# РОЗДІЛ 2 МЕТОДИ ТА ЗАСОБИ ОБРОБКИ ІНФОРМАЦІЇ

2.1 Первинна обробка даних для побудови моделі. Методи обробки та аналізу даних

Дано репрезентативну вибірку клієнтів банку(1000 клієнтів, 20 змінних).

\* процес «Data Preparation» реалізований на мові R, скрипт знаходиться у додатку до магістерської роботи.

Підготовка даних - це не лише перший крок, а й багато повторних кроків(алгоритмів) протягом аналізу, коли з'являються нові проблеми або з'являються нові дані. Тому не дивно, що є поширена думка, що до 80% процесу аналізу даних - це час, витрачений на їх підготовку[12].

Дані для вибірки можуть мати неточності, «викривлені» значення та ін. Щоб зробити модель більш точною наведемо кілька прикладів вдосконалення перинної структури даних.

2.1.1. Заміна пропущених значеннь

Часто в даних, з якими необхідно працювати, присутні пропуски, в результаті чого аналітик виявляється перед вибором: відкинути або ж заповнити пропущені значення. Взагалі існує кілька варіантів боротьби із пропущеними значеннями:

• якщо змінна має занадто багато відсутніх значень, то видаляємо її;

• Якщо у змінної дуже мало пропущених значень, то краще видалити клієнтів, у яких є пропущені значення;

• Якщо видалення клієнтів з відсутніми значеннями зменшить суттєво розмір вибірки, то заміняємо прощущені значення.

Заповнення пропусків часто, і цілком обґрунтовано, здається більш привабливим рішенням. Однак це не завжди так. Невдалий вибір методу заповнення пропусків може не тільки не поліпшити, а й сильно погіршити результати моделі.

Виняток ігнорування рядків з пропущеними значеннями стало рішенням за замовчуванням в деяких популярних прикладних пакетах, в результаті може виникнути уявлення, що дане рішення - правильне. Крім того, існують досить прості в реалізації і використанні методи обробки пропусків, що отримали назву «ad-hoc методи»: заповнення пропусків нулями, медіаной, середнім арифметичним значенням, введення індикаторних змінних тощо, простота яких може послужити причиною для вибору саме цих методів .

Ймовірно, саме через свою простоту ad-hoc методи широко використовувалися на зорі розвитку сучасної теорії обробки пропусків. І хоча за станом на сьогоднішній день відомо, що застосування цих методів може призводити до спотворення статистичних властивостей вибірки і, як наслідок, до погіршення результатів, одержуваних після такої обробки пропусків [13], їх як і раніше часто використовують. Так, відомі статті, присвячені збору та оцінки статистики використання методів заповнення пропусків в наукових роботах медичної тематики [14], з результатів яких можна зробити висновок, що навіть вчені часто віддають перевагу інтуїтивно-зрозумілим ad -hoc методам і ігнорування / видалення рядків, незважаючи на те, що застосування цих методів в контексті розв'язуваної задачі часом недоречно.

Механізми формування пропусків

Для того щоб зрозуміти, як правильно обробити пропуски, необхідно визначити механізми їх формування.

Розрізняють такі 3 механізму формування пропусків: MCAR, MAR, MNAR.

MCAR (Missing Completely At Random) - механізм формування пропусків, при якому ймовірність пропуску для кожного запису набору однакова. Наприклад, якщо проводилося соціологічне опитування, в якому кожному десятому респонденту одне рандомно вибране питання не ставилося, причому на всі інші поставлені запитання респонденти відповідали, то тут має місце механізм MCAR. В такому випадку ігнорування / виключення записів з даних не призведе до спотворення результатів.

MAR (Missing At Random) - на практиці дані зазвичай пропущені не випадково, а з огляду на деяких закономірностей. Пропуски відносять до MAR, якщо ймовірність пропуску може бути визначена на основі іншої наявної в наборі даних інформації (стать, вік, займана посада, освіта ...), яка не містить пропуски. В такому випадку видалення або заміна пропусків на значення «Пропуск», як і в разі MCAR, не призведе до суттєвого спотворення результатів.

MNAR (Missing Not At Random) - механізм формування пропусків, при якому дані відсутні в залежності від невідомих чинників. MNAR передбачає, що ймовірність пропуску могла б бути описана на основі інших атрибутів, але інформація по цим атрибутам в наборі даних відсутня. Як наслідок, ймовірність пропуску неможливо виразити на основі інформації, що міститься в наборі даних.

Розглянемо відмінності між механізмами MAR і MNAR на прикладі.

Люди, що займають керівні посади і / або які отримали освіту в престижному вузі частіше, ніж інші респонденти, які не відповідають на питання про свої доходи. Оскільки посада і освіту сильно корелюють з доходами, то в такому випадку пропуски в поле доходи вже не можна вважати абсолютно випадковими, тобто говорити про випадок MCAR не представляється можливим.

Якщо в наборі даних є інформація про освіту і посади респондентів, то залежність між підвищеною ймовірністю пропуску в графі доходів і цією інформацією може бути виражена математично, отже, виконується гіпотеза MAR. У разі MAR виключення пропусків цілком прийнятно.

Однак якщо інформація про займану посаду та освіті у нас відсутній, то тоді має місце випадок MNAR. При MNAR просто ігнорувати або виключити пропуски вже не можна, так як це призведе до значного спотворення розподілу статистичних властивостей вибірки.

Розглянемо прості методи обробки пропусків і пов'язані з ними проблеми.

Видалення / ігнорування пропусків

Complete-case Analysis (Listwise Deletion Method) - метод обробки пропусків, який застосовується в побудові моделі як метод за замовчуванням. Полягає у виключенні з набору даних записів / рядків або атрибутів / колонок, що містять пропуски.

У разі першого механізму пропусків (MCAR) застосування даного методу не призведе до суттєвого спотворення параметрів моделі. Проте видалення рядків призводить до того, що при подальших обчисленнях використовується не вся доступна інформація, стандартні відхилення зростають, отримані результати стають менш репрезентативними. У випадках коли пропусків в даних багато, це стає відчутною проблемою.

Крім того, в разі другого (MAR) і, особливо, третього механізму пропусків (MNAR) зміщення статистичних властивостей вибірки, значень параметрів побудованих моделей і збільшення стандартних відхилень стають ще сильніше.

Таким чином, незважаючи на широке поширення, застосування даного методу для вирішення практичних завдань обмежена.

