Державний торговельно-економічний університет  
Факультет інформаційних технологій  
Кафедра інженерії програмного забезпечення та кібербезпеки

**ЛАБОРАТОРНА РОБОТА №3**

**З ДИСЦИПЛІНИ «Моделі і структури даних»**

**НА ТЕМУ: «Задача бінарної класифікації»**

**Виконавець**:

Студентка групи 4-4

Авєріна Наталія Ігорівна

Київ 2024

**Лабораторна робота №3**

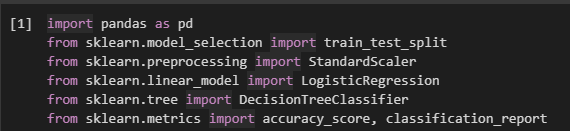
**Тема**: «**Задача бінарної класифікації**»

**Цілі**: розробити та навчити моделі логістичної регресії та дерева рішень. Оцінити продуктивність моделей за допомогою таких метрик як accuracy, precision, recall та f1-score. Зробити висновки згідно результатів порівняння моделей

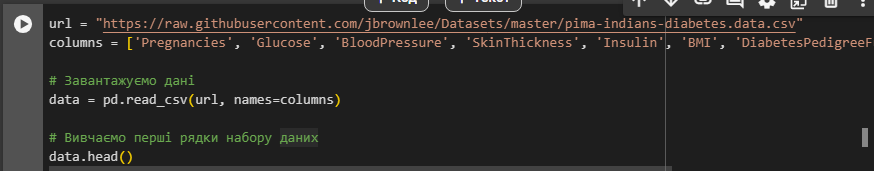
**Хід виконання:**

**Робота в Google Colab:** <https://colab.research.google.com/drive/1f-UPJXsJ49CfWDi-vTYpIrPcgnswUnfr?usp=sharing>

1. Завантажуємо необхідні бібліотеки



1. Завантажуємо дані



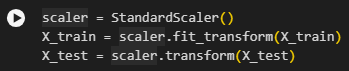
1. Розділяємо дані на ознаки та цільову змінну («Outcome»). Користуємося методом *drop* для того, щоб отримати лише ознаки шляхом видалення стовпчику цільової змінної.



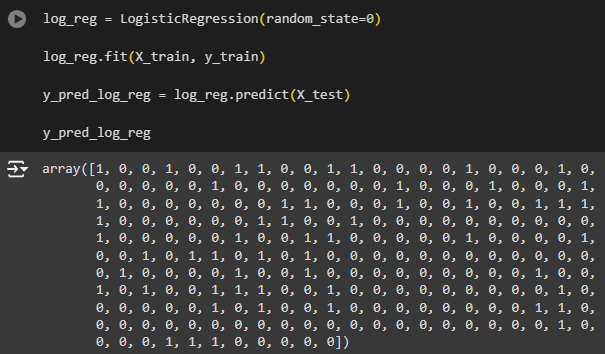
1. Розділяємо дані на тренувальну та тестову вибірку за допомогою функції *train\_test\_split* з бібліотеки *sklearn*.



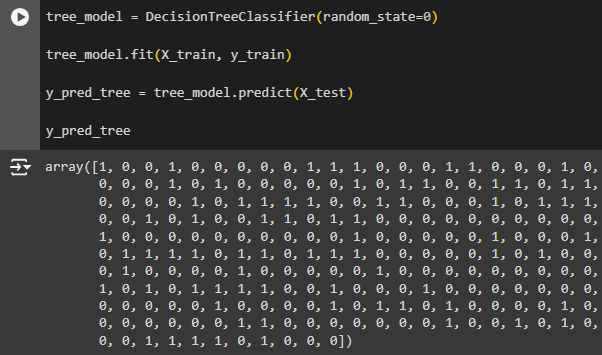
1. Масштабуємо дані. Використовуємо *StandardScaler*. *fit\_transform* підганяє статистики (середнє і стандартне відхилення) на тренувальних даних і одразу застосовує масштабування. *transform* застосовує ті ж самі статистики до тестових даних, не змінюючи їх. Це важливо, щоб уникнути витоку даних (data leakage).



