та метод Ньютона.

2.1.4 Скорингова карта

Скорингова карта – це набір характеристик (вік, дохід, професія, стаж роботи, наявність майна і т.д.) позичальника і певних вагових коефіцієнтів, виражених в балах (табл. 2.1). Клієнту банку нараховується певна кількість балів в залежності від тих даних, що він повідомив про себе. Максимальна сума кредиту, яку банк готовий надати позичальнику, розраховується залежно від кількості набраних скоринг-балів.

Таблиця 2.1 – Приклад скорингової карти

Показник	Значення	Бали
Вік	20-30	100
	30-35	107
	35-40	123
Кількість дітей	Немає	100
	Один	90
	Два	80
	Три	70
	Більше трьох	30
Дохід	1000-4000	120
	4001-6000	140
	6001-8000	160

Логістична регресія є найбільш поширеною статистичною моделлю побудови скорингової карти при бінарній залежній змінній.

Оцінювання коефіцієнтів логістичної регресії як скорингових балів є основним етапом розробки скорингової карти. Підсумковий скоринговий бал (total score) в шкалі натуральних логарифмів розраховується як сума оцінок

коефіцієнтів логістичної регресії перемножених на значення незалежних змінних:

$$total\ \ score = \beta_1' x_1 + \ldots + \beta_k' x_k, \tag{2.15}$$

де β_j' – оцінки коефіцієнтів логістичної регресії;

x_i – значення регресорів для i-го позичальника.

Для трансформації скорингових балів до лінійної шкали використовується техніка масштабування. Масштабування не змінює прогнозних властивостей скорингової карти, а лише трансформує скорингові бали в нову шкалу, зручну для використання. Скоринговий бал в лінійній шкалі є відношенням шансів «хороших» позичальників до «поганих» [16].

Для масштабування необхідно перш за все задати мінімум і максимум числової шкали (наприклад, 0 та 1000). В процесі масштабування важливу роль грають такі показники, як кількість балів, що подвоює шанси стати «хорошим» клієнтом, та значення шкали, в якому досягається задане відношення шансів «поганих» до «хороших». Найчастіше використовують скорингові карти, де кожні 20 балів подвоюють шанси стати «хорошим». В іншому варіанті – кожні 40 балів подвоюють шанси стати «хорошим» позичальником [16]. Для приведення коефіцієнтів логістичної регресії в скорингові бали в лінійній шкалі застосовують наступне перетворення:

$$бал = A + R * b_j', \tag{2.16}$$

де A – зміщення;

R – множник.

Множник визначається по формулі:

$$R = \frac{D}{\ln(2)}, \tag{2.17}$$

де D – бали, що подвоюють шанси.

Зміщення розраховується по формулі:

$$A = B - R * \ln(C), \tag{2.18}$$

де B – значення на шкалі балів, в якій відношення шансів складає C:1.

2.2 Попередній аналіз і обробка даних для побудови скорингової моделі

2.2.1 Збір даних

Першим і одним з основних етапів побудови скорингової моделі є збір достатньої кількості репрезентативної вибірки даних кредитної історії позичальників банку, тобто наявної інформація про виконання або невиконання своїх зобов'язань по кредитам. Точність прогнозу та успіх розробленої скорингової системи в цілому залежить від якості вихідних даних. Для побудови скорингових моделей необхідно використовувати надійні і очищені дані з мінімально допустимою кількістю «поганих» і «хороших» записів. Кількість необхідних даних визначається за допомогою вимог статистичної значущості і випадковості, але в принципі може бути різним. Для вирішення практичних задач розробки скорингових моделей, експерти в сфері банківського скорингу рекомендують використовувати не менше 2000 «поганих» та 2000 «хороших» записів про клієнтів, які вибираються випадковим чином з загальної історії клієнтів відповідного банку чи бюро кредитних історій. Крім цього, в спеціальних методах

скорингу можуть додатково знадобитися 2000 відхилених заявок, що дозволить проаналізувати причини відхилення. У вихідних даних для побудови скорингових моделей можуть бути внутрішні дані з анкет позичальників банку та зовнішні дані кредитних історій [16].

Необхідно виключити з вихідних даних інформацію про певний тип клієнтів. Це можуть бути нетипові позичальники, такі як VIP клієнти, співробітники банку, шахраї, кредити за вкраденими картками, чи померлі, неповнолітні, тощо. Також треба виключити з бази кредити з дуже великими сумами кредитів, нетиповими цілями позики, нестандартними умовами погашення. Одним з критеріїв відбору вихідних даних може бути тип кредитування, для якого необхідно розробити скорингову модель [17].

2.2.2 Визначення та обробка пропусків

Зазвичай історичні дані характеризуються відсутністю деяких необхідних значень або, навпаки, присутністю значень, що є некоректними та не можуть описувати ту чи іншу характеристику. Це можуть бути поля, значення яких більше не використовуються, або не були зафіксовані, або ж які були недоступні чи не були заповнені позичальниками тобто пропущенні значення; а також неправильно введені дані, викиди або значення, що дуже виділяються, тобто помилкові, некоректні дані. Є кілька методів для обробки даних з такими значеннями, наприклад:

а) виключити з аналізу всі дані з пропущеними значеннями, оскільки аналіз ведеться по всім змінним. У випадку роботи з реальними фінансовими даними такий спосіб в більшості випадків вилучає занадто багато даних;

б) виключити з моделі характеристики чи записи, в яких доля пропущених значень є більше певного порога (наприклад, більше 20%);
в) включити до аналізу нову характеристику (ідентифікатор), що відображає наявність пропуску по атрибуту клієнта;
г) замінити пропущені значення, базуючись на середньому значенні, або прогнозуванні (наприклад дерева рішень чи регресійні методи), або статистичних спеціалізованих методів (синтетичний розподіл).

2.2.3 Визначення цільової та незалежних змінних моделі

Від мети побудови скорингової моделі залежить вибір цільової (залежної) змінної. Цілі можуть бути різними: загальні – зменшення втрат за новими виданими кредитами, конкретними – скорочення непогашених кредитів по виданим кредитам протягом 4-х місяців після прийняття рішення щодо видачі. Залежна змінна може приймати кількісні та якісні значення. Найчастіше залежна змінна має категоріальний тип вимірювання і приймає дві категорії: «хороший» та «поганий» клієнт. До категорії «поганий» зазвичай відносять клієнтів, які мають прострочену заборгованість 90 днів і більше [17].

Незалежними змінними при побудові кредитної скорингової моделі виступають дані з кредитної заявки такі як: соціально-демографічні дані клієнта (вік, стать, сімейний стан, наявність дітей, посада, дохід тощо), інформація про позику (термін погашення кредиту, сума кредиту, розмір першого внеску, мета кредиту та інше). Також, на момент подачі заявки клієнтом, використовують дані з Бюро кредитних історій як основне джерело даних для формування незалежних змінних. Виділяють основні характеристики, які бюро може надати: рейтинг клієнта, детальна інформація

про наявність кредитів в інших банках, інформація про прострочені чи повністю погашені кредити в минулому, наявність інших банківських послуг і продуктів у клієнта. Також, для формування незалежних змінних, використовується внутрішня банківська кредитна історія позичальника: поточний баланс рахунку, заборгованість на даний момент, кількість рахунків, число попередніх кредитів у банку, найбільша сума заборгованості за попередніми кредитними рахунками [17].

Отже, незалежні вхідні змінні є досить різноманітними і можуть бути представлені в різних одиницях виміру в залежності від можливості об’єктивних вимірів відібраних характеристик. Для рішення практичних задач скорингові моделі можуть бути побудовані з наступними видами незалежних змінних: тільки з категоріальними, тільки з кількісними, одночасно з кількісними і категоріальними змінними. Найбільш використовуваними змінними при побудові скорингових моделей є категоріальні змінні. Основними перевагами категоризації кількісних змінних при побудові скорингових моделей є: полегшення обробки викидів та екстремальних значень кількісних змінних, відображення складних нелінійних зв’язків.

2.2.4 Відбір найбільш значущих змінних

До остаточної моделі необхідно включати лише найбільш значущі незалежні змінні, які при побудові скорингової моделі будуть мати найбільш прогностичні характеристики.

Для оцінки прогнозної сили атрибутів використовують показник зваженої сукупності – WOE (Weight of Evidence). WOE вимірює статистичну значущість кожного класу змінної і розраховується як:

$$WOE = \ln(\frac{d_i^{(1)}}{d_i^{(2)}}), \tag{2.19}$$

де $d_i^{(1)}$ відносна частка «хороших» кредитів в i-й категорії;

$d_i^{(2)}$ – відносна частка «поганих» кредитів в i-й категорії.

При розрахунку прогнозуючої сили характеристики в цілому значення WOE для кожного атрибута агрегуються в показник інформаційного значення, або індекс IV (Information Value).

Показник інформаційного значення прийнято використовувати в кредитному скорингу для оцінювання ступеня взаємозв'язку між залежною змінною і незалежними, IV обчислюється за формулою:

$$IV = \sum_{i=1}^{k}(d_i^{(1)} - d_i^{(2)}) * WOE_i, \tag{2.20}$$

де $d_i^{(1)}$ відносна частка «хороших» кредитів в i-й категорії;

$d_i^{(2)}$ – відносна частка «поганих» кредитів в i-й категорії;

WOE – значення зваженої сукупності i-ї категорії;

k – кількість категорій незалежної змінної.

Корисність змінної при побудові скорингової моделі визначається її інформаційним значення, чим воно вище тим більш змінна є корисною. Також при відборі змінних для побудови скорингової моделі можна використовувати наступні правила (табл. 2.2).

Таблиця 2.2– Оцінка значущості незалежної змінної за значенням IV

Значення IV	Прогнозна здатність
менше 0.02	Не має
від 0.02 до 0.1	Низька
від 0.1 до 0.3	Середня
від 0.3 до 0.5	Добра
більше 0.5	Дуже добра

2.2.5 Використання категоріальних змінних в регресійних моделях

Регресійні моделі, такі як лінійна та логістична регресії, дозволяють працювати лише з числовими змінними. Однак на практиці часто зустрічаються задачі з категоріальними змінними, які несуть в собі багато інформації. Для забезпечення можливості роботи з такими змінними необхідне коректне перетворення текстових значень у числовий формат.

Задавалося б, можна просто звести задачу з категоріальними змінними до задачі з числовими просто пронумерувавши значення змінних (табл. 2.3).

Таблиця 2.3 – Приклад перетворення змінної "Ціль кредиту"

Назва категорії	Номер категорії
Автомобіль	1
Відпочинок	2
Меблі	3
Побутова техніка	4
Покупки в магазинах	5

Однак такий підхід зазвичай закінчується невдачею. Адже вихідна множина значень категоріальних змінних є невпорядкованою, і прогнозуюча

модель буде враховувати той порядок який буде введено. Однак існую багато варіантів нумерацій з різним взаємним порядком, і не є очевидним, який саме порядок обрати.

Найвідомішим методом перетворення категоріальних змінних у числові є метод фіктивних змінних. Метод полягає у тому, що кожна категоріальна змінна розбивається на множину фіктивних бінарних змінних (табл. 2.4). При цьому, зазвичай, одна з категорій не кодується для того, щоб забезпечити функціональну незалежність множини створених змінних. Після чого категоріальна змінна змінюється на набір k-1 бінарних змінних, де k – кількість взаємовиключних категорій даної категоріальної змінної.

Таблиця 2.4 – Приклад перетворення змінної "Ціль кредиту"

Назва категорії	A1	A2	A3	A4
Автомобіль	0	0	0	0
Відпочинок	1	0	0	0
Меблі	0	1	0	0
Побутова техніка	0	0	1	0
Покупки в магазинах	0	0	0	1

Недоліками даного методу є те, що метод не враховує ні характеристики розподілу категорій, ні будь-який взаємозв’язок з цільовою змінною. Ще одним недоліком такого підходу є значне збільшення кількості змінних при перетворенні кожної категоріальної змінної у множину фіктивних змінних. Більшість з алгоритмів не зможуть обробити отриману кількість даних на реальних задачах.

В якості альтернативи методу фіктивних змінних можна використати значення зваженої сукупності WOE (Weight of evidence), який показує міру інформативності відповідного класу змінної та обчислюється за формулою:

$$WOE = \ln(\frac{d_i^{(1)}}{d_i^{(2)}}), \quad (2.21)$$

де $d_i^{(1)}$ і $d_i^{(2)}$ – відносні частки «хороших» і «поганих» кредитів в i-й категорії.

Метод зваженої сукупності дозволяє ставити числові значення ваг категорій змінної у відповідність категоріальним (табл. 2.5).

Таблиця 2.5 – Приклад перетворення змінної "Ціль кредиту"

Назва категорії	Значення WOE
Автомобіль	-0.21
Відпочинок	0
Меблі	0.19
Побутова техніка	-0.11
Покупки в магазинах	0.05

2.2.6 Формування навчальної та тестової вибірки

Одним з важливих етапів побудови скорингової моделі є апробація її на реальних даних та перевірка достовірності. Про ступінь валідації моделі каже здатність її правильно класифікувати об’єкти, здатність моделі відрізняти «хороших» позичальників від «поганих». Модель повинна коректно прогнозувати не тільки на навчальній вибірці, але й на практиці при її застосуванні [17]. Найпоширеніша стратегія перевірки моделі на адекватність– формування випадковим чином двох вибірок: навчальної – по ній будується модель, і тестової – призначена для перевірки моделі. Перевірка адекватності моделі, зазвичай, відбувається з використанням

навчальної і тестової вибірок в пропорціях близько 75-85% і 25-15% відповідно від вихідних даних. Якісна модель повинна демонструвати прийнятні прогнозуючі здібності як на навчальній, так і на тестовій вибірках. Схожі статистичні показники, розраховані на навчальній і тестовій вибірках є ознакою того, що модель на практиці буде стабільнішою та буде показувати адекватні прогнози.

Більш складна стратегія валідації моделі передбачає формування трьох і більше вибірок: перша вибірка використовується для оцінки параметрів моделі, друга вибірка використовується для перевірки моделі, якщо отримано значні відхилення результатів за навчальною і тестової вибірками, то з них видаляються викиди або змінні, що впливають на відхилення, і будується нова модель по об’єднанню першої і другої вибірки, результати нової моделі тестуються на третій вибірці.

2.3 Методи та підходи щодо оцінювання скорингових моделей

2.3.1 Прості методи оцінки параметрів моделі

Для перевірки та оцінки якості скорингових моделей використовуються сукупність статистичних критеріїв, засобів та процедур.

Одною з найбільш поширених оцінок якості моделей в задачах прогнозування є середня абсолютна (MAE – Mean Absolute Error) та середня квадратична (MSE – Mean Squared Error) помилки:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(d_i - y_i)^2, \tag{2.22}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|d_i - y_i|, \tag{2.23}$$

де N – кількість спостережень;

d_i – реальне значення цільової змінної i-го спостереження;

y_i – прогнозне значення.

Середньоквадратична помилка є значно чутливішою до великих відхилень у порівнянні з середньоабсолютною, і тому більш чутлива до викидів. При використанні будь-якої з двох помилок корисно буде провести аналіз об’єктів, що дають найбільшу помилку.

Середньоквадратична помилка дає гарні результати при порівнянні двох моделей або при контролі якості на етапі навчання, але не дає можливості зробити висновки про адекватності даної моделі. Наприклад, значення MSE = 10 є дуже поганою характеристикою моделі, цільова змінна якої приймає значення від 0 до 1, і навпаки дуже хорошим, якщо цільова змінна знаходиться в інтервалі (10000, 100000). У таких видках замість середньоквадратичної помилки прийнято використовувати коефіцієнт детермінації, або коефіцієнт R^2:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(d_i - y_i)^2}{\sum_{i=1}^{N}(\bar{y} - y_i)^2}, \qquad (2.24)$$

де $\bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i$ – середнє значення цільової змінної.

Якщо коефіцієнт детермінації прямує до одиниці, то модель є адекватною та має хороші прогностичні якості, якщо ж він наближається до нуля, то прогнозна здатність такої моделі можна порівняти за якістю з константним прогнозуванням.

Загальна точність моделі (CA – англ. Common Accuracy) – обчислюється як відношення вірно спрогнозованих значень до загальної кількості значень *N*:

$$CA = \frac{кількість\ вірно\ спрогнозованих\ значень}{N}. \quad (2.25)$$

В ідеалі CA повинен прямувати до 1.