Available-case analysis (він же Pairwise Deletion) - методи обробки, заснований на ігноруванні пропусків в розрахунках. Ці методи, як і Complete-case Analysis, теж часто застосовуються за замовчуванням.

Статистичні характеристики, такі як середні значення, стандартні відхилення, можна розрахувати, використовуючи всі непропущені значення для кожного з атрибутів / стовпців. Як і в разі Complete-case Analysis, за умови виконання гіпотези MCAR, застосування даного методу не призведе до суттєвого спотворення параметрів моделі.

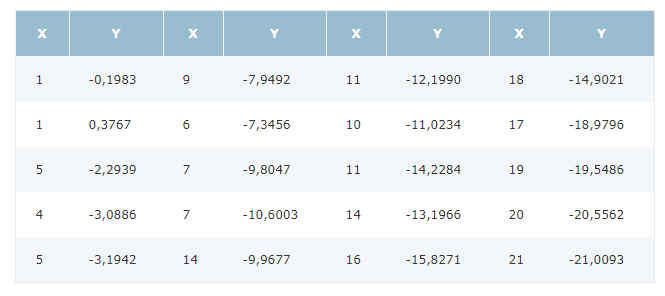
Перевага даного підходу в тому, що при побудові моделі використовується вся доступна інформація.

Головний же недолік даних методів полягає в тому, що вони можуть бути застосовані для розрахунку далеко не всіх показників і, як правило, пов'язані з алгоритмічними і обчислювальними труднощами, що призводять до некоректних результатів.

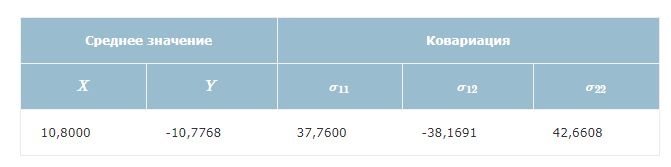
Наприклад, розраховані значення коефіцієнтів кореляції можуть опинитися поза діапазону [-1; 1]. Крім того, не завжди вдається однозначно відповісти на питання про оптимальний вибір числа змінних у моделі, якщо ми хочемо обрати оптимальне число змінних за допомогою методів, які засновані на розрахунку стандартних відхилень.

Наведемо приклад, який демонструє проблеми методів Available-case analysis.

Розглянемо наступну задачу: необхідно розрахувати лінійний коефіцієнт кореляції (коефіцієнт кореляції Пірсона) між двома факторами / змінними X і Y.



Таблиця 2.1.1. Значення для аналізу кореляції Пірсона

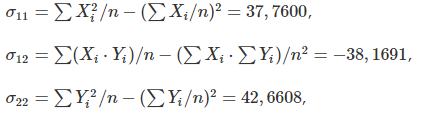


Таблиця 2.1.2. Значення кореляції Пірсона

Середнє значення X: 10,8000.

Середнє значення Y: -10,7768.

Оцінка (ко) варіації:



де n - число спостережень (n = 20).

Значення коефіцієнта кореляції:

F:\Дипломна робота_2\картинки\4_4.PNG

Розглянемо результати аналогічних розрахунків при наявності пропусків в даних (дані представлені в таблиці 2).

Таблиця 2.1.3. Дані із пропущеними значеннями

Тобто працюємо з тим же набором даних (що і в таблиці 2.1.1), з тією лише відмінністю, що в даному випадку нам невідомі два перших значення змінної X.

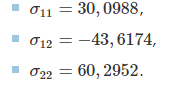
В рамках Available-case analysis підходу ми вважаємо середнє значення, використовуючи всю доступну інформацію, тобто для змінної X на основі 18 відомих значень, а для змінної Y на основі всіх 20 значень.

Таким чином, на основі таблиці 2 отримаємо наступні результати:

cреднее значення X: 11,8889,

cреднее значення Y: -10,7768,

оцінка (ко) варіації:



Значення коефіцієнта кореляції:

r = -1,0239.

Таким чином, розрахунок середнього значення на основі підходу Available-case Analysis призвела до зміщення даного значення, що в свою чергу, проявилося в розрахованому значенні коефіцієнта кореляції меншим -1. Таким чином, розраховане значення вийшло за межі теоретично можливого діапазону [-1; 1]), що суперечить фізичному змісту.

Якщо ж розрахувати значення коефіцієнта кореляції в рамках підходу Complete-case Analysis, то отримаємо значення коефіцієнта кореляції: -0,9311.

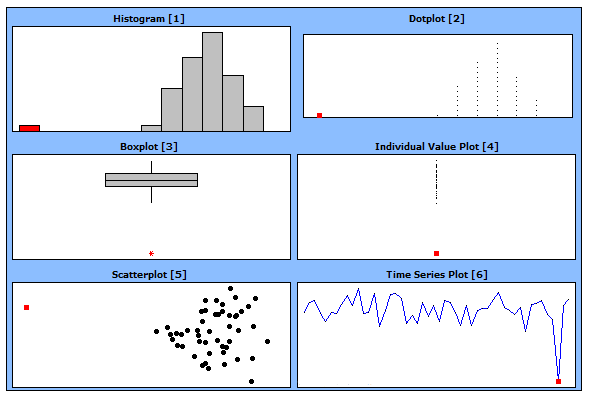
Коли гіпотеза MCAR не виконується, методи Available-case analysis так само, як і методи Complete-case Analysis призводять до суттєвих перекручень статистичних властивостей вибірки (середнього значення, медіани, варіації, кореляції).

До недоліків перших двох методів обробки пропусків (Complete-case Analysis і Available-case analysis) відноситься і те, що, далеко не завжди виключення рядків прийнятно. Нерідко процедури подальшої обробки даних припускають, що всі рядки і колонки беруть участь в розрахунках (наприклад, коли пропусків в кожному стовпчику і в рядках небагато).

Тобто, це було очевидно, але ми переконалися, що вставляти пропущені значення потрібно, проте потрібно вставляти їх покладаючись а певні залежності в даних.

2.1.2. Викиди (надто малі або надто великі значення)

Викид - це результат вимірювання, що виділяється із загальної вибірки. Іншими словами, викиди - це незвично низькі або високі значення спостережуваної величини, причому настільки, що це помітно неозброєним оком: в ході графічного аналізу спостережень ви можете помітити значення, яке не належить популяції спостережень. Визначити викиди можна за допомогою: гістограм [1], точкових [2] і ящикових [3] діаграм, діаграм індивідуальних значень [4], розсіювання [5] і навіть діаграм часових рядів [6]:



Таблиця 2.1.4. Різне представлення викидів

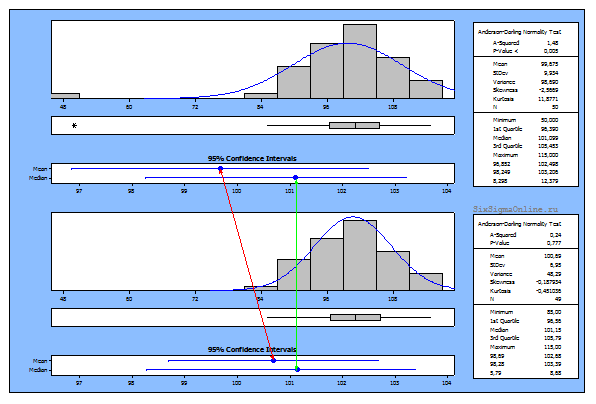
Червоні точки, зірочки і стовпці на діаграмах відповідають викидам.

В теорії статистичного аналізу немає однозначного критерію ідентифікації викидів, і це - перша причина, по якій викиди становлять небезпеку.

З визначення випливає, що всі незвично низькі або високі значення спостережуваної величини можуть бути викидами. Як же визначити, яке значення змінної є надзвичайно високим або низьким. Один з найпростіших способів: використовувати діапазон трьох стандартних відхилень навколо середнього значення. Ймовірність виходу величини за межі ± 3σ становить 0,0027, а значить, з великою часткою ймовірності, значення, яке виходить за межі ± 3σ не належить до популяції.

З іншого боку, можна привести ряд доводів проти цього твердження. Наприклад, воно втрачає сенс, якщо функція розподілу відрізняється від нормальної або розмір вибірки занадто малий, щоб представити генеральну сукупність значень. Крім того, з імовірністю 0,0027 спостереження все ж може вийти за межі діапазону трьох стандартних відхилень.

Друга небезпека, яку представляють викиди - спотворення статистик або результатів статистичних розрахунків. Такі показники як середнє арифметичне (Mean), стандартне відхилення (StDev), асиметрія (Skewness), ексцес (Kurtosis), а також критерій згоди з нормальним законом вельми схильні до впливу викидів. На відміну від середнього арифметичного, медіана менш схильна до впливу викидів. На наступному малюнку медіана і середнє арифметичне до і після виключення викиду позначені зеленою і червоною стрілками відповідно:



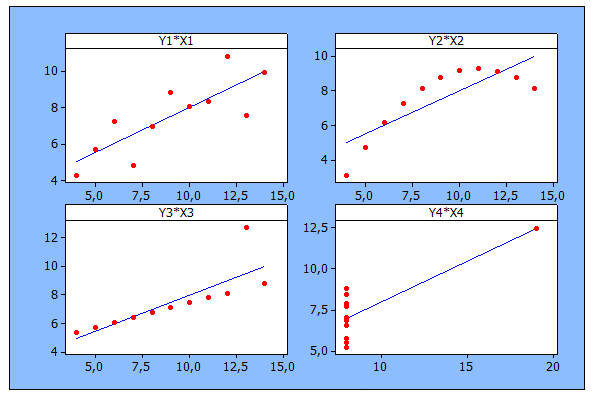
Таблиця 2.1.5 Вплив викидів на статистичну оцінку даних

Ще один класичний приклад - квартет Енскомба (Anscomb):

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **X1** | **Y1** | **X2** | **Y2** | **X3** | **Y3** | **X4** | **Y4** |
| 10 | 8,04 | 10 | 9,14 | 10 | 7,46 | 8 | 6,58 |
| 8 | 6,95 | 8 | 8,14 | 8 | 6,77 | 8 | 5,76 |
| 13 | 7,58 | 13 | 8,74 | 13 | 12,74 | 8 | 7,71 |
| 9 | 8,81 | 9 | 8,77 | 9 | 7,11 | 8 | 8,84 |
| 11 | 8,33 | 11 | 9,26 | 11 | 7,81 | 8 | 8,47 |
| 14 | 9,96 | 14 | 8,1 | 14 | 8,84 | 8 | 7,04 |
| 6 | 7,24 | 6 | 6,13 | 6 | 6,08 | 8 | 5,25 |
| 4 | 4,26 | 4 | 3,1 | 4 | 5,39 | 19 | 12,5 |
| 12 | 10,84 | 12 | 9,13 | 12 | 8,15 | 8 | 5,56 |
| 7 | 4,82 | 7 | 7,26 | 7 | 6,42 | 8 | 7,91 |
| 5 | 5,68 | 5 | 4,74 | 5 | 5,73 | 8 | 6,89 |

Таблиця 2.1.6. Дані квартету Енскомба

Квартет Енскомба - це чотири набори числових даних, які використовують як свідчення важливості візуальної оцінки спостережень в кореляційному і регресійному аналізі:



Таблиця 2.1.7. Приклади кореляції

Не дивлячись на відмінності взаємозв'язку змінних X і Y, у всіх чотирьох випадках статистичні показники, як і рівняння лінійної регресії, однакові:

**Характеристика Значення**

Середнє значення змінної X 9

Дисперсія змінної X 10

Середнє значення змінної Y 7,5

Дисперсія змінної Y 3,75

Коефіцієнт кореляції Пірсона 0,816

Рівняння лінійної регресії Y = 3 + 0.5X

Викид в третьому прикладі спотворює рівняння залежності, а в четвертому - змушує прийняти рішення про наявність кореляції, в той час як її насправді немає.

І, нарешті, третя небезпека, яку приховують викиди - це легкість їх невірного тлумачення, що, в свою чергу, призведе до невірного напрямку подальшого аналізу. Наявність викидів може означати помилку введення даних, недостатню величину вибірки або присутність спеціальної причини відхилення - дія конкретного фактора або причини. Діагностуючи викиди, легко допустити помилку, виключивши потрібні для аналізу дані або навпаки - розрахувавши показники процесу, використовуючи неправильні результати спостережень

Неуважне ставлення до викидів спостережень ставить під загрозу висновки про спостереження процесу і ставить під загрозу результати подальшого аналізу. Отже, виявивши незвично низькі або високі значення спостережуваної величини, дослідник повинен визначити причину їх появи, перш ніж робити висновки про спостерігається змінної або приступати до подальшого аналізу даних.

Методи боротьби із викидами:

* якщо виключити викиди із змінної, то це призведе до змін середнього значення та дисперсії. Враховуючи, що викиди також вважаються значеннями, виключення їх з аналізу робить цей підхід неадекватним для «регулювання» змінної;

## встановити всі відхилення до заданого процентилу даних; наприклад, 98% - це всі дані без викидів. Остальні: 1% - дані з надто малими значеннями, інший 1% - з надто великими значеннями. Ці дані з надто малими та великими значеннями замінюємо на найменші та ,відповідно, найбільші значення 98% даних без викидів.

Метод бінінгу використовують, щоб не змінювати цілісність даних при цьому позбавитися проблеми із викидами.

2.1.3. Бінінг

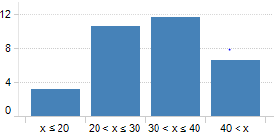
В скоринговії моделі можуть використовуватися в якості незалежних змінних категоріальні і кількісні предиктори. Багато розробників скорингових систем використовують завжди метод категоризації кількісних змінних(binning).

Рис. 2.1.2. Змінна «Вік» із застосуванням методу бінінгу

Категоризація кількісних змінних дозволяє досягти таких основних переваг при побудові скорингової карти: полегшити обробку викидів та екстремальних значень кількісних змінних; спростити інтерпретацію скорингової карти; відобразити складні нелінійні зв'язки. Розглянемо основні методи бінінгу, які застосовуються у скорингу на основі змінної «Вік» (age\_in\_yrs – вік, 1 о.в. – 1 рік).Розглянемо емпіричний розподіл змінної(без бінінгу):

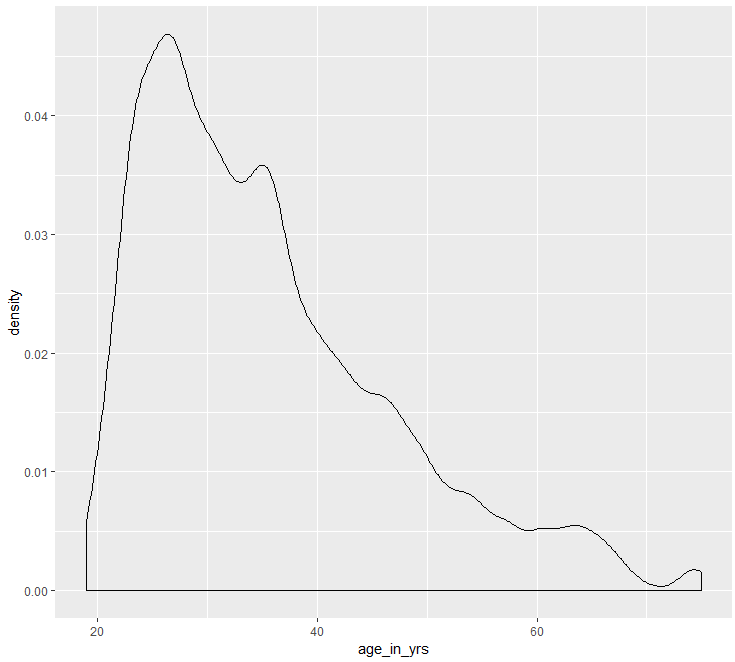
****

Рис.2.1.3. Емпіричний розподіл змінної «age\_in\_yrs»

## Методи бінінгу:

Типові методи, які широко застосовуються у побудовах моделі;

*Equal-width (distance):*

## ділить діапазон на N інтервалів однакового розміру;

## якщо A і B є найнижчими і найвищими значеннями змінної, то ширина інтервалів буде: W = (B-A) / N;

## чутливий до відхилень, нерівні дані не обробляються добре.

## **C:\Users\User\Desktop\практика\eq_width.PNG**

Рис.2.1.4. Застосування «Equal-width (distance)» на змінній «age\_in\_yrs»

## *Equal-depth (frequency):*

## Він ділить діапазон на N інтервалів, кожен з яких містить приблизно однакову кількість спостережень;

## Гарне масштабування даних.

## **C:\Users\User\Desktop\практика\eq_depth.PNG**

Рис.2.1.4. Застосування «Equal-width (frequency)» на змінній «age\_in\_yrs»

## Проте можна застосувати нові методи бінінгу:

## *“Jenks Natural Breaks” method:*

## метод почергово перебирає всі можливі варіанти класів, рахуючи при цьому міжкласову та міжгрупову дисперсію, отже, вимагає затратів по часу;

## ідея методу полягає у мінімізації дисперсії в класах і максималізації міжкласової дисперсії.

## **C:\Users\User\Desktop\практика\jenks.PNG**

Рис.2.1.5. Застосування «Jenks Natural Breaks» на змінній «age\_in\_yrs»

## Ще одним методом бінінгу, метод кластерного аналізу, - *K-means.*

## Jenks та K-means відрізняються тим, як вони мінімізуються в межах відстаней групи.

## Jenks використовує той факт, що одновимірні дані сортуються, що робить його більш швидким алгоритмом для одновимірних даних.

## K-means є більш загальними алгоритмом, він ефективний із даними великої розмірності.

## Надалі розглянемо методи обробки інформації за допомогою методу *K-means* та застосуємо метод на наших даних.

## Мета методу — розділити n спостережень на k кластерів, так щоб кожне спостереження належало до кластера з найближчим до нього [середнім значенням](https://uk.wikipedia.org/wiki/Середнє_арифметичне). Метод базується на мінімізації суми квадратів відстаней між кожним спостереженням та центром його кластера, тобто функції

,

де d — метрика, xi{\displaystyle x\_{i}} — і-ий об'єкт даних, а {\displaystyle m\_{j}(x\_{i})}mj(xi) — центр кластера, якому на j-ій ітерації приписаний елемент {\displaystyle x\_{i}} xi{\displaystyle x\_{i}} .