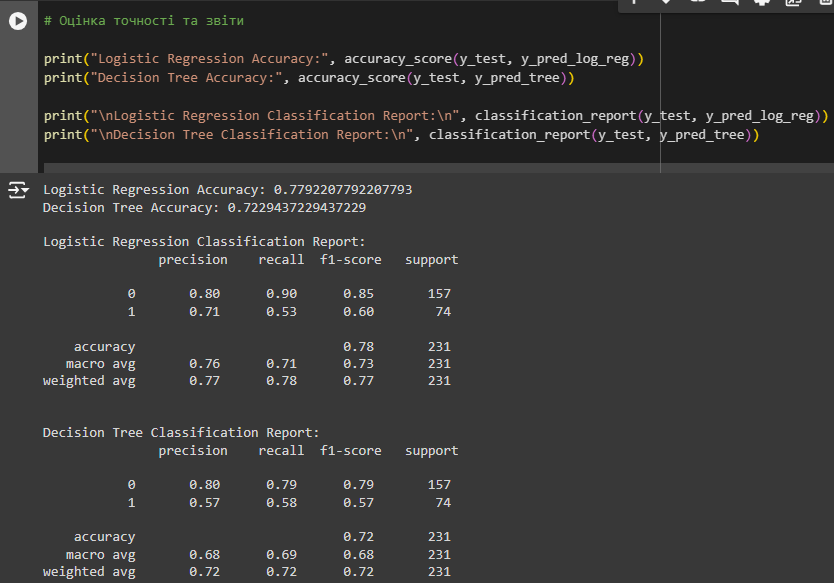
1. Визначаємо модель логістичної регресії (як екземпляр класа *LogisticRegression()*). Тренуємо модель на відмасштабованих навчальних даних та робимо прогнози *y\_pred\_log\_reg* на тестових відмасштабованих даних.



1. Визначаємо модель логістичної регресії (як екземпляр класа *DecisionTreeClassifier()*), тренуємо модель на відмасштабованих навчальних даних та робимо прогнози *y\_pred\_tree* на тестових відмасштабованих даних.



1. Виводимо оцінку точності та звіти.



**Висновки**:

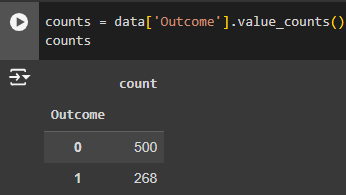
Точність логістичної регресії складає 77.9%, а дерева рішень – 72.3%. Згідно цих даних можна зробити висновок, що для цього датасету логістична регресія є більш ефективною моделлю.

Але для нашого датасету більш важливо правильно визначати, що людина не має діабету, адже набагато гірше, якщо людині з діабетом скажуть, що в неї його немає (FN), аніж людині без діабету скажуть, що в неї діабет (FP). Тому важливіше мінімізувати FN, щоб зменшити ризик пропуску реальних випадків діабету.

Закцентуймо увагу на значенні Recall (чутливість) класу 1. Значення дорівнює 53%. Recall показує, наскільки модель "чутлива" до виявлення всіх справжніх позитивних випадків. Високий показник recall означає, що модель пропускає дуже мало випадків діабету (низька кількість FN). В нашому випадку модель пропускає половину випадків, що є доволі поганим показником. В свою чергу модель Decision tree дала кращий показник – 58%.

Зменшення FP також важливе, оскільки помилкове виявлення діабету може призвести до непотрібних стресів та лікування пацієнтів, у яких діабет відсутній. Це стосується precision для класу 1, оскільки він показує, наскільки точні передбачення позитивних випадків. Логістична регресія дає кращі результати precision для класу 1 (71%), аніж дерево рішень (57%).

F1-score в нашому випадку особливо корисний, адже дані мають дисбаланс між класами (значно більше випадків без діабету (клас 0) ніж із діабетом (клас 1)).



F1-score моделі логістичної регресії дорівнює 60%, а дерево рішень 57%. Це свідчить про те, що логістична регресія має кращий баланс між точністю та чутливістю.

Логістична регресія показує кращі результати в порівнянні з деревом рішень у більшості метрик, особливо в точності і F1-score для обох класів. Проте дерево рішень демонструє вищу чутливість для класу 1, що є важливим у контексті медичного прогнозування, оскільки правильно виявити людей з діабетом критично важливо. Залежно від цілей моделі, можливо, варто розглянути комбінацію обох моделей або налаштувати їх для досягнення кращих результатів.

**Контрольні питання:**

1. Розкажіть про модель логістичної регресії (можна без формул, описово). Яку задачу вона розв'язує: регресії чи класифікації?

Логістична регресія – це статистичний метод, який використовується для бінарної класифікації, тобто для прогнозування ймовірності належності спостереження до одного з двох класів. Якщо ймовірність перевищує певний поріг (зазвичай 0.5), спостереження класифікується як клас 1, в іншому випадку — як клас 0. Кожне спостереження має бути незалежним один від одного, тобто немає кореляції між будь-якими вхідними змінними. Також у наборі даних не має бути викидів.

У рамках машинного навчання логістична регресія належить до сімейства керованих моделей машинного навчання.

Логістична регресія широко застосовується в різних сферах, таких як медицина (для прогнозування захворювань), фінанси (для оцінки кредитного ризику), маркетинг (для сегментації клієнтів) та багатьох інших.

1. В чому сутність дерев рішень?

Дерева рішень — це метод машинного навчання, який використовується для класифікації та регресії. Вони представляють собою графічну модель, що складається з вузлів (рішень) і гілок (розгалужень), які ведуть до результатів або класів. