Міністерство освіти і науки України

Національний технічний університет України

“Київський політехнічний інститут ім. Ігоря Сікорського”

Кафедра автоматизації проектування енергетичних процесів і систем

Звіт

з лабораторної роботи №5

«Ансамбль вирішальних дерев - випадковий ліс. Банківська справа. Оцінка кредитоспроможності клієнтів банку при видачі кредитів.»

Виконав:

студент 5-го курсу

ТЕФ

гр. ТМ-01мп  
Каліка Б.М.

Перевірив:

Проф. Федорова Н.В.

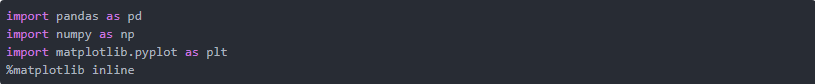
Київ – 2020

Для процесу вирішення ансамблю вирішальних дерев ми використовуємо клас DecisionTreeRegressor бібліотеки дерев. Також DecisionTreeRegressor відрізняються від матриць класифікації.

**Набір даних**

Набір даних, який ми використаємо для цього розділу, такий самий, як і в лабораторній роботі №4.

Тепер застосуємо наш алгоритм дерева рішень до цих даних, щоб спробувати передбачити можливість видачі кредиту на основі цих даних.



**Імпортуємо дані:**



**Аналіз даних**

Ми знову використаємо функцію head фрейму даних, щоб побачити, як насправді виглядають наші дані:



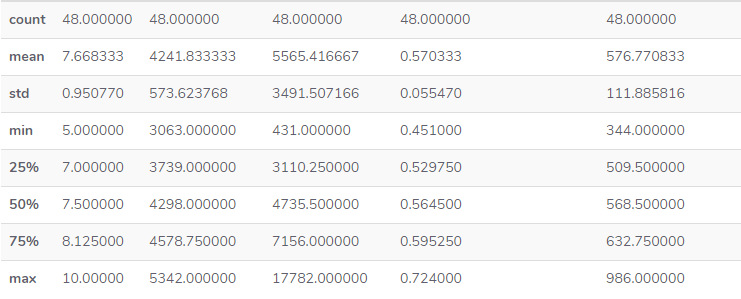
В результаті отримаємо:



Щоб переглянути статистичні деталі набору даних, виконаємо таку команду:



Отримаємо разультат:



**Підготовка даних**

Тут ми розділимо наші дані на атрибути та мітки, а отже, на навчальні та тестові набори.

Виконаємо такі команди, щоб розділити дані на мітки та атрибути:



Тут змінна X містить усі стовпці з набору даних, за винятком стовпця "transations", який є міткою. Змінна y містить значення зі стовпця " transations", що означає, що змінна X містить набір атрибутів, а змінна y містить відповідні мітки.

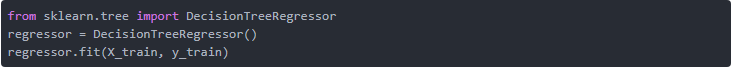
Виконаємо наступний код, щоб розділити наші дані на навчальні та тестові набори:



**Навчання та прогнозування**

Клас, який ми будемо використовувати тут, - це клас DecisionTreeRegressor.

Щоб навчити дерево, ми створимо екземпляр класу DecisionTreeRegressor і викличемо метод fit:



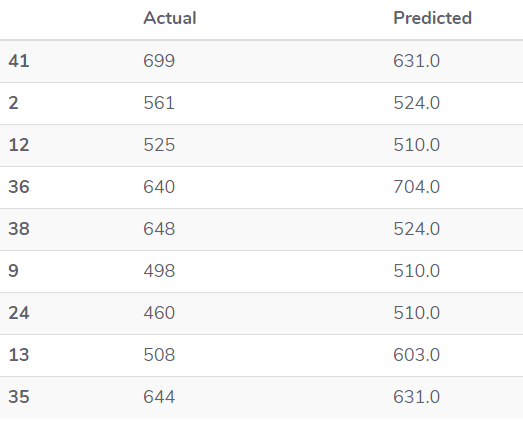
Для прогнозування тестового набору використаємо метод прогнозування:



А тепер давайте порівняємо деякі з наших передбачуваних значень з фактичними значеннями і побачимо, наскільки ми були точними:



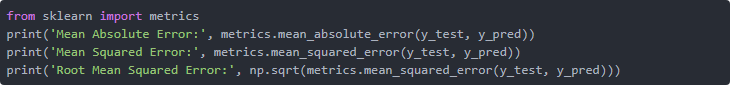
Результат буде:



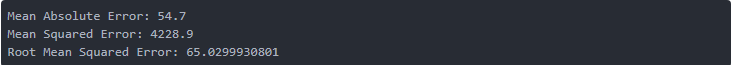
Тут важливо пам'ятати, що у іншому випадку порівняні записи можуть бути різними, залежно від розподілу тренувань та тестування. Оскільки метод train\_test\_split випадковим чином розділяє дані, ми, ймовірно, не матимемо однакових навчальних та тестових наборів.

**Оцінка алгоритму**

Для оцінки ефективності алгоритму регресії загальновживаними показниками є середня абсолютна похибка (mean absolute error), середня квадратична помилка (mean squared error) та середньоквадратична помилка (root mean squared error). Бібліотека Scikit-Learn містить функції, які можуть допомогти нам обчислити ці значення. Для цього скористайтеся цим кодом із пакету **metrics**:



Результат повинен виглядати приблизно так:



Середня абсолютна похибка нашого алгоритму становить 54,7, що становить менше 10 відсотків середнього значення всіх значень у стовпці "Transactions". Це означає, що наш алгоритм продемонстрував точну роботу.