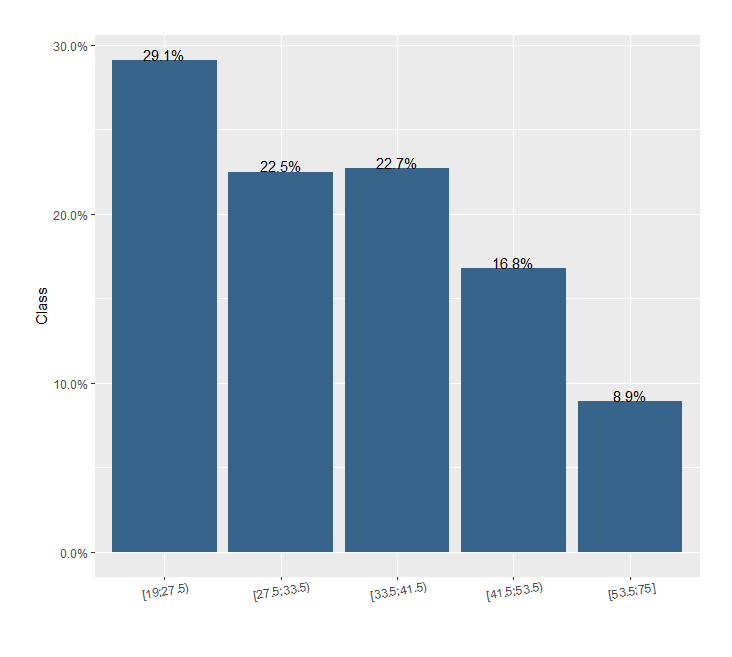


Рис.2.1.5. Застосування методу «K-means» на змінній «age\_in\_yrs»

Ми провели ознайомлення із методами обробки інформації. Надалі проведемо аналітичну діяльність над магістерським завданням та протестуємо запропоновані методи аналізу інформації.

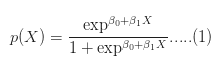
2.2. Розробка та аналіз методів, які можуть покращити прогностичну властивість скорингової моделі

2.2.1. LASSO регресія

LASSO (least absolute shrinkage and selection operator - потужний метод оцінки коефіцієнтів регресії. Лассо є по суті методом регуляризації. Даний метод зменшує перенавчання, використовуючи менш складні функції . Один із способів зробити це - викинути менш важливі змінні після перевірки того, що вони не є важливими. Це можна зробити, вивчивши p-значення коефіцієнтів і відкинувши ті змінні, коефіцієнти яких не є значущими. Проте, це займає багато часу для проблем класифікації з багатьма незалежними змінними. У таких ситуаціях лассо пропонує спосіб моделювання залежної змінної при автоматичному виборі значущих змінних шляхом зменшення коефіцієнтів неважливих змінних до нуля.

Як діє логістична регресія із використанням Лассо-регулізації?

Припустимо, що результат (передбачена змінна) і предиктори позначаються відповідно Y і X, і два класи, що представляють інтерес, позначаються + і - відповідно. Ми хочемо моделювати умовну ймовірність того, що результат Y дорівнює +, враховуючи, що вхідними змінними (предикторами) є X. Умовна ймовірність позначається p (Y = + | X), яку ми будемо скорочувати як p(X):



Можна переконатися, що p(X) дійсно лежить між 0 і 1, оскільки X змінюється від - ∞ до + ∞. На малюнку також показано типову S-подібну криву, характерну для логістичної регресії.

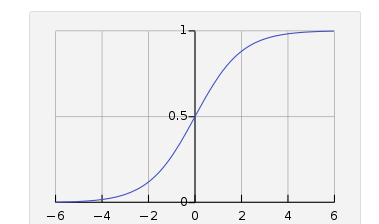


Рис.2.2.1. Приклад логістичної регресії

З’ясуємо звідки походить назва «логістична». Еквівалентним способом вираження наведеного рівняння є:

F:\Дипломна робота_2\картинки\модель\3.PNG

Значення ліворуч - логарифм шансів. Отже, модель являє собою лінійну регресію логарифмічних коефіцієнтів, іноді званий logit, і звідси назва логістична.

Проблема полягає в тому, щоб знайти значення beta\_0 і beta\_1 і т.д., які призводять до p(X), що найбільш точно класифікує всі спостережувані точки даних - тобто ті, які належать до позитивного класу, мають ймовірність максимально наближеною до 1 і ті, що належать до негативного класу, мають ймовірність максимально близьку до 0. Один із способів сформулювати цю проблему:

F:\Дипломна робота_2\картинки\модель\4.PNG

Де представлені продукти над i та j, які проходять над + і - класифікованими точками відповідно. Такий підхід, який називається оцінкою максимальної правдоподібності, досить часто зустрічається у багатьох налаштуваннях машинного навчання, особливо тих, що стосуються ймовірностей.

Більше того, мінімізується ймовірність негативного логарифму правдоподібності, що, звичайно, є таким же, як максимізація ймовірності правдоподібності. Мінімізується таким чином:

F:\Дипломна робота_2\картинки\модель\5.PNG

Проте, це теоретичні деталі, які я згадую тільки для повноти. Як ви побачите далі, вони мало впливають на практичне використання логістичної регресії із лассо регулізацією.

Lasso регресія

Отже, значення коефіцієнтів логістичної регресії beta\_0 і beta\_1 і т.д. виявляються шляхом мінімізації ф-ції правдоподібності ,описаного в рівнянні (3). Ще одним поширеним методом регуляризаціє є рідж регуляризацяя. Рідж і лассо регуляризації працюють шляхом додавання штрафу на функцію правдоподібності. У разі рідж регресії термін штрафу – це сума «beta» коефіцієнтів у квадраті, а у випадку ласо - це сума «beta» коефіцієнтів взятих по модулю .Тобто для рідж резуляризації:

F:\Дипломна робота_2\картинки\модель\6.PNG

Для лассо:

F:\Дипломна робота_2\картинки\модель\7.PNG

Де лямбда - це вільний параметр, який зазвичай вибирається таким чином, щоб отримана модель мінімізувала помилку вибірки.

У разі рідж регресії ефект регулізації полягає в зменшенні коефіцієнтів, які найбільше впливають на помилку моделі. Іншими словами, це зменшує величину коефіцієнтів, що сприяють збільшенню L. На відміну від цього, у випадку регресії лассо, ефект штрафу проявляється у зменшенні цих коефіцієнтів практично до нуля! Це добре, тому що це означає, що регресія лассо працює як селектор змінних, тобто вибирає найважливіші змінні, ті, які мають найбільшу прогностичну силу(мають найменші значення p).

Ми розглянули оптимізацію логістичної регресії, одну з найпростіших (і найстаріших) методів класифікації в арсеналі практикуючих методів. Незважаючи на простоту (або, треба сказати, через це), логістична регресія добре працює для багатьох бізнес-проблем, які часто мають прості рішення. Більш того, через її простоту вона менш схильна до перенавчання, ніж методи, побудовані на деревах рішень. Але даний метод можна оптимізувати усуненням змінних, які сприяють перенавчанню, змінні будуть усунені за допомогою лассо регуляризації, без погіршення точності. Враховуючи ці переваги та властиву їй простоту, не дивно, що логістична регресія залишається одним із найпопулярніших методів для скоронгових моделей.