2.3.2 ROC-крива та індекс Gini

Також ефективним способом оцінки точності моделі, що класифікує вхідні дані на два класи є побудова і аналіз ROC-кривої (Receiver Operating Characteristic. ROC-крива відображає залежність долі правильно класифікованих позитивних прикладів від долі неправильно класифікованих негативних прикладів. Перші частки називаються істинно позитивними, а інші частки – хибно негативними. Також передбачається, що у класифікатора є певний змінний параметр, зміна якого дозволить отримувати те чи інше розбиття. Цей параметр має назву порог відсікання (cut-off value). Залежно від його значення в результаті будуть різні значення помилок I і II роду [18].

Розглянемо більш детально таблицю спряженості (confusion matrix), вона будується базуючись на результатах класифікації моделі і фактичної приналежності прикладів класам (табл.2.6), де:

а) TP (True Positives) – вірно класифіковані позичальники, що повернули кредит;
б) TN (True Negatives) – вірно класифіковані позичальники, що не повернули кредит;
в) FN (False Negatives) –позичальники, що повернули кредит, класифіковані як ті що не повернули (помилка I роду);
г) FP (False Positives) –позичальники, що не повернули кредит, класифіковані як ті що повернули (помилка II роду).

Таблиця 2.6 – Таблиця спряженості

	Дійсна приналежність	
Результат прогнозування	Негативний	Позитивний
Негативний	TN (Істинно негативний)	FN (Хибно негативний)
Позитивний	FP (Хибно позитивний)	TP (Істинно позитивний)

Зазвичай для оцінки якості моделі використовуються не абсолютні значення, а відносні – долі (rates), виражені у відсотках:

Доля істинно позитивних прикладів (True Positives Rate):

$$TPR = \frac{TP}{TP + FN} * 100\%. \tag{2.26}$$

Доля хибно позитивних прикладів (False Positives Rate):

$$FPR = \frac{FP}{TN + FP} * 100\%. \tag{2.27}$$

Для побудови ROC-кривої вводиться ще наступні поняття: чутливість і специфічність моделі. За допомогою цих понять визначається об’єктивна значущість будь-якого бінарного класифікатора.

Чутливість (Sensitivity) моделі – це доля істинно позитивних випадків:

$$Sensitivity = TPR = \frac{TP}{TP + FN} * 100\%. \tag{2.28}$$

Специфічність (Specificity) моделі – доля істинно негативних випадків, що були коректно визначені моделлю:

$$Specificity = \frac{TN}{TN + FP} * 100\% = 100\% - FPR. \quad (2.28)$$

Зазвичай найчастіше правильно класифікує позитивні приклади модель з високою чутливістю, а модель з високою специфічністю навпаки, краще справляється з виявленням негативних прикладів.

ROC-криву (рис. 2.3) отримують у наступний спосіб:

– спочатку розраховуємо значення чутливості та специфічності для кожного значення порога відсікання, змінюючи його від 0 до 1 з певним кроком dx (наприклад, 0.01);

– будуємо графік залежності: по осі ординат відкладається значення чутливості, по осі абсцис відкладається значення розраховане наступним чином: $100\% - Specificity$ (сто відсотків мінус специфічність).

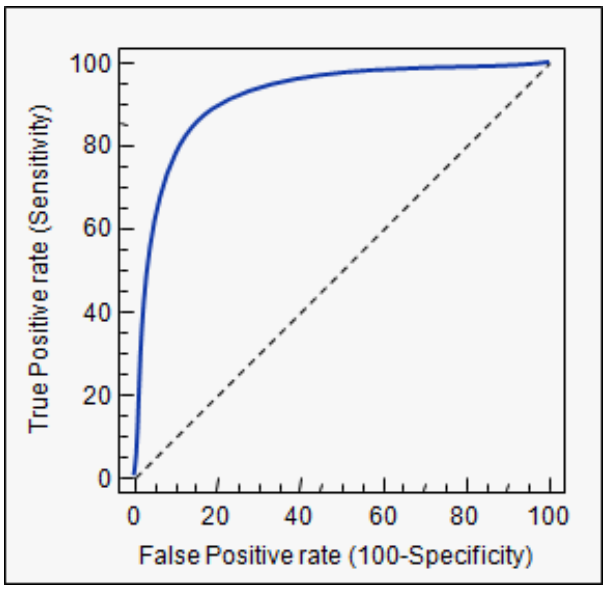


Рисунок 2.3 – Приклад побудованої ROC-кривої

Вибір оптимального значення порогового значення залежить від яка помилка є більш допустимою, першого чи другого роду при класифікації.

При зниженні порога в моделі буде переважати чутливість, тобто здатність моделі правильно виявляти позичальників, що будуть мати прострочені платежі. Також в якості оптимального порогу відсікання можна обрати точку балансу між чутливістю і специфічністю [18].

Для порівняння різних моделей (або моделей з різними параметрами) використовується площа під ROC-кривою – AUC (Area Under Curve). Площа AUC змінюється в діапазоні від 0.5 до 1 (табл. 2.7).

Таблиця 2.7 – Оцінка якості моделі за значенням площі AUC

Значення AUC	Якість моделі
0.9-1	Відмінна
0.8-0.9	Дуже добра
0.7-0.8	Добра
0.6-0.7	Середня
0.5-0.6	Незадовільна

Слід зазначити, що призначення показника площі під кривою лише для порівняльного аналізу моделей між собою. Показник площі під кривою не несе ніякої інформативності про чутливість і специфічність моделі.

При аналізі якості моделі з використання значення площі під ROC-кривою зазвичай розраховують індекс Джині. Цей показник трансформує значення площі під кривою в діапазон значень від 0 до 1, чим вище його значення, тим вище дискримінуюча здатність моделі. Індекс Джині розраховується наступним чином:

$$GINI = 2 * AUC - 1, \tag{2.30}$$

де AUC – площа по ROC-кривої.

2.3.3 Статистика Колмогорова-Смірнова

Для оцінювання прогнозної здатності моделі в кредитному скорингу використовується тест або статистика Колмогорова-Смірнова. У цьому тесті проводиться перевірка статистичної гіпотези, що дві довільні вибірки є складовою однієї генеральної сукупності. У випадку скорингу відбувається порівняння двох кумулятивних розподілів скорингових балів «хороших» і «поганих» позичальників [19]. Значення статистики Колмогорова-Смірнова (рис. 2.4) розраховується як максимальна різниця між значеннями кумулятивних функцій розподілу «поганих» і «хороших» клієнтів:

$$KS = \max_x \left| F_m(x) - G_n(x) \right| * 100\%, \tag{2.31}$$

де $F_m(x)$ – кумулятивний розподіл скорингового балу для «поганих» клієнтів;

$G_n(x)$ – кумулятивний розподіл скорингового балу для «хороших» клієнтів;

m – кількість «поганих» клієнтів;

n– кількість «хороших» клієнтів.

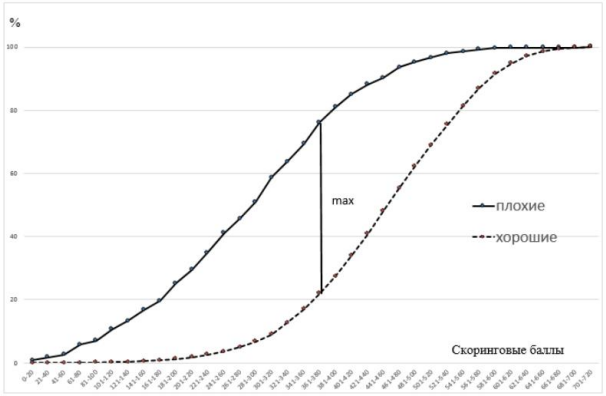


Рисунок 2.4 – Ілюстрація розрахунку статистики Колмогорова-Смірнова

Визначимо алгоритм обрахунку статистики Колмогорова-Смірнова і перевірки гіпотези рівності двох функцій розподілу. Ранжуємо клієнтів в порядку збільшення скорингового балу і групуємо їх. Ознакою групування виступає отриманий скоринговий бал [19]. Далі в кожній отриманій групі клієнтів розраховуємо наступні показники:

- кількість «хороших» клієнтів;
- кількість «поганих» клієнтів;
- відношення шансів «поганих» до «хорошим» клієнтів;
- відсоток «поганих» та «хороших» позик;
- кумулятивну суму «поганих» та «хороших» позик;
- кумулятивний відсоток «поганих» та «хороших» позик;
- загальний кумулятивний відсоток поганих позик від їх загальної кількості;
- різницю між кумулятивними відсоткам поганих і хороших позик.

Після чого необхідно вирахувати максимальну різницю між кумулятивним відсотком «хороших» і «поганих» позик і розрахувати за формулою значення статистики Колмогорова-Смірнова. Обчислене значення порівнюється із значенням взятим з таблиці розподілу Колмогорова-Смірнова з вибраним рівнем значущості або при числі «поганих» і «хороших» клієнтів більше 80 можна обрати наближене порогове значення, що розраховується за наступною формулою:

$$z(a)\sqrt{\frac{m+n}{mn}}, \tag{2.32}$$

де $z(a)$ – значення, що відповідає обраному рівнем значущості.

Якщо розраховане значення статистики за формулою (2.31) менше значення по таблиці або значення розрахованого за формулою (2.32), то гіпотеза рівності двох функцій розподілів відкидається.

Значення статистики Колмогорова-Смірнова можуть змінюватися в діапазоні від 0 до 100. Високі значення статистики Колмогорова-Смірнова, говорять про кращу здатність моделі до класифікації. Зазвичай значення статистики Колмогорова-Смірнова, лежать в діапазоні від 20-25 до 75-80, а крайні значення статистика не приймає.

Висновки до розділу 2

В другому розділі було проведено огляд існуючих математичних методів прогнозування, які можна використовувати для оцінювання кредитоспроможності фізичних осіб, а саме дерева рішень, лінійна імовірнісна модель, логістична регресія та скорингова карта.

Метод дерев рішень є досить зручним способом візуалізації процесу прийняття рішення щодо видачі кредиту у вигляді ієрархічної, послідовної структури правил. Однак для великого об’єму даних, які можуть приймати широкий діапазон значень, побудована модель дерева рішень буде досить складною. Було розглянуто різні алгоритми побудови дерев рішень. Найбільш ефективним та швидким алгоритмом вважається статистичний метод побудови бінарних дерев – QUEST.

Розглянуто лінійну імовірнісну модель, у формі лінійної регресії, яка є досить простою у побудові та розрахунках. Дана модель має ряд недоліків. Було запропоновано способи вирішення певних недоліків та досліджено такі вдосконалені методи ЛІМ, як логістична регресія та скорингова карта.

У розділі наведено алгоритми, які використовують для побудови та навчання логістичної регресії.

Досліджено процес попереднього аналізу і обробки даних для побудови скорингової моделі, який включає в себе: збір даних, обробку пропусків, визначення цільової змінної, відбір найбільш значущих змінних, формування навчальної та тестової вибірки. Запропоновано метод перетворення категоріальних змінних у числові, для використання змінних у регресійних моделях.

Також були розглянуті критерії оцінки якості отриманих прогнозуючих моделей: загальна точність моделі, ROC-крива, індекс Gini, а також помилки 1-го й 2-го роду. Останні особливо важливі з погляду фінансової установи, оскільки помилка 1-го роду означає, що банк втратить певну суму грошей, а помилка 2-го роду означає, що банк не доодержить деякий прибуток.

РОЗДІЛ 3 СИСТЕМА ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ ДЛЯ ВИЗНАЧЕННЯ КРЕДИТОСПРОМОЖНОСТІ ФІЗИЧНИХ ОСІБ

У цьому розділі наведений опис розробленої в рамках магістерської дисертації комп'ютерної програми SAB Analytical Studio. Програма призначена для прогнозування кредитоспроможності фізичних осіб на основі застосування теорії регресійного аналізу. Дана програма реалізована мовою програмування C# в середовищі розробки Microsoft Visual Studio 2012.

Програмний продукт дозволяє користувачам з різним рівнем підготовки проводити необхідну попередню обробку даних для побудови прогнозуючої моделі, будувати скорингову модель, та одержувати статистичні характеристики й прогнозні дані на основі побудованої моделі.

Інтерфейс користувача інтуїтивно зрозумілий і створений таким чином, щоб провести оператора від моменту завантаження вхідних даних до виведення та збереження результатів.

Для обчислення коефіцієнтів рівняння логістичної регресії в програмі реалізовано метод максимальної правдоподібності з використанням методу градієнтного спуску. Спеціально для полегшення реалізації алгоритму ММП був розроблений модуль Matrix, який представляє собою набір процедур для роботи з матрицями.

За технічним рівнем SAB Analytical Studio належить до настільних програмних продуктів, тому що обслуговує тільки один користувацький комп'ютер. Система не розрахована на мережеву роботу.

3.1 Аналіз архітектури системи

У цьому розділі розглядається система підтримки прийняття рішень для прогнозування кредитоспроможності фізичних осіб із застосуванням методу логістичної регресії.

На рисунку 3.1 наведена структура розробленої СППР. Як можна побачити з цієї структури СППР представляє собою широкий комплекс засобів для аналізу та обробки даних.

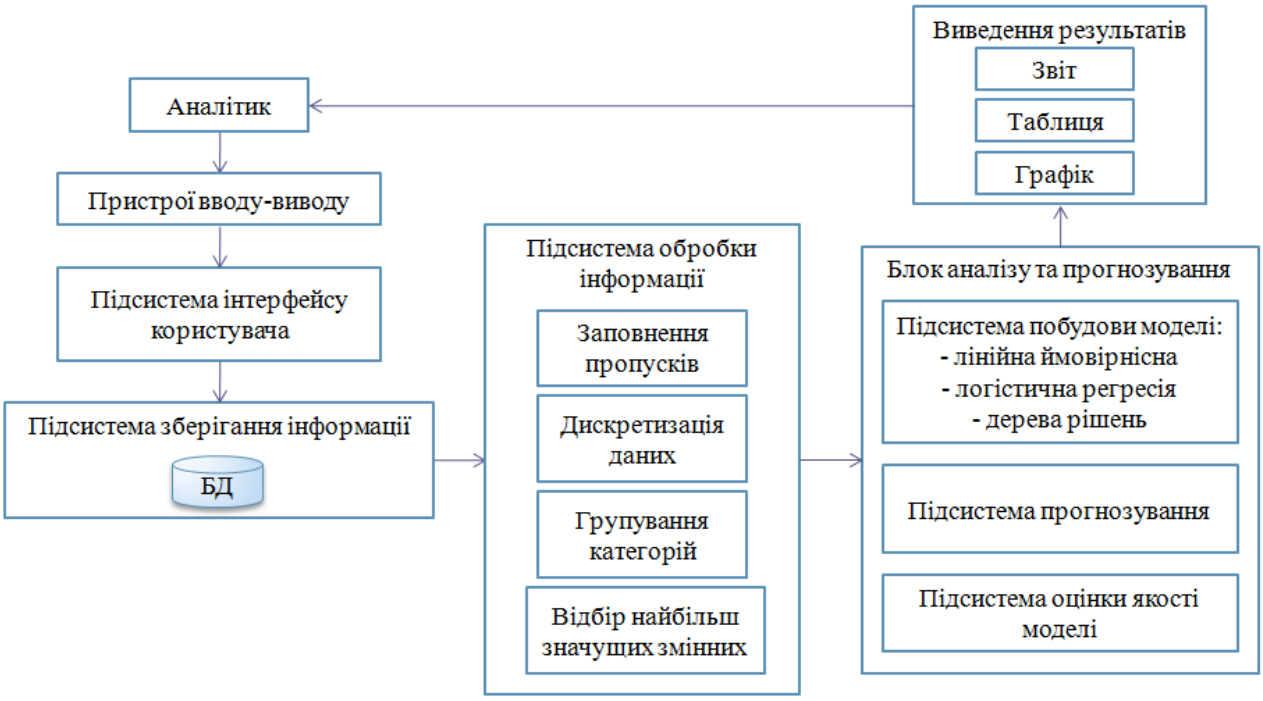


Рисунок 3.1 – Структура системи підтримки прийняття рішень

Пристрої вводу-виводу надають користувачу можливість завантажувати дані в СППР. Для цього підсистема вводу-виводу функціонально зв’язана з підсистемою інтерфейсу користувача.

Підсистема інтерфейсу користувача призначена для здійснення зв’язку між користувачами СППР та внутрішніми елементами системи і забезпечує ввід та вивід інформації для ОПР і експертів, а також надає доступ до

зовнішніх запам’ятовуючих пристроїв ПЕОМ. Інтерфейс дозволяє операторові вводити інформацію, дані, команди, параметри і запити в систему та одержувати вихідну інформацію в зручному для сприйняття вигляді.

Підсистема зберігання інформації складається з бази даних, яка призначена для накопичення статистичних даних, з метою їх подальшого аналізу, обробки та використання.

Підсистема обробки інформації призначена для перевірки даних, що поступають, на наявність пропусків, заповнення цих пропусків, категоризації неперервних даних.

Блок аналізу та прогнозування складається з трьох підсистем: підсистема побудови моделі, підсистема прогнозування, підсистема оцінювання якості моделі.

Підсистема побудови моделі реалізована за допомогою логістичної регресії з використанням методу максимальної правдоподібності.

Підсистема оцінювання якості моделі обчислює статистичні показники побудованої моделі, які характеризують прогнозуючу здатність моделі.

Підсистема виведення результатів представляє собою набір графіків, таблиць та звітів для прийняття рішення експертом. Представлення результатів прогнозування та критеріїв оцінювання моделі дає змогу зробити висновки щодо можливості використання отриманої моделі для прогнозування.

Для створення СППР застосовувались технології .Net та середовище розробки Microsoft Visual Studio 2012.

Оскільки для роботи методу ММП необхідно працювати з матрицями, то спеціально був розроблений модуль Matrix, який представляє собою набір процедур для роботи з матрицями.

На теперішній час більшість людей користується саме операційними системами класу Windows, тому інтерфейс користувача був створений на базі технологій WinForms з використанням вікон та іконок. Це спрощує роботу

користувача, робить зайвим запам’ятовування великої кількості команд для виконання певних дій та подає отримані результати у зручному вигляді.

3.2 Основні технічні вимоги для коректної роботи програми

Для роботи програмного продукту необхідна наявність персонального комп’ютера з наступними мінімальними характеристиками:

а) операційна система Windows 7/8/10;
б) тактова частота процесора 1 ГГц;
в) оперативна пам’ять розміром 512 Мбайт;
г) вільний дисковий простір: 5 Мбайт для розміщення виконавчого файлу, вхідних даних і результатів роботи;
д) клавіатура та комп’ютерна мишка;
е) монітор з розподільчою здатністю 1024×768;
ж) інсталяція .Net Framework версії 4.5.

3.3 Інструкція з експлуатації програмного продукту

Робота з усіма елементами інтерфейсу є стандартною для програмного забезпечення, що працює на платформі операційної системи MS Windows. Усі можливі не коректні введення даних обробляються системою та попереджують користувача інформаційними повідомленнями.

Основний робочий екран ПП має структуру наведену на рисунку 3.2.

Головну форму програми можна інтуїтивно розділити на три частини:

– меню користувача («Область 1» на рис. 3.2), яке складається з команд для завантаження файлів даних, обробки та аналізу даних, побудови прогнозуючої моделі, збереження файлів;
– дерево проекта («Область 2» на рис. 3.2), яке містить три батьківські вершини, що характеризують процес роботи системи: Data files (Файли даних), Prediction models (Прогнозуючі моделі), Results (Результати);
– робоча область програми («Область 3» на рис. 3.2), в якій відкриваються обрані вікна з дерева проекту.

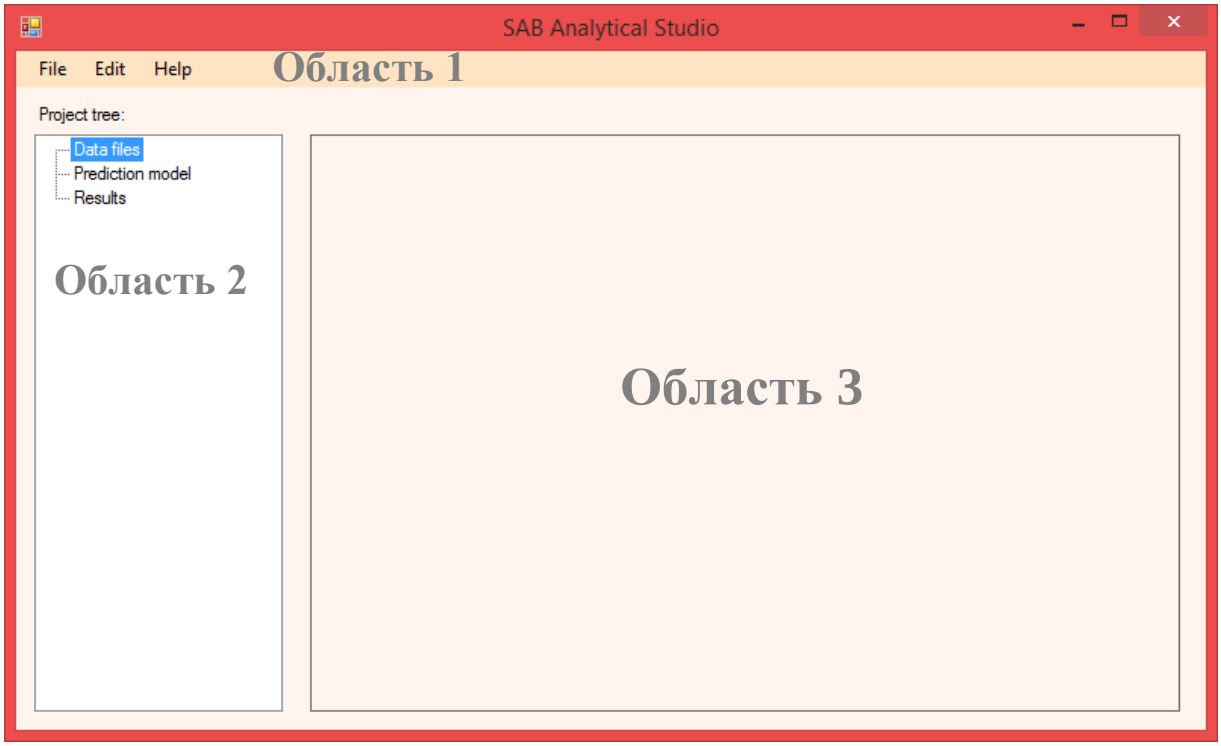


Рисунок 3.2 – Структура основного робочого екрану

3.3.1 Завантаження даних

Програма дозволяє завантажувати довільну кількість наборів даних, які можуть бути в форматах – xls/xlsx/csv. Для підготовки даних до аналізу найкраще застосовувати програму Microsoft Excel. Дані можуть приймати як числові, так і строкові значення.

Для завантаження даних необхідно перейти по головному меню: File (Файл) → Open Data file (Відкрити файл даних). Після чого відкриється стандартне діалогове вікно Windows вибору директорії з файлами (рис. 3.3), в якому слід обрати формат необхідного файлу з даними, та безпосередньо сам файл даних.

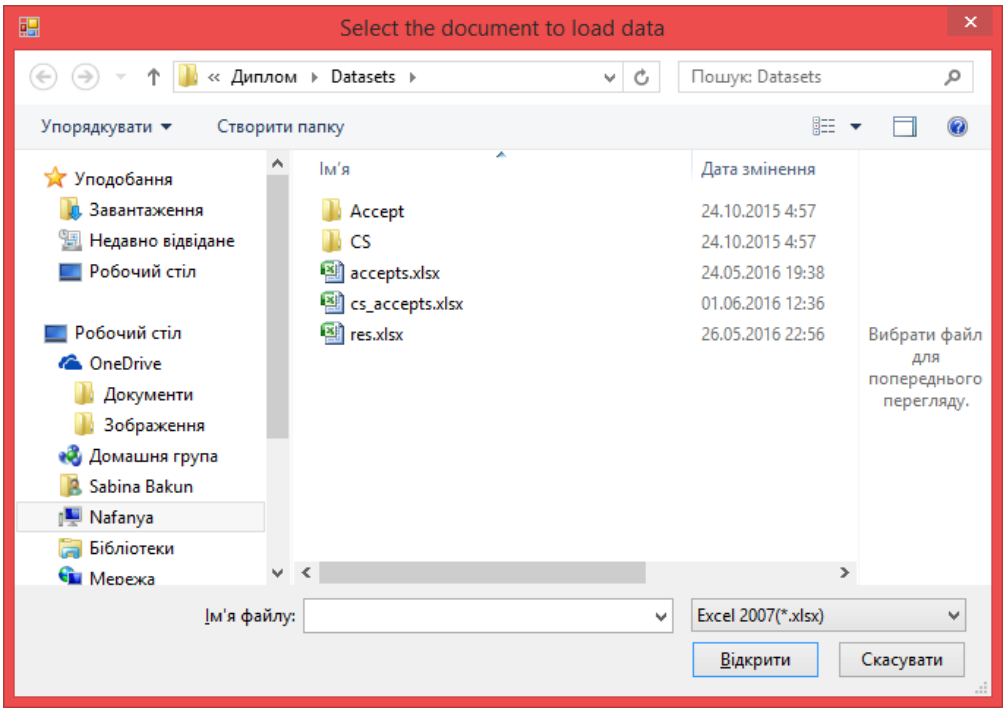


Рисунок 3.3 – Вибір файлу для завантаження

Після вибору файлу даних відкриється вікно з попереднім переглядом метаданих про обраний файл (рис. 3.4), а саме буде відображено: формат файлу та шлях його розташування в пам’яті комп’ютера. У цьому вікні

необхідно вказати чи містить набір даних перший рядок з назвами змінних, поставивши або прибравши відповідний прапорець.

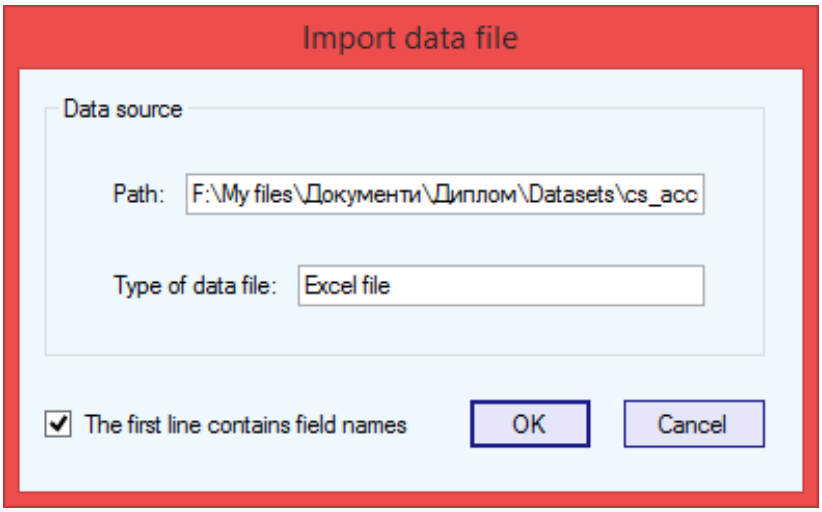


Рисунок 3.4 – Вікно з інформацією про обраний файл

Після натискання кнопки «ОК» назва файлу додається до дерева проекту в вершину Data files (Файли даних), дані файлу заносяться до аналітичної системи і відображаються на головному екрані програми, що зображено на рисунку 3.5.

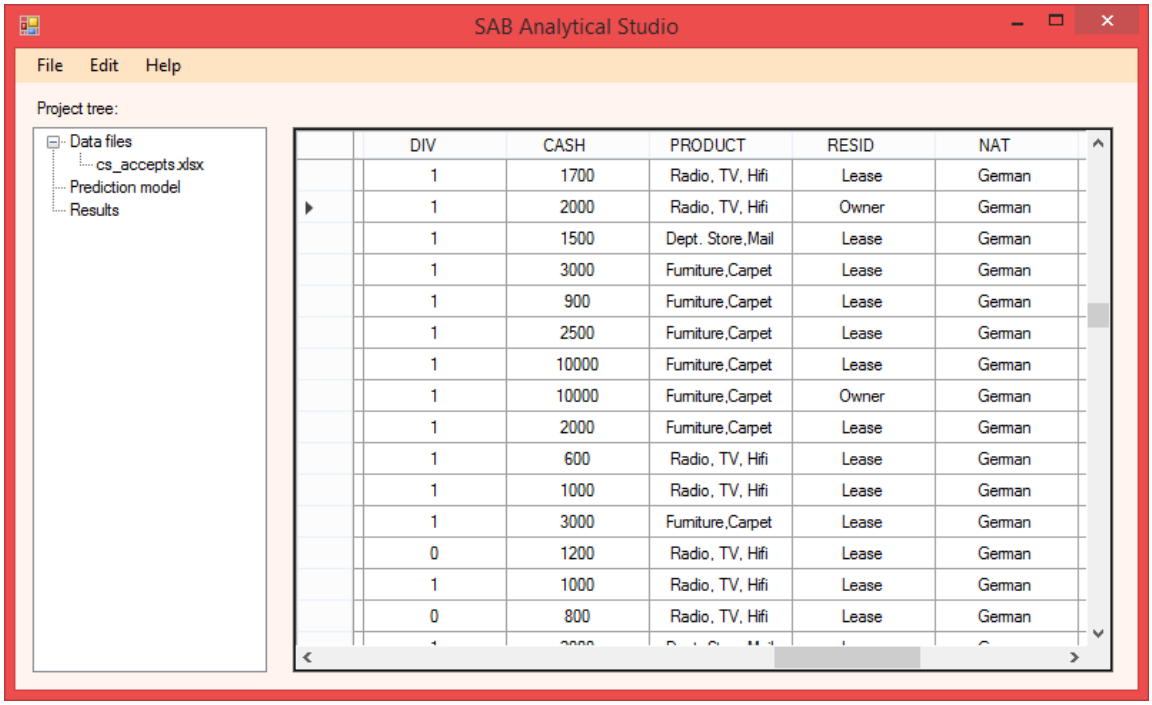


Рисунок 3.5 – Завантажені в програму дані

3.3.2 Обробка вхідних даних

Розроблена інформаційно-аналітична система надає користувачам такі можливості виконання попередньої обробки даних, як заповнення пропусків та дискретизація неперервних значень.

Для того, щоб відкрити вікно, яке дозволяє заповнювати пропуски даних, необхідно перейти по головному меню: Edit (Редагувати) → Missing value (Відсутні значення). Після цього з'явиться вікно, зображене на рисунку 3.6.

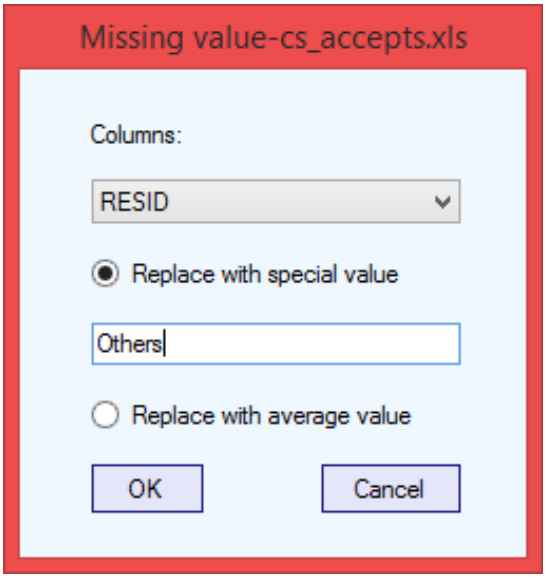


Рисунок 3.6 – Вікно для обробки пропусків даних

У даному вікні можна обрати один із стовпчиків, які містять пропуски, та метод заповнення пропусків: спеціальним значенням, вказавши його, або середнім значенням (для неперервних змінних). Після натиснення кнопки «ОК» та успішного виконання процедури, з'явиться інформаційне вікно (рис. 3.7), в якому буде написано скільки порожніх значень було заповнено в обраному стовпчику.