2.2.2. Випадковий ліс

Bagging - використовує паралельне навчання базових класифікаторів. В ході беггінга відбувається наступне:

З безлічі вихідних даних випадковим чином відбирається кілька підмножин, що містять кількість прикладів, що відповідає кількості прикладів вихідного.

Оскільки відбір здійснюється випадковим чином, то набір прикладів завжди буде різним:

* деякі приклади потраплять в кілька підмножин, а деякі не потраплять ні в одне.
* На основі кожної вибірки будується класифікатор.
* Висновки класифікаторів агрегируются (шляхом голосування або усереднення).

Оскільки беггінг – це ансамбль дерев рішень, тобто модель, яка побудована на багатьох деревах рішень. Згодом переконаємося, що результат точності прогнозу буде кращим, ніж одинична модель на тому ж наборі даних.

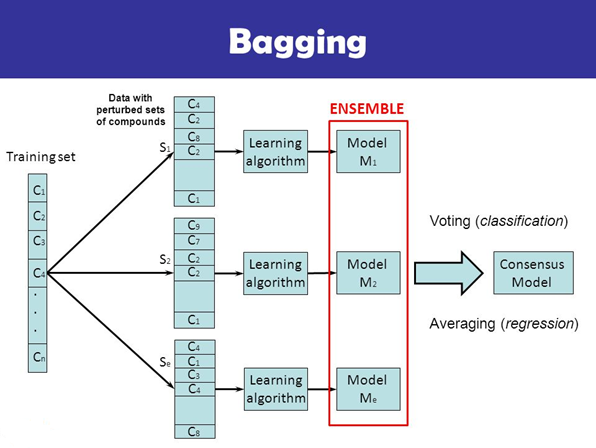


Рис.2.2.2. Bagging

Random forest (англ. Випадковий ліс) - алгоритм машинного навчання, що полягає у використанні комітету (ансамблю) дерев. Алгоритм поєднує в собі ідею методу беггінга. Алгоритм застосовується для задач класифікації, регресії і кластеризації.

Для задання алгоритму спочатку треба задати початкові параметри. Нехай навчальна вибірка складається з N прикладів, загальна кількість змінних у вибірці - M, кількість параметрів у кожному дереві – m.

Всі дерева будуються незалежно один від одного на різних частинах(«ділянках») вибірки:

Згенеруємо випадкову підвибірку з повторенням розмірності N з навчальної вибірки. (Таким чином, деякі приклади потраплять в неї кілька разів, а приблизно N / 3 прикладів не ввійдуть в неї взагалі).

Побудуємо дерево, що класифікує приклади даної підвибірки, причому в ході створення чергового вузла дерева будемо вибирати ознаки, на основі яких проводиться розбиття, не з усіх M ознак, а лише з m випадково обраних. Вибір найкращого з цих m ознак можна здійснюватися різними способами. Використовується невизначеність Джині(Gini Impurity), що застосовується також в алгоритмі побудови вирішальних дерев CART або критерій приросту інформації(Entropy and Information Gain).

Дерево будується до повного вичерпання підвибірки і не піддається процедурі прунінга (на відміну від вирішальних дерев, побудованих за таким алгоритмам, як CART або C4.5).

Класифікація об'єктів проводиться шляхом голосування: кожне дерево комітету відносить об'єкт до одного з класів, і перемагає той клас, за який проголосувала найбільша кількість дерев.

Оптимальне число дерев підбирається таким чином, щоб мінімізувати помилку класифікатора на тестовій вибірці.

Оскільки приблизно N / 3 прикладів не ввійдуть в тренувальну вибірку, тому на цих даних можна обрахувати точність моделі. Вибірку, що не ввійшла у навчальну називають – out-of-bag.

Як і в CARD та C4.5, Random Forest також має метод обрахунку найбільш інформативної змінної.

Після побудови моделі на навчальній вибірці кожну змінну тестують на даних out-of-bag та обраховують похибку для кожної змінної. Оскільки змінна задіяна у декількох деревах навчальної вибірки, тому і похибок буде декілька. Щоб отримати одне значення ці похибки усереднюють. Потім значення змінної перемішують на навчальній вибірці і знову рахують похибку на даних out-of-bag. Чим більша різниця похибки до і після перемішання значень змінної, тим більша інформативність змінної. Даний підхід має недолік – завищує значущість категоріальних змінних, якщо великий об’єм вибірки.

Основною перевагою даного методу є точність. Якість краща, ніж у моделі класифікації нейронних мереж та методу опорних векторів.Також здатна ефективно обробляти дані з великим числом ознак і класів. Однаково добре обробляються як неперервні, так і дискретні ознаки. Метод не чутливий до «викидів» та легко справляється із пропущеними значеннями. Існують методи оцінювання значущості окремих ознак в моделі.

Ці всі переваги роблять прогностичну властивість скорингової моделі, побудованої на методі ВЛ, набагато кращою, ніж на CART або C4.5.

Але також треба врахувати, що результати погано інтерпретується, на відміну від інших алгоритмів дерев рішень.

2.2.3

Градієнтний бустинг

Boosting - метод побудови ансамблю моделей, при якому створюється послідовність композиції алгоритмів, де кожна модель повинена компенсувати помилки, допущені композицією попередніх моделей.

F:\Дипломна робота_2\картинки\модель\11.PNGВідобразимо ідею бегінгу в алгоритмі градієнтного бустингу. Нам потрібно знайти модель F(x). Ми оцінюємо ефективність моделі, вибираючи функцію втрат L(y, F (x)). У нашому випадку функція втрат - це функція правдоподібності:

(Детальний огляд функції у розділі 2.2.2.(LASSO регресія)

F (x) = f (x) + λh(x) + λh'(x) + ...

f(x) -> модель над фактичним y.

h(x) -> модель, яка побудована на основі залишкової помилки (y-y'), де y' - прогноз f(x).

h''(x) побудована над залишком комбінованої моделі f (x) + h'(x). І так далі, до того моменту, коли модель досягне найкращої прогностичної здібності.

Де λ – коефіцієнт, який наз.learning rate(від англ. - швидкість навчання) – кількість інформації, яку ми будемо брати із нової моделі h.

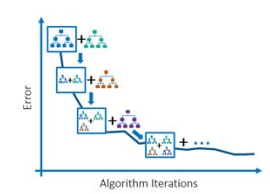


Рис.2.2.3. Алгоритм градієнтного бустингу

Отже, ми розглянули методи моделювання, за яким скорингова модель показує кращу точність. Ми отримаємо кращу прогностичну модель, якщо використаємо логістичну регресію із методом регуляризації – LASSO регресія. А також, якщо використати методи ансамблів, які основані та методах дерев рішень, то ми отримаємо набагато кращі результати, ніж на одиничній моделі дерев рішень.

2.3.1. Засоби обробки інформації

Статистичний аналіз є невід'ємною частиною наукового дослідження. Якісна обробка даних підвищує шанси опублікувати статтю в солідному журналі, і вивести дослідження на міжнародний рівень. Існує багато програм, здатних забезпечити якісний аналіз, однак більшість з них платні, і часто ліцензія коштує від кількох сотень доларів і вище. Як засіб обробки інформації я використала мову R, за яку не треба платити, а її надійність і популярність конкурують з кращими комерційними стат. пакетами: статистична мова програмування - R.

Що таке R?

Перш ніж дати чітке визначення, слід зазначити, що R - це щось більше, ніж просто програма: це і середовище, і мова.

R - це середовище обчислень, розроблене вченими для обробки даних, математичного моделювання та роботи з графікою. R можна використовувати як простий калькулятор, можна редагувати в ньому таблиці з даними, можна проводити прості статистичні аналізи (наприклад, t-тест, ANOVA або регресійний аналіз) і більш складні тривалі обчислення, перевіряти гіпотези, будувати векторні графіки і карти. Це далеко не повний перелік того, що можна робити в цьому середовищі. Варто відзначити, що вона поширюється безкоштовно і може бути встановлена ​​як на Windows, так і на операційні системи класу UNIX (Linux і MacOS X). Іншими словами, R - це вільний і багатоплатформовий продукт.

R - це мова програмування, завдяки чому можна писати власні програми (скрипти) за допомогою керуючих конструкцій, а також використовувати і створювати спеціалізовані розширення (пакети). Пакет - це набір R функцій, файлів з довідковою інформацією і прикладами, зібраних разом в одному архіві. R пакети грають важливу роль, так як вони використовуються як додаткові розширення на базі R. Кожен пакет, як правило, присвячений конкретній темі, наприклад: пакет 'ggplot2' використовується для побудови векторних графіків певного дизайну, а пакет 'qtl' ​​ідеально підходить для генетичного картування. Таких пакетів в бібліотеці R налічується на даний момент більше 7000! Всі вони перевірені на предмет помилок і знаходяться у відкритому доступі.

Як виглядає середу R?

Існує багато "оболонок" для R, зовнішній вигляд і функціональність яких можуть сильно відрізнятися. Але ми коротко розглянемо лише три найбільш популярних варіанти: Rgui, Rstudio і R, запущений в терміналі Linux / UNIX у вигляді командного рядка. Rgui - це стандартний графічний інтерфейс (https://cran.r-project.org/), вбудований в R за замовчуванням. Ця оболонка має вигляд командного рядка у вікні, званим консоллю. Командний рядок працює за принципом "питання-відповідь".

наприклад:

> 2 + 2 \* 2 # наше запитання / запит

[1] 6 # відповідь комп'ютера

Однак, для запису складного алгоритму команд в Rgui існує додаткове вікно, де пишеться програма (скрипт). Третім елементом даної оболонки є графічний модуль, який з'являється при необхідності відображення графіків.

На наведеному нижче малюнку, показана повна версія Rgui: консоль (зліва), скриптовими вікно і графічний модуль (праворуч).

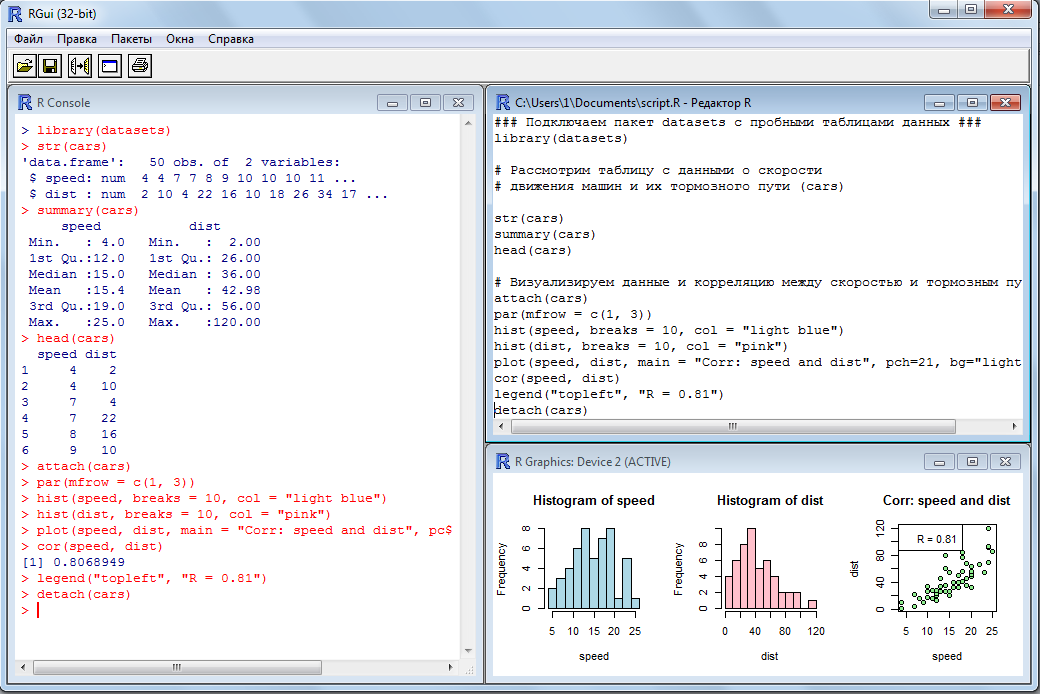


Рис2.3.1. Rgui - зовнішній вигляд

Rstudio - інтегроване середовище розробки (IDE) (https://www.rstudio.com/). На відміну від Rgui, у даній оболонки є заздалегідь розділені області та додаткові модулі (наприклад, історія команд, робоча область). На думку деяких користувачів, Rstudio має більш зручний інтерфейс, що спрощує роботу з R. Ряд особливостей, таких як підсвічування різними кольорами і автоматичне завершення коду, зручна навігація по скрипту робить Rstudio привабливою не тільки для новачків, але і для досвідчених програмістів.