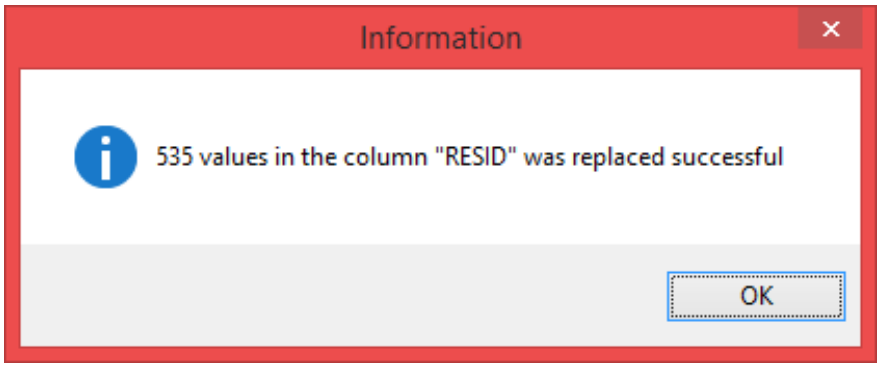


Рисунок 3.7 – Інформаційне вікно після заповнення пропусків

Для того, щоб відкрити вікно, яке дозволяє трансформувати неперервні дані в категоріальні, необхідно перейти по головному меню: Edit (Редагувати) → Discretization (Дискретизація). Після цього з’явиться вікно, зображене на рисунку 3.8.

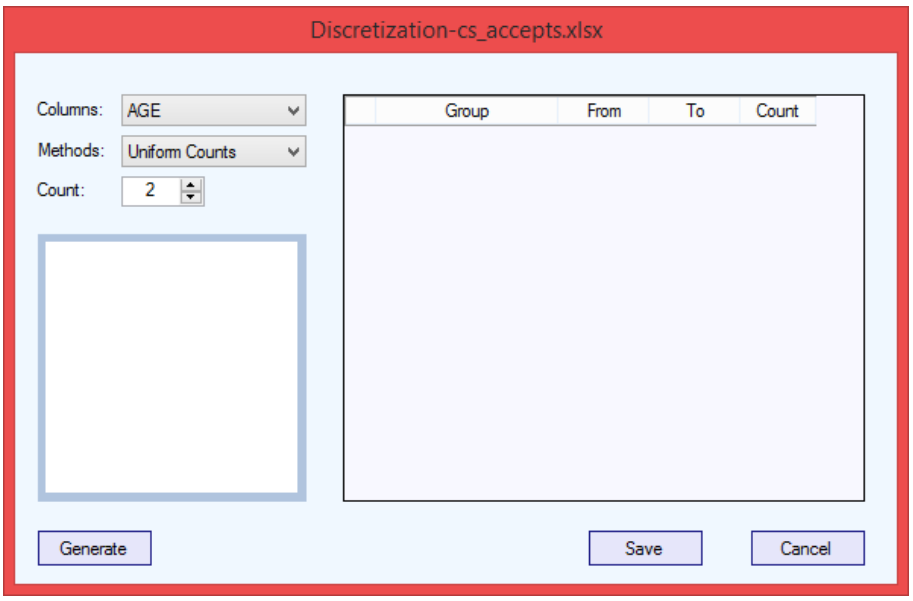


Рисунок 3.8 – Вікно для дискретизації даних

У даному вікні необхідно обрати неперервну змінну, яку необхідно дискретизувати, кількість категорій (від 2 до 20), та один з методів дискретизації: однакова ширина проміжків (Uniform Width); однакова кількість спостережень (Uniform Count).

Після натискання кнопки «Generate» (Згенерувати), у вікні з’явиться кругова діаграма розподілу спостережень за категоріями та табличка з даними по кожній з категорій (рис. 3.9 – 3.10).

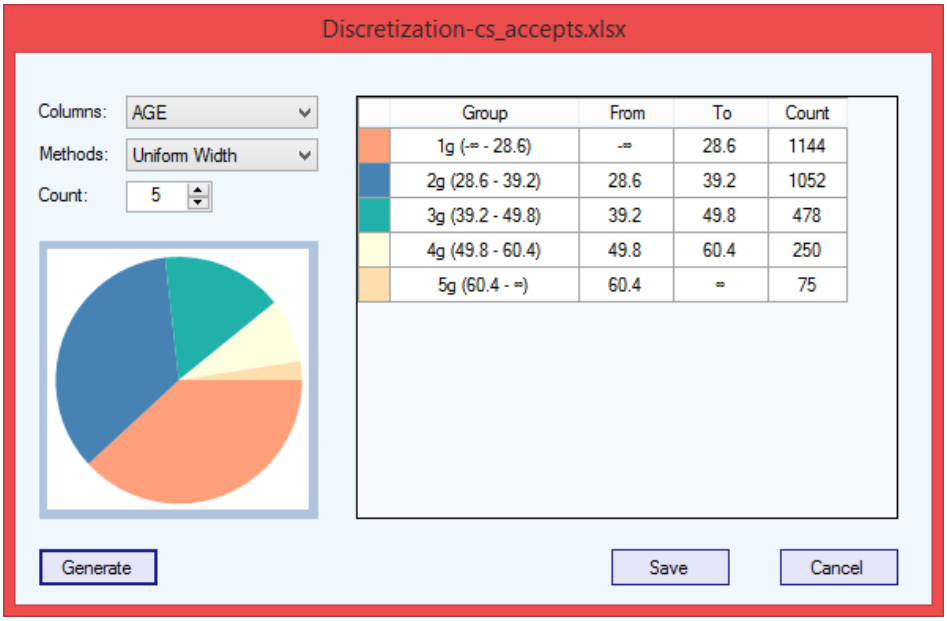


Рисунок 3.9 – Дискретизація даних методом Uniform Width

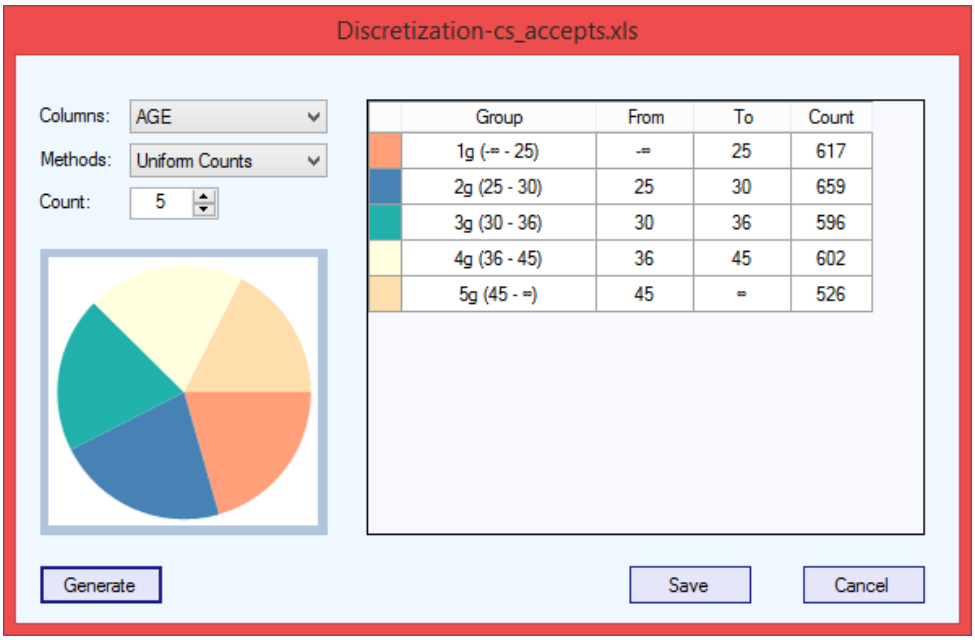


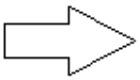
Рисунок 3.10 – Дискретизація даних методом Uniform Count

Для збереження даних, необхідно натиснути кнопку «Save» (Зберегти), після чого відкриється головне вікно програми, в якому обраний для

дискретизації стовпчик з неперервними даними заміниться на відповідний йому стовпчик зі значеннями категорій (рис. 3.11).

До дискретизації

AGE
46
34
31
39
32
23
42
35
26
24
20
44
23
44
36



Після дискретизації

AGE
5g (45-∞)
3g (30-36)
3g (30-36)
4g (36-45)
3g (30-36)
1g (-∞-25)
4g (36-45)
3g (30-36)
2g (25-30)
1g (-∞-25)
1g (-∞-25)
4g (36-45)
1g (-∞-25)
4g (36-45)
4g (36-45)

Рисунок 3.11 – Результат дискретизації даних

3.3.3 Побудова прогнозуючої моделі

Основною функцією програми є побудова прогнозуючої моделі на основі логістичної регресії. Побудова моделі виконується в два кроки. Спочатку необхідно перейти по головному меню: Edit (Редагувати) → Build model (Побудова моделі). Після цього з’явиться перше вікно, зображене на рисунку 3.12.

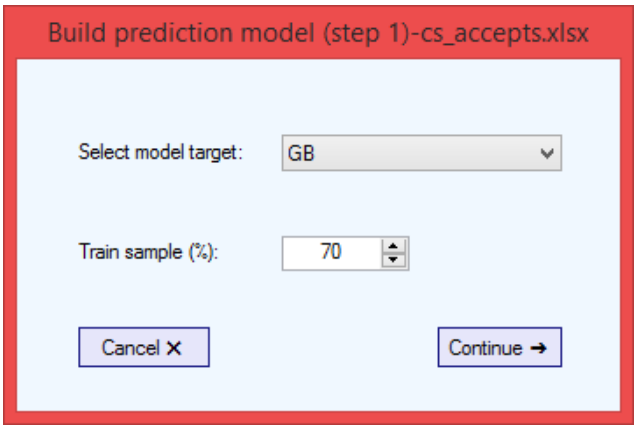


Рисунок 3.12 – Вікно побудови моделі (крок 1)

У цьому вікні необхідно вибрати цільову змінну прогнозування, яка приймає тільки два значення, тобто є бінарною, встановити відсоток набору даних, який буде використовуватися для навчання моделі та натиснути кнопку «Continue» (Продовжити), щоб перейти на наступний крок побудови моделі. Якщо обрана цільова змінна виявиться не бінарною програма видасть помилку (рис. 3.13).

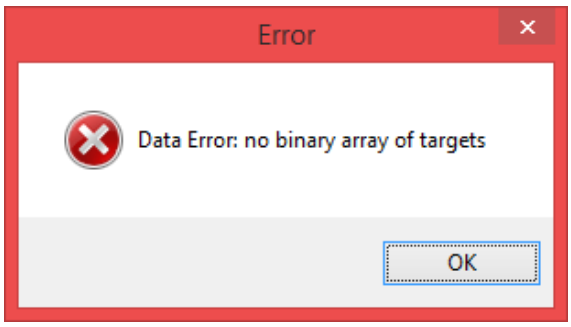


Рисунок 3.13 – Повідомлення про невідповідність цільової змінної

Якщо цільова змінна приймає коректні значення відкриється наступне вікно побудови моделі (рис. 3.14).

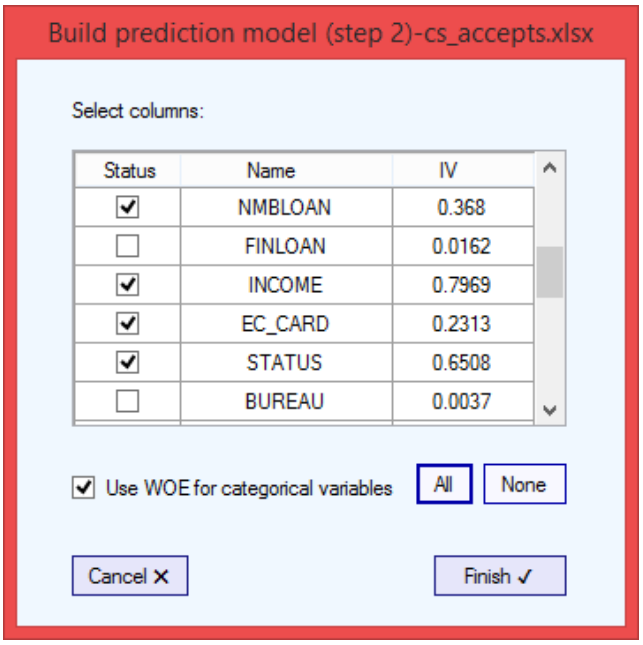


Рисунок 3.14 – Вікно побудови моделі (крок 2)

В даному вікні виводяться назви всіх стовпчиків набору даних, окрім обраної цільової змінної, та їх інформаційне значення (IV). Програма автоматично виділяє прапорцями ті стовпці, у яких інформаційне значення вище за 0.03. Користувач може сам обрати необхідні стовпці для побудови моделі, а також виділити всі змінні натиснувши «All» (Все), або зняти прапорці зі всіх змінних за допомогою кнопки «None» (Нічого).

Оскільки логістична регресія працює тільки з числовими значеннями, програмно реалізовано перекодування категоріальних змінних в числові за допомогою порядкової нумерації категорій, або за допомогою використанням значень коефіцієнта зваженої сукупності (WOE). За замовчуванням використовується перший варіант кодування, для того щоб обрати другий варіант, необхідно поставити прапорець «Use WOE for categorical variables» (Використання WOE для категоріальних змінних).

Після налаштування всіх параметрів необхідно натиснути «Finish» (Завершити) для побудови моделі та прогнозування.

3.3.4 Виведення результатів прогнозування

Результатом виконання операції побудови моделі є два вікна, які додаються в дерево проекту у відповідні вузли «Prediction model» (Прогнозуюча модель) та Results (Результати).

Перше вікно називається «Model» (Модель) та містить у собі таблицю з відновленою логістичною моделлю, а саме з оцінками параметрів логістичної регресії (рис. 3.15). Також у цьому вікні приведено обчислені статистичні коефіцієнти оцінки якості моделі, такі як загальна точність моделі (CA) та індекс GINI, і графічна характеристика – ROC-крива.

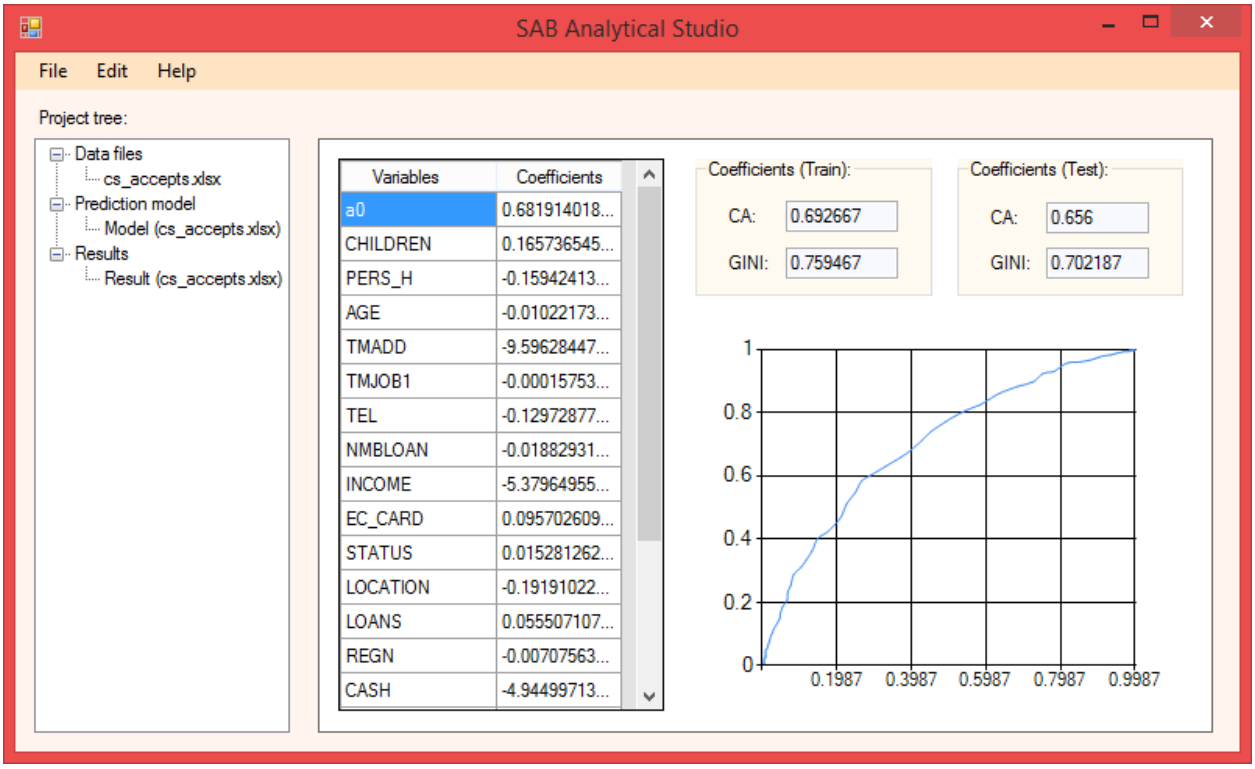


Рисунок 3.15 – Вікно з результатами побудови моделі

Друге вікно називається «Result» (Результат) та відображає таблицю з даними, які приймали участь в прогнозуванні, відновленні значення побудованої моделі, та інформацію про наявність похибок першого й другого роду (рис. 3.16).

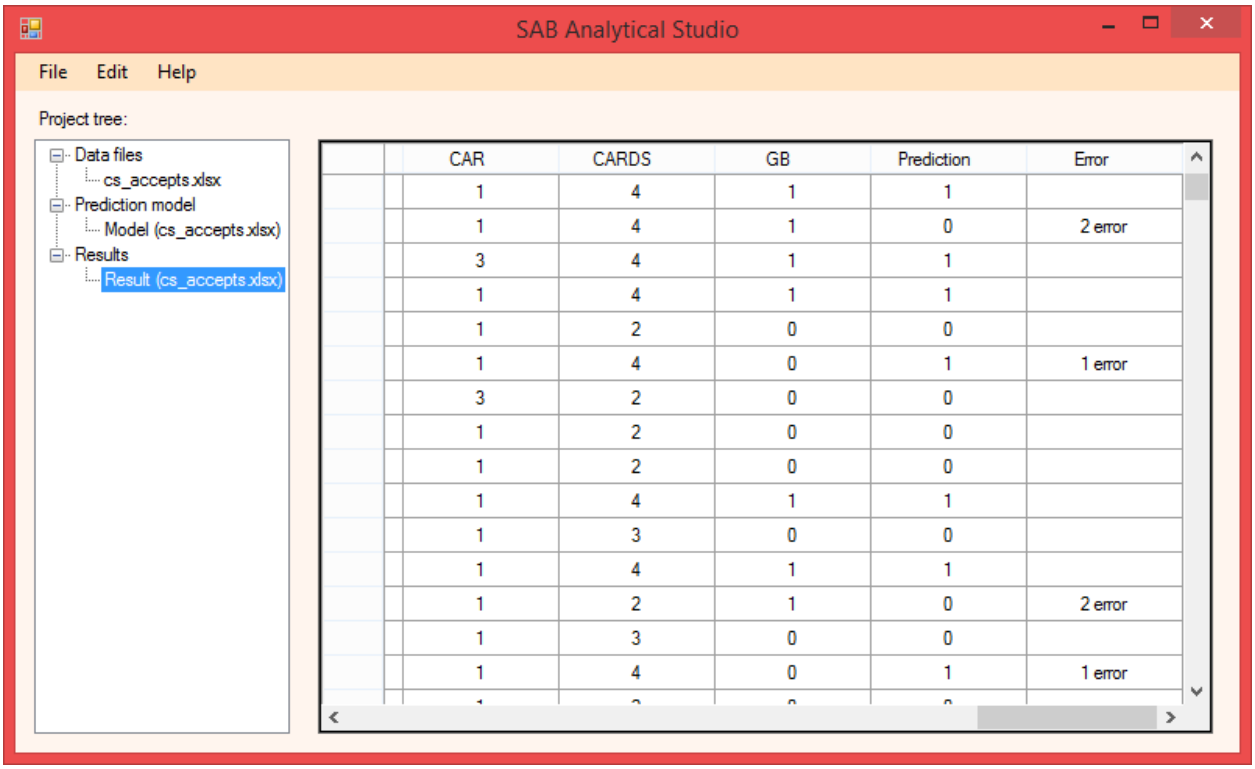


Рисунок 3.16 – Вікно з результатами прогнозування

Програма дозволяє зберігати отримані результати прогнозування та параметри побудованої моделі. Для цього слід відкрити на головній формі програми відповідне вікно з інформацією, яку необхідно зберегти, натиснути File (Файл) → Save (Зберегти), після чого відкриється діалогове вікно, в якому треба вказати ім’я файлу та шлях місця розташування для збереження.

3.4 Результати апробації програмного продукту

Для апробації програмного продукту на реальних статистичних даних було використано вибірку даних з німецького банку, що надає кредити фізичним особам.

Набір даних містить 3000 записів по клієнтах, у яких вже закінчився строк кредитування, та включає в себе інформацію щодо 17 показників (анкетних даних) по кожній особі.

Опишемо кожен показник більш детально (табл. 3.1).

Таблиця 3.1 – Опис змінних набору даних

Назва змінної	Опис
AGE	Вік
CAR	Наявність транспортного засобу
CARDS	Тип банківської карти
CASH	Запитувані грошові кошти
CHILDREN	Кількість дітей
EC_CARD	Наявність банківської карти
FINLOAN	Кількість закритих кредитів
GB	«Хороший» чи «поганий» клієнт
INCOME	Дохід особи
LOANS	Кількість відкритих кредитів
NAT	Національність особи
NMBLOAN	Кількість кредитів, наданих цим банком
PRODUCT	Ціль кредиту
REGN	Регіон
RESID	Тип місця проживання
STATUS	Сімейний стан
TITLE	Стать особи

Вихідний набір даних було розбито на навчальну та тестову вибірки розміром 70% та 30% відносно даного набору.

Для аналізу запропонованого методу трансформації категоріальних змінних у числові було побудовано дві скорингові моделі: з використанням порядкової нумерації категорій, та з використанням коефіцієнтів WOE (рис. 3.17-3.18).

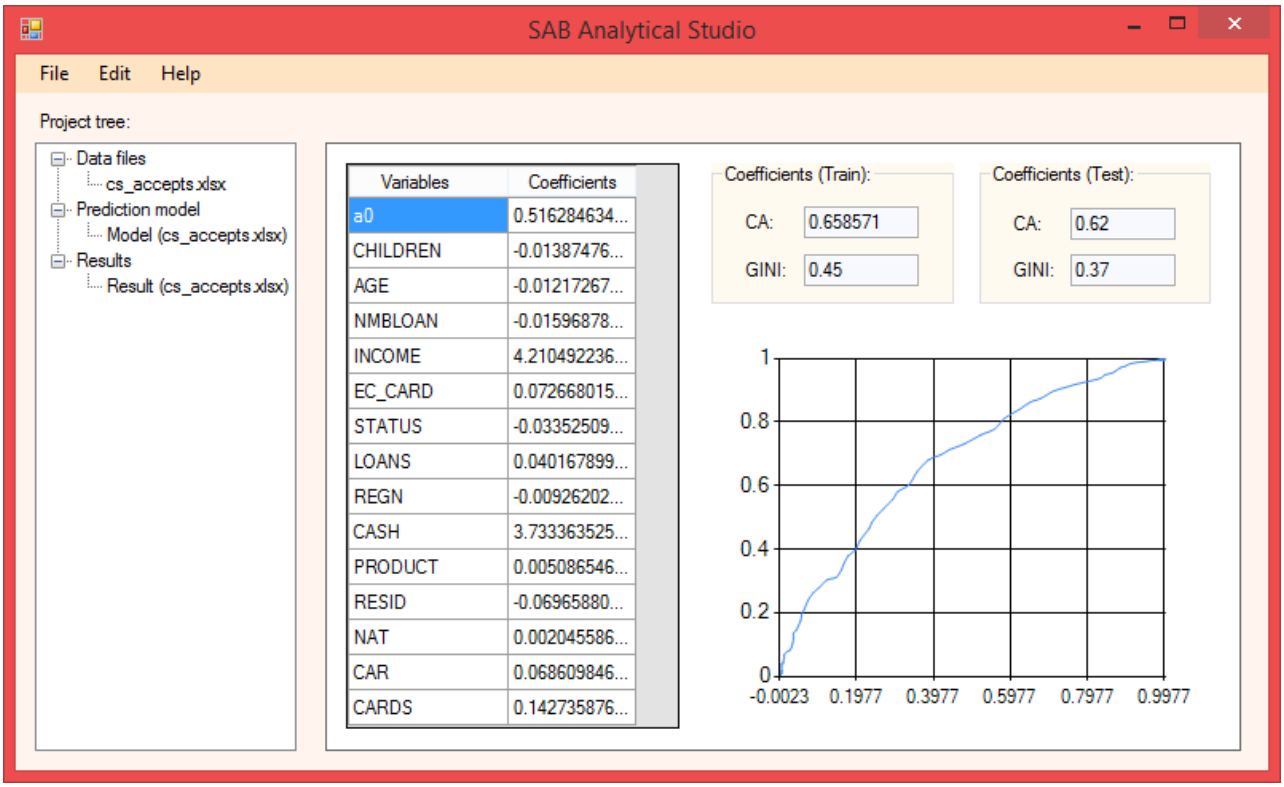


Рисунок 3.17 – Результати прогнозування без використання WOE

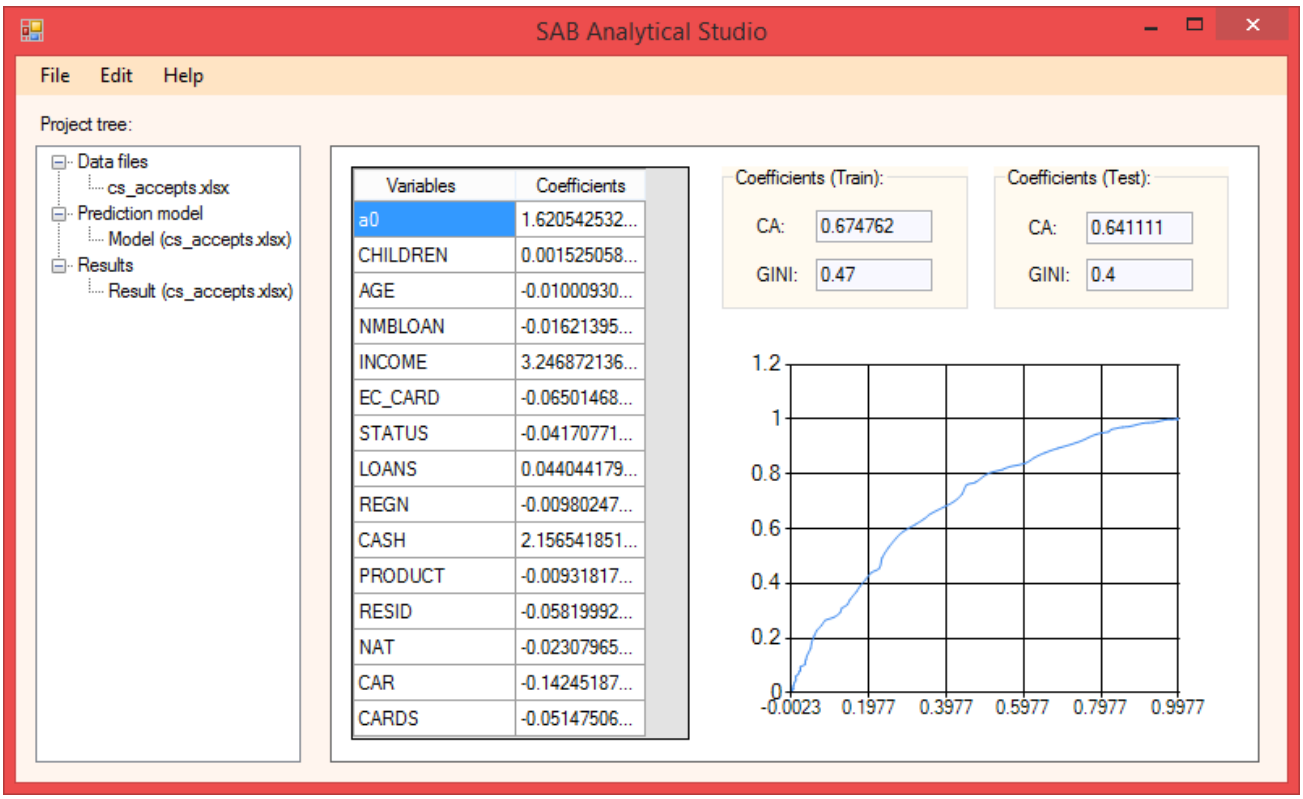


Рисунок 3.18 – Результати прогнозування з використанням WOE

Статистичні характеристики якості побудованих моделей було зведено до таблиці 3.2.

Таблиця 3.2 – Порівняльна таблиця характеристик якості скорингових моделей з використанням WOE та без використання WOE

	Навчальна вибірка (Train)		Тестова вибірка (Test)	
	Загальна точність (CA)	Індекс GINI	Загальна точність (CA)	Індекс GINI
Модель без використання WOE	0.658571	0.45	0.62	0.37
Модель з використанням WOE	0.674762	0.47	0.641111	0.4

Як можна побачити з таблиці 3.2, побудована модель з використанням коефіцієнту WOEдля трансформації категоріальних змінних в числові має кращі значення індексу GINI та загальної точності моделі на навчальній та тестовій вибірках.

Для порівняння якості побудованої скорингової моделі в програмному продукті SAB Analytical Studio з використанням методу логістичної регресії, було побудовані моделі у вигляді дерев рішень і скорингової карти в системі SAS Enterprise Miner.

Отримані моделі у вигляді скорингової карти та дерев рішень зображені на рисунках 3.19 та 3.20.

Scorecard

		Scorecard Points
Age	AGE< 23	-10
	23<= AGE< 28	1
	28<= AGE< 31	11
	31<= AGE< 46, _MISSING_	19
	46<= AGE	31
Credit Cards	CHEQUE CARD, MASTERCARD/EUROC, OTHER CREDIT CAR	32
	AMERICAN EXPRESS, NO CREDIT CARDS, VISA MYBANK, VISA OTHERS, _MISSING_, _UNKNOWN_	5
EC_card holders	0.00, _MISSING_, _UNKNOWN_	15
	1.00	6
Income	INCOME< 1000, _MISSING_	18
	1000<= INCOME< 1900	7
	1900<= INCOME< 2500	9
	2500<= INCOME< 3000	12
	3000<= INCOME	15

Рисунок 3.19 – Побудована скорингова карта в SAS Enterprise Miner

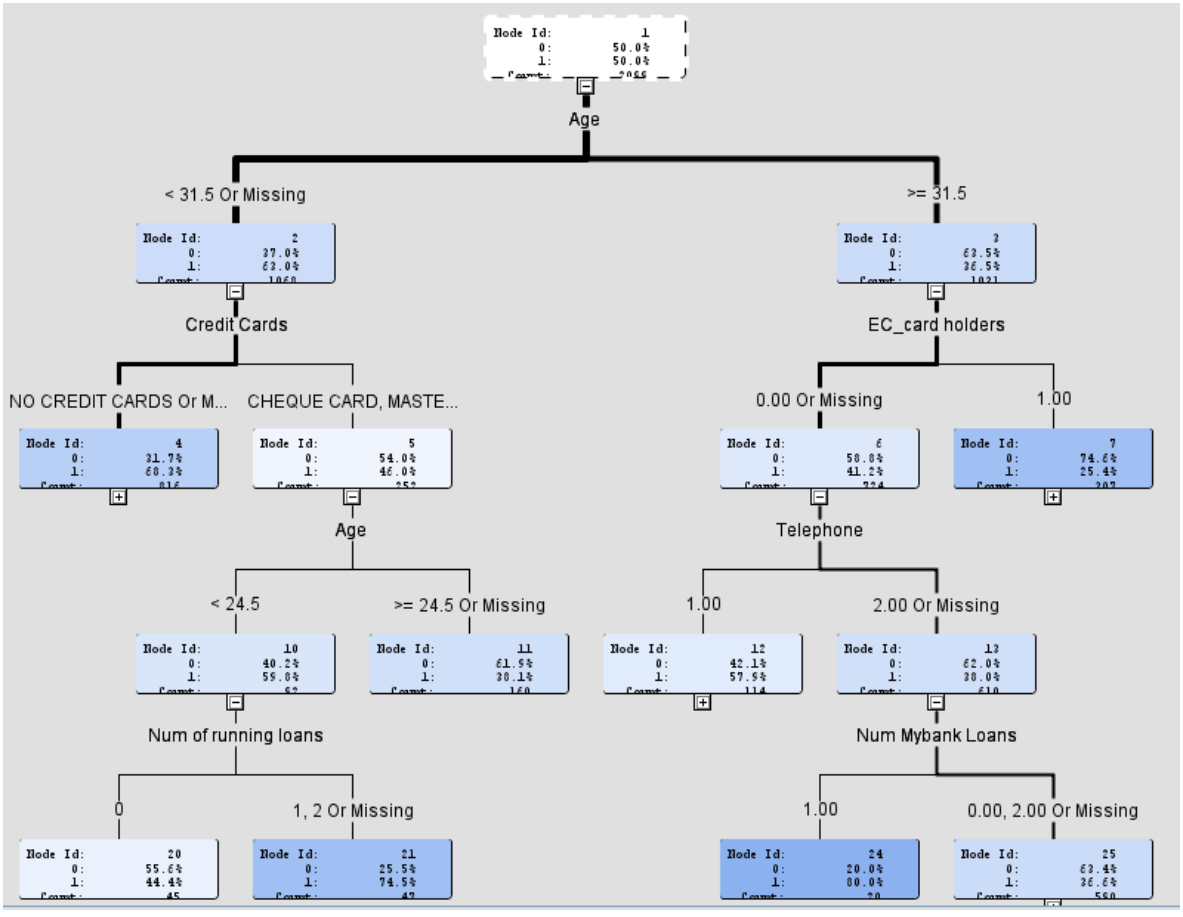


Рисунок 3.20 – Побудоване дерево рішень в SAS Enterprise Miner

Порівняння статистичних характеристик якості досліджуваних моделей наведено у таблиці 3.3

Таблиця 3.3 – Порівняльна таблиця характеристик побудованих моделей

	Навчальна вибірка (Train)		Тестова вибірка (Test)	
	Загальна точність (CA)	Індекс GINI	Загальна точність (CA)	Індекс GINI
Логістична регресія (SAB Analytical Studio)	0.674762	0.47	0.641111	0.4
Дерева рішень (SAS Enterprise Miner)	0.686041	0.445	0.681465	0.403
Скорингова карта (SAS Enterprise Miner)	0.665079	0.476	0.622642	0.419

Найкращий результат за показником загальної точності моделі (CA) дав метод дерев рішень в системі SAS Enterprise Miner, на другому місці опинилась логістична регресія, реалізована у розробленому програмному продукті, значення загальної точності якої відрізняється лише на 0.01 від загальної точності дерев рішень. За індексом GINI найкращою виявилась скорингова карта, на другому місці – логістична регресія.