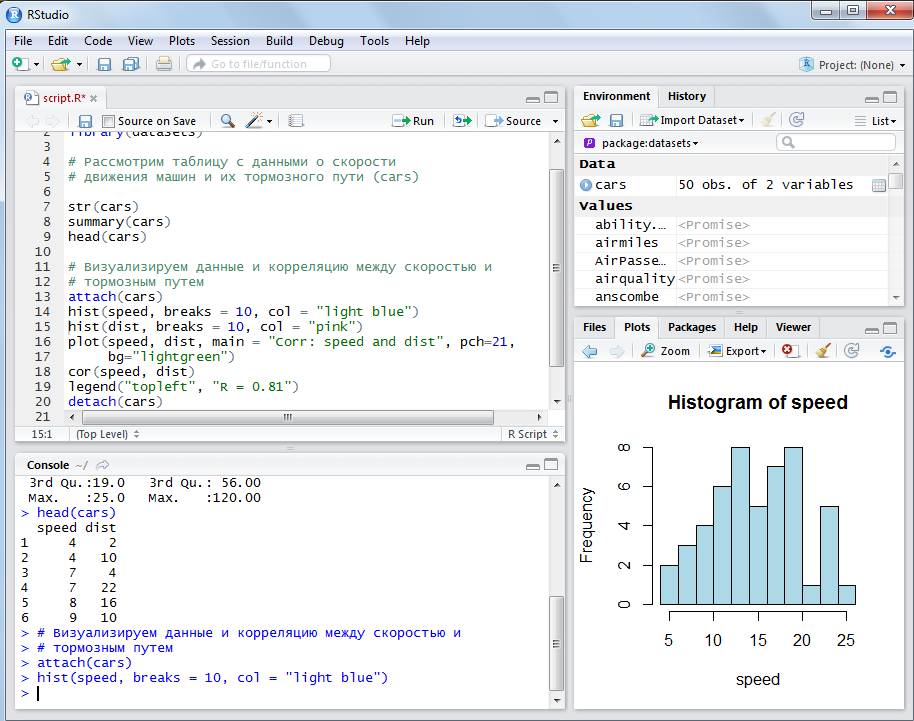


Рис.2.3.2.Rstudio зовнішній вигляд

R в терміналі Linux / UNIX.

Даний варіант кращий для аналізу великого обсягу даних через сервер, суперкластер або суперкомп'ютер. Більшість з них працюють на операційних системах класу Linux / UNIX, доступ до яких здійснюється через термінал команд (наприклад, bash). R в терміналі є додаток, запущене у вигляді командного рядка.

Порівняння R з мовами програмування MatLab, Python і Julia

Серед мов програмування, використовуваних в статистичних розрахунках, лідируючі позиції займають R і Matlab. Вони схожі між собою, як за зовнішнім виглядом, так і по функціональності. Історично MatLab був орієнтований на прикладні науки інженерних спеціальностей, тому його сильними сторонами є мат. моделювання та розрахунки, до того ж він набагато швидше R. Але так як R розроблявся як вузькопрофільна мова для статистичної обробки даних, то багато експериментальних стат. Методів з'являлися і закріплювалися саме в ньому. Цей факт і нульова вартість зробили R ідеальним майданчиком для розробки та використання нових пакетів, що застосовуються в фундаментальних науках.

Іншими "конкуруючими" мовами є Python і Julia. На мою думку, Python, будучи універсальний мовою програмування, більше підходить для машинного навчання, ніж для статистичного аналізу та візуалізації. А ось статистична мова Julia - досить молодий і претензійний проект. Основною особливістю цієї мови є швидкість обчислень, в деяких тестах перевищує R в 100 разів! Поки Julia знаходиться на ранній стадії розвитку і має мало додаткових пакетів і послідовників, але в віддалений перспективі Julia - це, мабуть, єдиний потенційний конкурент R.

Таким чином, в даний час мова R є одним з провідних статистичних інструментів в світі. Він активно застосовується в генетиці, молекулярній біології, науках про навколишнє середовище (екологія, метеорологія) та сільськогосподарських дисциплінах. Також R все більше використовується в обробці медичних даних, витісняючи з ринку такі комерційні пакети, як SAS і SPSS.

Переваги середовища R:

безкоштовна;

багатий арсенал стат. методів;

якісна векторна графіка;

більше 7000 перевірених пакетів;

гнучка у використанні:

- дозволяє створювати / редагувати скрипти і пакети,

- взаємодіє з іншими мовами, такими: C, Java і Python,

- може працювати з форматами даних для SAS, SPSS та STATA;

активна спільнота користувачів та розробників;

регулярні оновлення, хороша документація і тех. підтримка.

Недоліки:

невеликий обсяг інформації російською/українською мовою (хоча за останні п'ять років з'явилося кілька навчальних курсів і цікавих книг).

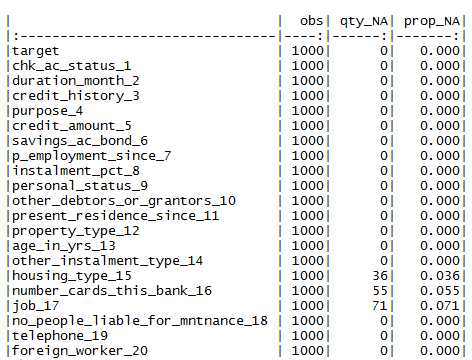
Невисока швидкість обрахунків.

2.4. Висновки

У даному розділі містяться методи обробки даних, за допомогою яких якість скорингової моделі стає кращою. Зокрема, ідеї обробки пропущених значень та відсіву викидів, а це потрібно для того щоб побудувати адекватну скорингову модель. Описаний метод біннінгу – розбиття неперервних данх на категорії(для обробки даних при побудові моделей). Також розглянуті сучасні алгоритми для оцінки кредитоспроможності клієнтів банку. Це моделі інтелектуального аналізу даних - логістична регресія з використанням LASSO регуляризації, дерева рішень, Random forest, градієнтний бустинг. Описані основні ідеї бегінгу та бустінгу.

Тобто, ми описали основні засоби та етапи побудови скорингової моделі за допомогою яких можна оптимізувати її точність.

Проаналізуємо пропущені значення у нашій вибірці:



Отже, змінні «Тип власності», «К-сть карток в цьому банку» та «Посада» мають пропущені значення.