Висновки до розділу 3

У третьому розділі було описано спроектовану систему підтримки прийняття рішень для оцінки кредитоспроможності фізичних осіб. Дана система складається з таких структурних елементів: пристрої вводу-виводу, підсистема інтерфейсу користувача, підсистема зберігання інформації, підсистема обробки інформації, блок аналізу та прогнозування, і виведення результатів.

На основі запропонованої СППР в рамках магістерської дисертації було розроблено ПП SAB Analytical Studio. Програмний продукт дозволяє завантажувати дані, проводити попередній аналіз та обробку даних, будувати прогнозуючі моделі, а також обчислювати статистичні характеристики якості побудованої моделі та зберігати результати прогнозування.

Визначено мінімальні технічні характеристики персонального комп'ютера для коректної та повноцінної роботи програмного забезпечення, а саме: тактова частота процесору, об'єм оперативної пам'яті, об'єм пам'яті на диску, операційна система, додаткове програмне забезпечення, що підтримує роботу розробленого програмного продукту, та периферійні пристрої необхідні для повноцінної роботи оператора.

Проведено детальний огляд інтерфейсу користувача. Розглянуто функціональні можливості програмного забезпечення та описану покрокову роботу SAB Analytical Studio з візуальним відображенням у вигляді рисунків робочого екрану програмного продукту.

Розроблений програмний продукт апробовано на реальних статистичних даних з німецького банку, що надає кредити фізичним особам. Побудовано дві скорингові моделі: з використанням коефіцієнту WOE для трансформації категоріальних змінних в числові, та з використанням порядкової нумерації. Порівнявши статистичні характеристики побудованих моделей можна побачити, що використання коефіцієнту WOE значно покращує прогнозуючу здатність моделі.

Було проведено порівняльний аналіз методу логістичної регресії, реалізованого в SAB Analytical Studio, з методами дерев рішень та скорингової карти, побудованими в системі SAS Enterprise Miner. За критерієм загальної точності моделі найкращий результат виявився у методу дерев рішень, а індекс Gini виявився найкращим у скорингової карти. Логістична регресія в SAB Analytical Studio показала середні результати за обома критеріями, що вказує на її хорошу прогнозуючу здатність.

РОЗДІЛ 4 РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ

В останні роки набув великої популярності такий вид малого підприємництва як стартап. Стартап-проект – є комерційним проектом, який знаходиться в стані розробки, або нещодавно вийшов на ринок. Характерною особливістю стартапу, що відрізняє його від малого бізнесу, є оригінальність та інновації, він не може бути копією вже реалізованих ідей. При цьому проект не обов’язково повинен бути масштабного характеру, головне, щоб він був креативним, а його завдання – спрощувати людям будь-які дії в їх повсякденному житті.

Наразі, з появою Інтернету та сучасних технологій, стало простіше заходити на ринок, знаходити інвесторів та споживачів. З’явилося набагато більше можливостей для розвитку свого проекту за кордоном, ніж раніше. Проте розробка стартапу є досить ризикованим завданням. Не всім вдається довести свій стартап-проект до ринкового впровадження. За статистикою успіху досягає лише 10-20% від усіх стартап-проектів.

Запуск стартапу передбачає цілий ряд обов'язкових дій, в межах яких визначають ринкові перспективи стартапу, графік розробки, принципи організації виробництва, заходи з залучення інвесторів та аналіз ризиків.

4.1 Опис ідеї проекту

У таблиці 4.1 подано зміст ідеї стартап-проекту, можливі напрямки застосування та основні вигоди, що може отримати користувач товару. У таблиці 4.2 визначені сильні, слабкі та нейтральні сторони проекту.

Таблиця 4.1 – Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Програмний продукт для прогнозування кредитоспроможності фізичних осіб на основі застосування теорії регресійного аналізу	Банківські установи	Дозволяє користувачам з різним рівнем підготовки проводити необхідну попередню обробку даних для побудови прогнозуючої моделі, будувати скорингову модель та одержувати прогнозні дані на основі побудованої моделі

Таблиця 4.2 – Визначення сильних, слабких та нейтральних характеристик ідеї проекту

№ п/п	Техніко-економічні характеристики ідеї	(потенційні) товари/концепції конкурентів			
		Мій проект	Deductor Credit Scorecard Modeler	IBM SPSS Modeler	SAS Enterprise Miner
1.	Ціна	Низька	Середня	Висока	Висока
2.	Функціонал	Вузький	Вузький	Широкий	Широкий

Отже, з табл. 4.2 можна визначити, що ціна є сильною характеристикою для потенційного товару, а функціонал, зважаючи на напрямки застосування товару, є нейтральною властивістю.

4.2 Технологічний аудит ідеї проекту

За результатами аналізу таблиці 4.3 можна зробити висновок про можливість технологічної реалізації проекту.

Таблиця 4.3 – Технологічна здійсненність ідеї проекту

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1	Програмний продукт для прогнозування кредитоспроможності фізичних осіб на основі застосування теорії регресійного аналізу	Прогнозування на основі методу лінійної регресії	Наявна	Доступна
2		Прогнозування на основі методу логістичної регресії	Наявна	Доступна
Обрана технологія реалізації ідеї проекту: прогнозування на основі методу логістичної регресії				

4.3 Аналіз ринкових можливостей запуску стартап-проекту

Визначення ринкових можливостей, які можна використати під час ринкового впровадження проекту, та ринкових загроз, які можуть перешкодити реалізації проекту, дозволяє спланувати напрями розвитку проекту із урахуванням стану ринкового середовища, потреб потенційних клієнтів та пропозицій проектів-конкурентів.

Проведемо аналіз попиту: наявність попиту, обсяг, динаміка розвитку ринку (табл. 4.4).

Таблиця 4.4 – Попередня характеристика потенційного ринку стартапу

№ п/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	3
2	Загальний обсяг продаж, грн/ум.од	100 000 ум.од
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Немає
5	Специфічні вимоги до стандартизації та сертифікації	Немає
6	Середня норма рентабельності в галузі (або по ринку), %	75%

За результатами аналізу таблиці 4.4 можна зробити висновок, що ринок є привабливим для входження за попереднім оцінюванням.

Визначимо потенційні групи клієнтів, їх характеристики, та сформуємо орієнтовний перелік вимог до товару для кожної групи (табл. 4.5).

Таблиця 4.5 – Характеристика потенційних клієнтів стартап-проекту

№ п/п	Потреба, що формує ринок	Цільова аудиторія	Відмінності у поведінці груп клієнтів	Вимоги споживачів
1	Прийняття рішення щодо видачі кредитів фізичним особам	Банківські установи	Відмінність сфер діяльності клієнтів (кредитування фізичних осіб, юридичних осіб)	Висока точність прогнозування. Простий у використанні. Швидкодія при обробці значного об’єму інформації

Проведемо аналіз ринкового середовища: таблиці факторів, що сприяють ринковому впровадженню проекту, та факторів, що йому перешкоджають (табл. №№ 4.6-4.7).

Таблиця 4.6 – Фактори загроз

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1	Наявність великої конкуренції	Вихід на ринок великої компанії	Вихід з ринку. Обрати нову цільову аудиторію. Передбачити переваги продукту, щоб повідомити про них саме після виходу великої компанії на ринок
2	Зміна потреб користувачів	Користувачам необхідні рішення з іншим функціоналом	Передбачити можливість додавання нового функціоналу до продукту

Таблиця 4.7 – Фактори можливостей

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1	Відсутність конкуренції	Відсутність аналогічних продуктів для користувача на вітчизняному ринку	Локалізація та адаптація сервісу для локальних груп. Адаптація до вітчизняних особливостей
2	Поява нових цільових груп клієнтів	Потреба в аналогічному продукті в інших сферах діяльності	Адаптація продукту під нові сфери використання

Проведемо аналіз пропозиції: визначимо загальні риси конкуренції на ринку (табл. 4.8).

Таблиця 4.8 – Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1. Вказати тип конкуренції - монополістична	Існує декілька фірм-конкурентів	Підтримка якості продукту та постійні вдосконалення
2. За рівнем конкурентної боротьби - інтернаціональний	Фірми конкуренти з різних країн	Підтримувати продукт на національному ринку
3. За галузевою ознакою - внутрішньогалузева	Продукт використовується в одній галузі	Вдосконалювати продукт для застосування в інших галузях
4. Конкуренція за видами товарів: - товарно-родова	Присутня конкуренція з боку товарів-замінників	Розширювати функціонал продукту
5.За характером конкурентних переваг - нецінова	Вдосконалення якості продукції, технології виробництва, інновацій	Випускати нові товари, які принципово відрізняються від своїх попередників та представляють модернізований варіант старої моделі
6. За інтенсивністю - немарочна	Роль торгової марки незначна	Приділяти увагу якості продукту а не бренду компанії

Після аналізу конкуренції проведемо більш детальний аналіз умов конкуренції в галузі (за моделлю 5 сил М. Портера) (табл. 4.9).

Таблиця 4.9 – Аналіз конкуренції в галузі за М. Портером

	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
Складові аналізу	IBM SPSS Modeler, Deductor Credit Scorecard Modeler	SAS Enterprise Miner	Диференціація витрат, розширення каналів збуту	Контроль якості продукту	Наявність більш широкого функціоналу, зручнішого інтерфейсу
Висновки:	Середня конкурентна боротьба з вже існуючими на ринку гравцями	Є можливості виходу на ринок, але є і конкуренти. Строки – пів року.	Постачальники не диктують умови роботи	Клієнти диктують умови роботи на ринку	Обмеження для роботи на ринку через товари замінники

На основі аналізу конкуренції (табл. 4.9), а також із урахуванням характеристик ідеї проекту (табл. 4.2), вимог споживачів до товару (табл. 4.5) та факторів маркетингового середовища (табл. №№ 4.6-4.7) визначимо та обґрунтуємо перелік факторів конкурентоспроможності (табл. 4.10).

Таблиця 4.10 – Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Ціна	Більш доступна ціна збільшує кількість потенційних клієнтів
2	Функціонал	Функціонал направлений на предметну область
3	Зручний інтерфейс	Зручний інтерфейс робить продукт більш привабливим для клієнтів

За визначеними факторами конкурентоспроможності (табл. 4.10) проведемо аналіз сильних та слабких сторін стартап-проекту (табл. 4.11).

Таблиця 4.11 – Порівняльний аналіз сильних та слабких сторін «SAB Analytical Studio»

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з “SAB Analytical Studio”						
			–3	–2	–1	0	+1	+2	+3
1	Ціна	18		+					
2	Функціонал	10					+		
3	Зручний інтерфейс	12				+			

Складемо SWOT-аналіз (матриця аналізу сильних (Strength) та слабких (Weak) сторін, загроз (Troubles) та можливостей (Opportunities)) (табл. 4.12) на основі виділених ринкових загроз та можливостей, та сильних і слабких сторін (табл. 4.11).

Таблиця 4.12 – SWOT-аналіз стартап-проекту

Сильні сторони: ціна, зручний інтерфейс	Слабкі сторони: функціонал
Можливості: Низька конкуренція, поява нових потреб споживачів	Загрози: Висока конкуренція, не відповідність потребам споживачів

На основі SWOT-аналізу визначимо альтернативи ринкової поведінки (перелік заходів) для виведення стартап-проекту на ринок та орієнтовний оптимальний час їх ринкової реалізації з огляду на потенційні проекти конкурентів, що можуть бути виведені на ринок (табл. 4.13).

Таблиця 4.13 – Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Створення програмного забезпечення	80%	3 місяців
2	Створення веб-сервісу	60%	5 місяців

4.4 Розроблення ринкової стратегії проекту

Розроблення ринкової стратегії першим кроком передбачає визначення стратегії охоплення ринку: опис цільових груп потенційних споживачів (табл. 4.14).

Таблиця 4.14 – Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Банки	Висока	Високий	Середня	Середня складність
2	Інші фінансові установи	Середня	Середній	Помірна	Висока складність
Які цільові групи обрано: 1					

Для роботи в обраних сегментах ринку сформуємо базову стратегію розвитку (табл. 4.15).

Таблиця 4.15 – Визначення базової стратегії розвитку

№ п/п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції	Базова стратегія розвитку*
1	Надання товару важливих з точки зору споживача відмітних властивостей, які роблять товар відмінним від товарів конкурентів	Визначити потреби кожної з цільових груп, розробити стратегії приваблення споживачів та маркетингові комунікації	Оперативне реагування на зміни в ринковому попиті, орієнтованість на кінцевого споживача, висока якість продукту	Стратегія диференціації

Оберемо стратегію конкурентної поведінки (табл. 4.16).

Таблиця 4.16 – Визначення базової стратегії конкурентної поведінки

№ п/п	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки*
1	Не є першопрохідцем	Шукати нових	Ні	Стратегія заняття конкурентної ніші

Сформуємо ринкову позицію, за якою споживачі мають ідентифікувати проект(табл. 4.17).

Таблиця 4.17 – Визначення стратегії позиціонування

№ п/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкуренто-спроможні позиції власного проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту
1	Простий та зручний користувацький інтерфейс, надійність та безпека, швидкість роботи продукти	Стратегія диференціації	Позиція на основі порівняння продукту компанії з продуктами конкурентів. Відмінні особливості споживачів	Автоматизація робочих процесів, зниження кредитних ризиків, зниження навантаження та часу

4.5 Розроблення маркетингової програми стартап-проекту

У табл. 4.18 підсумуємо результати попереднього аналізу конкурентоспроможності товару.

Таблиця 4.18 – Визначення ключових переваг концепції потенційного товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами
1	Автоматизація робочих процесів	Продукт автоматизує такі процеси, як обробка даних та прийняття рішення щодо видачі кредиту	Після впровадження продукту процес прийняття рішення щодо видачі кредиту стає автоматизований
2	Зменшення кредитних ризиків	Продукт зменшує кредитні ризики банківських установ	Висока точність прогнозування знижує кредитні ризики банківських установ
3	Зниження навантаження та часу	Продукт знижує навантаження на персонал банківських установ та зменшує час видачі кредиту	Персоналу банків не потрібно самостійно аналізувати великий об’єм даних, що знижує навантаження на прискорює роботу

Розроблена трирівнева маркетингова модель товару(табл. 4.19).

Таблиця 4.19 – Опис трьох рівнів моделі товару

<table>
<tr><td>Рівні товару</td><td colspan="3">Сутність та складові</td></tr>
<tr><td>I. Товар за задумом</td><td colspan="3">Програмний продукт для прогнозування кредитоспроможності фізичних осіб. Повинен бути зручним, швидким та безпечним</td></tr>
<tr><td rowspan="5">II.
Товар у реальному виконанні</td><td>Властивості/характеристики</td><td>М/Нм</td><td>Вр/Тх /Тл/Е/Ор</td></tr>
<tr><td>1. Попередня обробка даних
2. Побудова скорингової моделі
3. Прогнозування кредитоспроможності</td><td></td><td></td></tr>
<tr><td colspan="3">Якість: проходження тестування</td></tr>
<tr><td colspan="3">Пакування: відсутнє</td></tr>
<tr><td colspan="3">Марка: “SAB Analytical Studio”</td></tr>
<tr><td rowspan="2">III. Товар із підкріпленням</td><td colspan="3">До продажу: відсутнє</td></tr>
<tr><td colspan="3">Після продажу: навчання персоналу, супровід, технічна підтримка</td></tr>
<tr><td colspan="4">Вихідний код програмного продукту є закритим, та не передається клієнтам і третім особам. На програмний продукт оформлено авторське право</td></tr>
</table>

Визначимо цінові межі, якими необхідно керуватись при встановленні ціни на товар (табл. 4.20).

Таблиця 4.20 – Визначення меж встановлення ціни

№ п/п	Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	2500$	2000$	Високий рівень доходів	Базова покупка та впровадження: нижня межа - 1000$, верхня межа - 2000$.

Визначимо оптимальну систему збуту (табл. 4.21).

Таблиця 4.21 – Формування системи збуту

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Цільові клієнти – банківські установи, які бажають впровадити у своїй роботі сучасні засоби, які допоможуть автоматизувати робочі процеси. Вони цікавляться інноваційними рішеннями, відвідують тематичні семінари та конференції	Формування попиту і стимулювання збуту. Встановлення контактів із споживачами. Просування маркетингової інформації	Нульова або однорівнева (сервіс безпосередньо продається споживачам та через посередників)	Прямий канал збуту до споживача, мінімізувати витрати на додаткові канали збуту

Розроблена концепція маркетингових комунікацій, що спирається на попередньо обрану основу для позиціонування, визначену специфіку поведінки клієнтів (табл. 4.22).

Таблиця 4.22 – Концепція маркетингових комунікацій

№ п/п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1	Цільові клієнти – банківські установи, що займаються кредитуванням фізичних осіб, і бажають автоматизувати процес видачі кредиту, та зменшити кількість неповернутих кредитів. Вони цікавляться інноваційними рішеннями, відвідують тематичні семінари та конференції	Конференції, форуми, новини у сфері інноваційних технологій, періодичні видання у професійних галузях	Позиція на основі порівняння продукту компанії з продуктами конкурентів. Відмінні особливості споживачів	- інформувати про новий продукт та його переваги; - сформувати сприятливу думку; - сформувати образ марки та її виробника у свідомості споживачів; - збільшити потік покупців	Зменшуємо кредитні ризики. Прискорюємо та автоматизуємо процес видачі кредитів

Висновки до розділу 4

В даному розділі проведено аналіз створення та виведення на ринок стартап-проекту на основі програмного продукту, який було розроблено в рамках магістерської дисертації.

В межах цього аналізу було розроблено опис самої ідеї проекту, визначено загальні напрями використання товару, проаналізовано ринкові можливості щодо впровадження проекту, визначено відмінності від конкурентів та розроблено стратегію виходу на ринок.

Узагальнюючи проведений аналіз, можна зазначити, що є можливість ринкової комерціалізації проекту. Наявний попит, динаміка ринку зростає. З огляду на потенційні групи клієнтів, а саме банківські установи, та високий рівень конкурентоспроможності проекту, є достатні перспективи для впровадження стартапу. Отже, подальша імплементація проекту є доцільною.

ВИСНОВКИ

Дана робота присвячена аналізу, побудові та використанню прогнозуючих моделей для оцінювання кредитоспроможності фізичних осіб і ризику банків при наданні кредитів.

Після ознайомлення з теоретичним матеріалом щодо понять кредитоспроможності та кредитного скорингу, основними статистичними та математичними методами побудови скорингових моделей для прогнозування ймовірності повернення кредитів, було побудовано СППР для прийняття рішень щодо надання споживчих кредитів фінансовими установами.

В якості практичного прикладу застосування СППР, було розроблено програмний продукт SAB Analytical Studio з використанням технологій .Net у середовищі розробки Microsoft Visual Studio 2012.У даній системі було реалізовано метод логістичної регресії для прогнозування кредитоспроможності позичальників. Для знаходження оцінок параметрів регресійної моделі було використано метод максимальної правдоподібності з використанням методу градієнтного спуску.

Проведено порівняльний аналіз отриманої моделі логістичної регресії в програмному продукті SAB Analytical Studio з побудованими скоринговими моделями в системі SAS Enterprise Minerна основі методів дерев рішень та логістичної регресії.

Отримані результати порівняння статистичних характеристик якості побудованих прогнозуючих моделей показали, що метод логістичної регресії, реалізований у розробленому програмному продукті, дає не гірші результати ніж інші методи, реалізовані у комерційних аналогах.

Результати магістерської дисертації:

- запропоновано архітектуру систему підтримки прийняття рішень для оцінювання кредитоспроможності фізичних осіб;

- розроблено програмний продукт для аналізу та обробки даних, побудови скорингової моделі на основі логістичної регресії та прогнозування кредитоспроможності;
- розроблений ПП апробовано на вибірці з 3000 клієнтів німецького банку;
- реалізовано метод максимальної правдоподібності з використанням методу градієнтного спуску;
- запропоновано спосіб перекодування категоріальних змінних в інтервальні;
- виконано порівняльний аналіз з іншими методами прогнозування, реалізованими в комерційних системах.

Подальшими напрямками роботи можуть бути питання, що стосуються:

- вдосконалення розробленого методу побудови скорингової моделі;
- розробки нових підходів щодо визначення ступеня значимості регресорів;
- реалізації методів інтелектуального аналізу даних іншого типу, наприклад, нейронних мереж, методу групового врахування аргументів, методу опорних векторів.

Розроблений програмний продукт показав прийнятні результати, що підтверджує раціональність використання обраного методу.

ПЕРЕЛІК ПОСИЛАНЬ

1. Bielecki T. R. Credit Risk: Modeling, Valuation, Hedging / T. R. Bielecki, M. Rutkowsky. – Berlin: Springer, 2002. – 500 p.
2. Van Gruening H. Analyzing and Managing Banking Risks / H. Van Gruening, S. B. Bratanovic. – Washington: The World Bank, 2003. – 386 p.
3. Aven T. Foundations of Risk Analysis: A Knowledge and Decision-Oriented Perspective / T. Aven. – New York: John Wiley & Sons, 2003. – 198 p.
4. Муравйова М. Ю. Шляхи вдосконалення оцінки кредитоспроможності позичальників банками України [Електронний ресурс] / М. Ю. Муравйова // Управління розвитком. – 2012. – №14 (135). – Режим доступу:http://www.nbuv.gov.ua/old_jrn/Soc_Gum/Uproz/2012_14/u1214mur.pdf.
5. Вдовенко Л. О. Економічна сутність та значення кредитоспроможності підприємств [Електронний ресурс] / Л. О. Вдовенко // Облік і фінанси. – 2012. – № 1. – С. 108-111. – Режим доступу: http://nbuv.gov.ua/UJRN/Oif_apk_2012_1_23.
6. Бучко І. Є.Скоринг як метод зниження кредитного ризику банку [Електронний ресурс] / І. Є. Бучко // Вісник Університету банківської справи Національного банку України. – 2013. – № 2. – Режим доступу:http://nbuv.gov.ua/UJRN/VUbsNbU_2013_2_37.
7. Клейнер Г. Б. История современного кредитного скоринга [Электронный ресурс] / Г. Б. Клейнер, Д. С. Коробов // Проблемы региональной экономики: электронный журнал. – 2012. – Вып. 17. – С. 45-49. Режим доступа: http://www.regec.ru/articles/2012/vol1/5.pdf.

8. Ишина И. В. Скоринг - модель оценки кредитного риска [Электронный ресурс] / И. В. Ишина, М. Н. Сазонова // Аудит и финансовый анализ. – 2007. – № 4. – Режим доступа: http://www.auditfin.com/fin/2007/4/Ishina/Ishina%20.pdf.
9. Самойлова С. С. Скоринговые модели оценки кредитного риска [Электронный ресурс] / С. С. Самойлова, М. А. Курочка // Социально-экономические явления и процессы. – 2014. – № 3 (61). – Режим доступа: http://cyberleninka.ru/article/n/skoringovye-modeli-otsenki-kreditnogo-riska.
10. Лункіна Т. І. Використання скоринг моделі при управлінні ризиками споживчого кредитування [Електронний ресурс] / Т. І. Лугкіна // Ефективна економіка. – 2015. – № 2. – Режим доступу: http://www.economy.nayka.com.ua/?op=1&z=3792.
11. Терентьев А. Н. SAS BASE: Основы программирования / А. Н. Терентьев, В. Н. Домрачев, Р. И. Костецкий – К.: Эдельвейс, 2014. – 304 с.
12. Anderson B. S. Developing Credit Scorecards Using SAS Credit Scoring for Enterprise Miner 5.3 / B. S. Anderson, R. W. Thompson. – Cary: SAS Institute Inc, 2009. – 41 p.
13. Кожухівська О. А. Прогнозування ризиків кредитування фізичних осіб за математичними моделями [Електронний ресурс] / О. А. Кожухівська // Вісник Національного університету «Львівська політехніка». Інформаційні системи та мережі. – 2013. – № 770. – С. 177-185. – Режим доступу:http://nbuv.gov.ua/UJRN/VNULPICM_2013_770_23.
14. Романенко А. В. Логистическая регрессия // Молодёжь и наука: Сборник материалов VIII Всероссийской научно-технической конференции студентов, аспирантов и молодых учёных, посвященной 155-летию со дня рождения К. Э. Циолковского [Электронный ресурс]. – Красноярск: Сибирский федеральный ун-т, 2012.– Режим доступа: http://conf.sfu-kras.ru/sites/mn2012/section21.html

15. Калиткин Н. Н. Численные методы / Н.Н. Калиткин. – М.: Наука, 1978. – 592 с.
16. Сиддики Н. Скоринговые карты для оценки кредитных рисков. Разработка и внедрение интеллектуальных методов кредитного скоринга / Н. Сиддики; пер. с англ. Е. Ильичева. – М.: Манн, Иванов и Фербер, 2014. – 268 с.
17. Сорокин А. С. Построение скоринговых карт с использованием модели логистической регрессии [Электронный ресурс] / А. С. Сорокин // Науковедение. – 2014. – Вып. 2 (21). – Режим доступа: http://naukovedenie.ru/PDF/180EVN214.pdf.
18. Кузнєцова Н. В. Порівняльний аналіз характеристик моделей оцінювання ризиків кредитування / Н. В. Кузнєцова, П. І. Бідюк // Наукові вісті НТУУ «КПІ». – 2010. – №1. – С. 42-53.
19. Сорокин А.С. К вопросу валидации модели логистической регрессии в кредитном скоринге [Электронный ресурс] / А. С. Сорокин // Науковедение. – 2014. – Вып. 2 (21). – Режим доступа: http://naukovedenie.ru/PDF/ 173EVN214.pdf.

ДОДАТОК А ІЛЮСТРАТИВНИЙ МАТЕРІАЛ ДОПОВІДІ

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

ДОДАТОК Б ТАБЛИЦЯ СТАТИСТИЧНИХ ДАНИХ

У таблицях Б.1 та Б.2 наведено перші 15 та останні 15 значень вибірки клієнтів німецького банку, по яких проводився порівняльний аналіз методів побудови скорингових моделей.

Таблиця Б.1 – Вихідні статистичні дані щодо перших 9 показників

TITLE	CHILDREN	AGE	NMBLOAN	FINLOAN	INCOME	EC_CARD	STATUS	LOANS
R	0	46	0	0	0	1	V	0
H	4	34	1	1	3200	0	V	2
H	3	31	1	1	3300	0	V	3
R	0	39	0	0	1500	0	W	1
H	3	32	2	1	0	1	V	1
H	0	23	2	1	0	1	U	1
R	0	42	0	0	1900	0	V	0
H	2	35	0	0	0	1	V	2
H	1	26	0	0	1700	1	V	0
H	1	24	2	1	3400	0	U	2
R	0	20	0	0	0	1	U	0
H	1	44	0	1	0	1	G	2
H	0	23	0	0	1900	0	U	1
H	1	44	0	0	0	1	V	0
H	6	36	2	1	3500	0	V	0
……								
H	1	31	2	1	3200	1	V	6
H	0	56	2	1	2200	0	U	0
R	0	42	2	1	1600	0	U	1
R	0	42	2	1	1500	0	U	1
H	2	35	2	1	3500	0	V	3
R	2	30	2	1	0	1	G	1
H	0	21	2	0	2700	0	U	1
H	0	57	2	1	2000	0	V	2
H	0	29	2	1	2600	0	V	2
H	3	33	1	1	3200	0	V	2
H	0	26	2	1	2300	0	U	2
H	0	34	2	0	2500	0	U	1
H	0	32	2	1	2800	0	G	3
R	0	31	2	1	2100	0	G	1
R	0	27	2	1	2200	0	V	1

Таблиця Б.2 – Вихідні статистичні дані щодо останніх 8 показників

REGN	CASH	PRODUCT	RESID	NAT	CAR	CARDS	GB
0	2000	Radio, TV, Hifi	Lease	German	Car	Cheque card	0
4	6000	None	Owner	Turkish	Car	no credit cards	1
4	0	None	Lease	Turkish	Car	no credit cards	1
0	2500	Furniture,Carpet	Lease	German	Without Vehicle	no credit cards	1
0	2500	Furniture,Carpet	Lease	German	Car	Cheque card	0
0	700	Furniture,Carpet	Lease	German	Car	Cheque card	1
0	9000	Furniture,Carpet	Lease	German	Car	no credit cards	0
0	1700	Furniture,Carpet	Lease	German	Car	Cheque card	0
0	800	Radio, TV, Hifi	Lease	German	Car	Cheque card	1
0	1000	Radio, TV, Hifi	Lease	German	Car	no credit cards	1
0	4000	Furniture,Carpet	Lease	German	Without Vehicle	Cheque card	0
0	800	Radio, TV, Hifi	Lease	German	Car	Cheque card	0
0	600	Dept. Store,Mail	Lease	German	Car	no credit cards	1
0	6000	Furniture,Carpet	Lease	German	Car	Cheque card	1
0	1000	Dept. Store,Mail	Lease	German	Car	no credit cards	0
……							
6	600	Radio, TV, Hifi	Lease	German	Car	Cheque card	0
4	3000	Radio, TV, Hifi	Lease	German	Without Vehicle	no credit cards	1
3	700	Dept. Store,Mail	Lease	German	Without Vehicle	no credit cards	1
3	1800	Radio, TV, Hifi	Lease	German	Without Vehicle	no credit cards	1
4	1200	Dept. Store,Mail	Lease	German	Without Vehicle	no credit cards	0
9	1100	Furniture,Carpet	Lease	German	Car	Cheque card	1
8	700	Radio, TV, Hifi	Lease	German	Car	no credit cards	1
4	600	Radio, TV, Hifi	Lease	German	Car	no credit cards	0
5	1000	Radio, TV, Hifi	Lease	German	Car	no credit cards	0
5	2500	Radio, TV, Hifi	Lease	German	Car	no credit cards	1
2	5000	Radio, TV, Hifi	Lease	German	Without Vehicle	no credit cards	1
5	3000	Furniture,Carpet	Lease	German	Car	no credit cards	0
5	2500	Radio, TV, Hifi	Lease	German	Car	no credit cards	1
4	4000	Furniture,Carpet	Lease	German	Car	no credit cards	0
4	2500	Radio, TV, Hifi	Lease	German	Without Vehicle	no credit cards	1

ДОДАТОК В ЛІСТИНГ ПРОГРАМИ

В.1 Програмний код методу максимальної правдоподібності

```
class MMP
{
      public static double f(double[] x, double[] Teta)
      {
         var result = (double)(1 / (1 + Math.Exp(-1 * Matrix.MultT1(Teta, x))));
         return result;
      }

      public static double[] grad(double[,] X, double[] y, double[] Teta)
      {
         double[] sum = new double[Teta.Length];
         for (int i = 0; i < sum.Length; ++i)
         {
            sum[i] = 0;
         }

         for (int i = 0; i < y.Length; ++i)
         {
            double[] x = Matrix.GetRow(X, i);
            sum = Matrix.Sum(sum, Matrix.MultNum(x, (y[i] - f(x, Teta))));
         }
         return sum;
      }

      public static double Error(double[,] X, double[] y, double[] Teta)
      {
         double sumSquaredError = 0;
         for (int i = 0; i < y.Length; ++i) // each data
         {
            double[] x = Matrix.GetRow(X, i);
            double computed = f(x, Teta);
            double desired = y[i];
            sumSquaredError += (computed - desired) * (computed - desired);
         }
         return sumSquaredError / y.Length;
      }

 }
```

В.2 Програмний код матричних процедур

```
class Matrix
{
    public static double[,] Mult(double[,] matr_1, double[,] matr_2)
    {
        if (matr_1.GetLength(1) != matr_2.GetLength(0)) throw new Exception(“Error”);
        double[,] result = new double[matr_1.GetLength(0), matr_2.GetLength(1)]; // 0-rows , 1-column
        for (int i = 0; i < matr_1.GetLength(0); i++)
        {
            for (int j = 0; j < matr_2.GetLength(1); j++)
            {
                for (int k = 0; k < matr_2.GetLength(0); k++)
                    result[i, j] += matr_1[i, k] * matr_2[k, j];
            }
        }
        return result;
    }

    public static double[,] MultT(double[,] Tmatr_1, double[,] Tmatr_2)
    {
        if (Tmatr_1.GetLength(0) != Tmatr_2.GetLength(1)) throw new Exception(“Error”);
        double[,] matr_2 = Tran(Tmatr_2);
        double[,] matr_1 = Tran(Tmatr_1);
        double[,] result = new double[matr_1.GetLength(0), matr_2.GetLength(1)]; // 0-rows , 1-column
        for (int i = 0; i < matr_1.GetLength(0); i++)
        {
            for (int j = 0; j < matr_2.GetLength(1); j++)
            {
                for (int k = 0; k < matr_2.GetLength(0); k++)
                    result[i, j] += matr_1[i, k] * matr_2[k, j];
            }
        }
        return result;
    }

    public static double[] Mult(double[,] matr, double[] vec)
    {
        if (matr.GetLength(1) != vec.Length) throw new Exception("Матрицы нельзя перемножить");
        double[] result = new double[matr.GetLength(0)];
        for (int i = 0; i < matr.GetLength(0); i++)
        {
            for (int j = 0; j < vec.Length; j++)
                result[i] += matr[i, j] * vec[j];
        }
        return result;
    }

    public static double[,] Tran(double[,] matr)
    {
        double[,] result = new double[matr.GetLength(1), matr.GetLength(0)];
        for (int i = 0; i < matr.GetLength(0); i++)
            for (int j = 0; j < matr.GetLength(1); j++)
                result[j, i] = matr[i, j];
        return result;
    }

    public static double[] GetColumn(double[,] matr, int index)
    {
        if (matr.GetLength(1) <= index) throw new Exception("Error index > demension");
```

```
        double[] result = new double[matr.GetLength(0)];
        for (int i = 0; i < matr.GetLength(0); i++)
            result[i] = matr[i, index];
        return result;
    }

    public static double[] GetRow(double[,] matr, int index)
    {
        if (matr.GetLength(0) <= index) throw new Exception("Error index > demension");
        double[] result = new double[matr.GetLength(1)];
        for (int i = 0; i < matr.GetLength(1); i++)
            result[i] = matr[index, i];
        return result;
    }

    public static double[,] MultNum(double[,] matr, double number)
    {
        double[,] result = (double[,])matr.Clone();
        for (int i = 0; i < result.GetLength(0); i++)
            for (int j = 0; j < result.GetLength(1); j++)
                result[i, j] *= number;
        return result;
    }

    public static double[] MultNum(double[] vec, double number)
    {
        double[] result = (double[])vec.Clone();
        for (int i = 0; i < result.GetLength(0); i++)
            result[i] *= number;
        return result;
    }

    public static double[,] Sum(double[,] matr_1, double[,] matr_2)
    {
        if ((matr_1.GetLength(0) != matr_2.GetLength(0)) || (matr_1.GetLength(1) != matr_2.GetLength(1)))
throw new Exception("Error");
        double[,] result = (double[,])matr_1.Clone();
        for (int i = 0; i < result.GetLength(0); i++)
            for (int j = 0; j < result.GetLength(1); j++)
                result[i, j] += matr_2[i, j];
        return result;
    }

    public static double[] Sum(double[] vec_1, double[] vec_2)
    {
        if (vec_1.Length != vec_2.Length) throw new Exception("Error");
        double[] result = (double[])vec_1.Clone();
        for (int i = 0; i < result.Length; i++)
            result[i] += vec_2[i];
        return result;
    }

    public static double[] Sum(double[] vec, double num)
    {
        double[] result = (double[])vec.Clone();
        for (int i = 0; i < result.Length; i++)
            result[i] += num;
        return result;
    }
```

В.3 Програмний код дискретизації змінних

```
public partial class Discret
{
   private double[,] Uniform_count(double[] X, int n)
   {
      double[,] Categ_Uniform = new double[n, 2];
      int N = X.Length;
      double[] X_sort = new double[N];
      for (int i = 0; i < N; i++)
         X_sort[i] = X[i];
      Array.Sort(X_sort);

      int perc = N / n;
      int ost = N - perc * n;
      int k = 0;
      Categ_Uniform[0, 0] = Double.MinValue;
      Categ_Uniform[n - 1, 1] = Double.MaxValue;
      for (int i = 0; i < n; i++)
      {
         int num = 1;
         if (i != 0)
            Categ_Uniform[i, 0] = X_sort[k];
         if (i != n - 1)
         {
            Categ_Uniform[i, 1] = X_sort[k + 1];
            while (num < perc)
            {
               num++;
               k++;
               Categ_Uniform[i, 1] = X_sort[k + 1];
            }
            k++;
            if (ost > 0)
            {
               Categ_Uniform[i, 1] = X_sort[k + 1];
               k++;
               ost--;
            }
            while (X_sort[k - 1] == X_sort[k])
            {
               Categ_Uniform[i, 1] = X_sort[k + 1];
               k++;
            }
            perc = (N - k) / (n - (i + 1));
            ost = (N - k) - perc * (n - (i + 1));
         }
      }
      return Categ_Uniform;
   }

   private double[,] Uniform_width(double[] X, int n)
   {
      double[,] Categ_Uniform = new double[n, 2];
      double perc = (X.Max() - X.Min()) / n;
      double temp = X.Min() + perc;
      for (int i = 0; i < n; i++)
      {
         if ((i != 0) || (i != n - 1))
         {
```

```
            Categ_Uniform[i, 0] = temp - perc;
            Categ_Uniform[i, 1] = temp;
        }
        else
        {
            if (i == 0)
                Categ_Uniform[0, 0] = Double.MinValue;
            else
                Categ_Uniform[n - 1, 1] = Double.MaxValue;
        }
        temp += perc;
    }
    return Categ_Uniform;
  }
}
```

В.4 Програмний код перетворення категоріальних змінних в числові

```
 private double[] ConvertWOE(List<string> list, double[] Y)
 {
    double[] Cat = new double[list.Count];
    string[] Ar = new string[list.Count];
    for (int i = 0; i < list.Count(); i++)
       Ar[i] = list[i];
    list.Sort();
    list = list.Distinct().ToList();
    double[] woe = new double[list.Count];
    double p;
    for (int i = 0; i < list.Count; i++)
    {
       int n = 0, n_y = 0;
       for (int j = 0; j < Ar.Length; j++)
       {
          if (Ar[j] == list[i])
          {
             if (Y[j] == 0)
                n_y++;
             n++;
          }
       }
       p = (double)n_y / n;
       woe[i] = Math.Log((double)p / (1 - p));
    }
    quickSort(list, woe, 0, woe.Length - 1);
    for (int i = 0; i < Ar.Length; i++)
    {
       for (int j = 0; j < list.Count; j++)
          if (Ar[i] == list[j])
             Cat[i] = j+1;
    }
    return Cat;
 }

 private double[] ConvertCat(List<string> list)
 {
    double[] Cat = new double[list.Count];
    string[] Ar = new string[list.Count];
    for (int i = 0; i < list.Count(); i++)
       Ar[i] = list[i];
```

```
        list.Sort();
        list = list.Distinct().ToList();
        for (int i = 0; i < Ar.Length; i++)
        {
            for (int j = 0; j < list.Count; j++)
                if (Ar[i] == list[j])
                    Cat[i] = j + 1;
        }
        return Cat;
}
```

В.5 Програмний код розрахунку статистичних коефіцієнтів якості

```
    public double CA(double[,] X, double[] Y, double[] Teta)
    {
        int n = 0;
        double[] res = Matrix.Mult(X, Teta);
        for (int i = 0; i < res.Length; i++)
        {
            if (res[i] >= 0.5) res[i] = 1;
            elseres[i] = 0;
            if (res[i] == Y[i]) n++;
        }
        return (double)n / (double)res.Length;
    }

    public double Gini(double[,] X, double[] Y, double[] Teta, ref double[] Se, ref double[] Sp)
    {
        double[] cutoff = new double[100];
        double[] res = Matrix.Mult(X, Teta);
        double step = (double)(res.Max() - res.Min()) / (double)100;
        double temp = 0;
        List<string> str = new List<string>();
        for (int i = 0; i < cutoff.Length; i++)
        {
            cutoff[i] = (double)res.Min() + (double)temp;
            temp += step;
        }
        for (int k = 0; k < cutoff.Length; k++)
        {
            int TP = 0;  int TN = 0; int FP = 0; int FN = 0; int res_temp = 0;
            for (int i = 0; i < res.Length; i++)
            {
                if (res[i] >= cutoff[k])res_temp = 1;
                elseres_temp = 0;
                if (res_temp == Y[i])
                {
                    if (Y[i] == 0) TP++;
                    elseTN++;
                }
                else
                {
                    if (Y[i] == 0) FN++;
                    elseFP++;
                }
            }
            Se[k] = Math.Round((double)TP / (double)((double)TP + (double)FN), 4);
            Sp[k] = Math.Round(1 - (double)TN / (double)((double)TN + (double)FP), 4);
        }
```

```
}
double Square(double[] y, double[] x)
{
    double res = 0;
    for (int i = 0; i < y.Length - 1; ++i)
        res += (double) (y[i] + y[i + 1]) * (x[i + 1] - x[i]) / (double) 2;
    return res;
}
```

В.6 Програмний код розрахунку коефіцієнту інформаційного значення

```
double IV(List<string> list, double[] Y)
{
    double iv = 0;
    string[] Ar = new string[list.Count];
    for (int i = 0; i < list.Count(); i++)
        Ar[i] = list[i];
    list.Sort();
    list = list.Distinct().ToList();
    for (int i = 0; i < list.Count; i++)
    {
        int n = 0, n_y = 0;
        double woe = 0;
        double p = 0;
        for (int j = 0; j < Ar.Length; j++)
        {
            if (Ar[j] == list[i])
            {
                if (Y[j] == 0)
                    n_y++;
                n++;
            }
        }
        p = (double)n_y / n;
        woe = Math.Log((double)p / (1 - p));
        if (double.IsInfinity(woe))
            woe = 0;
        iv += (p - (1 - p)) * woe;
    }
    return iv;
}
```

ДОДАТОК Г АВТОРСЬКЕ ПРАВО НА ТВІР

МІНІСТЕРСТВО ЕКОНОМІЧНОГО РОЗВИТКУ І ТОРГІВЛІ УКРАЇНИ
(Мінекономрозвитку України)

вул. М. Грушевського, 12/2, м. Київ, 01008, тел. 523-93-94, факс 226-31-81
Web: http://www.me.gov.ua , e-mail: meconomy@me.gov.ua , код ЄДРПОУ 37508596

РІШЕННЯ

ПРО РЕЄСТРАЦІЮ АВТОРСЬКОГО ПРАВА НА ТВІР

Міністерство економічного розвитку і торгівлі України розглянуло заяву

Бакун Сабіна Антонівна, вул. Софії Русової, 3-А, кв. 124, м. Київ, 02140
(повне ім'я автора, адреса)

заявка від **02.03.2018** **№ 78478**

про реєстрацію авторського права на твір і прийняло рішення зареєструвати авторське право на твір **Комп'ютерна програма "SAB Analytical Studio"; Бакун Сабіна Антонівна**
(вид, повна, скорочена (за наявності) назва твору, повне ім'я, псевдонім (за наявності) автора (ів))

Внесення відомостей до Державного реєстру свідоцтв про реєстрацію авторського права на твір та видача свідоцтва будуть здійснені за умови сплати збору за оформлення і видачу свідоцтва про реєстрацію авторського права на твір відповідно до п.3 постанови Кабінету Міністрів України від 27 грудня 2001 року № 1756 "Про державну реєстрацію авторського права і договорів, які стосуються права автора на твір".

Якщо протягом трьох місяців від дати одержання заявником рішення про реєстрацію авторського права на твір Управління державних реєстрацій Департаменту інтелектуальної власності Міністерства економічного розвитку і торгівлі України не одержало документ про сплату збору за оформлення і видачу свідоцтва у розмірі та порядку, визначених законодавством, або копію документа, що підтверджує право на звільнення від сплати зазначеного збору, заявка вважається відхиленою і реєстрація авторського права та публікація відомостей про реєстрацію Управлінням державних реєстрацій Департаменту інтелектуальної власності Міністерства економічного розвитку і торгівлі України не проводиться.

Державний секретар
Міністерства економічного розвитку
і торгівлі України **О. Ю. Перевезенцев**

М.П.

Рисунок Г.1 – Копія рішення про реєстрацію авторського права на твір Комп'ютерна програма «SAB Analytical Studio»

УКРАЇНА

СВІДОЦТВО

про реєстрацію авторського права на твір

№ 77488

Комп'ютерна програма "SAB Analytical Studio"
(вид, назва твору)

Автор(и) Бакун Сабіна Антонівна
(повне ім'я, псевдонім (за наявності))

Дата реєстрації 12.03.2018

МІНІСТЕРСТВО ЕКОНОМІЧНОГО РОЗВИТКУ І ТОРГІВЛІ УКРАЇНИ ІДЕНТИФІКАЦІЙНИЙ КОД 37508596

М.П.

Державний секретар Міністерства економічного розвитку і торгівлі України О. Ю. Перевезенцев

УКРАЇНА • UKRAINE • УКРАЇНА • UKRAINE • УКРАЇНА • UKRAINE • УКРАЇНА • UKRAINE • УКРАЇНА • UKRAINE • УКРАЇНА

ПК «Україна». Зам. 17-2020. 2017 р. IV кв.

Рисунок Г.2 – Копія свідоцтва про реєстрацію авторського права на твір Комп’ютерна програма «SAB Analytical Studio»

ДОДАТОК Д НАУКОВІ ПУБЛІКАЦІЇ

Статті

1. Бакун С.А. Методика побудови скорингових карт із використанням платформ SAS / С.А. Бакун, П.І. Бідюк // Наукові вісті Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». - 2016. - № 2. - С. 23-32. – Режим доступу: http://nbuv.gov.ua/UJRN/NVKPI_2016_2_5

Участь у конференціях

2. Бакун С.А. Методи перетворення категоріальних змінних в числові / С.А. Бакун, О.М. Терентьєв // Системний аналіз та інформаційні технології: Матеріали XX міжнародної науково-технічної конференції «Системний аналіз та інформаційні технології». – К.: НТУУ «КПІ ім. Ігоря Сікорського», 2018.
3. Бакун С.А. Using SAS Enterprise Miner to build scoring card to evaluation solvency of individuals / С.А. Бакун, А.А. Литвинюк, О.М. Терентьєв // Системний аналіз та інформаційні технології: Матеріали XVIII міжнародної науково-технічної конференції «Системний аналіз та інформаційні технології». –К.: НТУУ «КПІ ім. Ігоря Сікорського», 2016.
4. Бакун С.А. Using of credit scoring to deciding on the loan / С.А. Бакун, А.А. Литвинюк, О.М. Терентьєв // Інформаційно-комунікаційні технології навчання: Матеріали всеукраїнської науково-практичної інтернет-конференції «Інформаційно-комунікаційні технології навчання». – Умань: УДПУ ім. Павла Тичини, 2016.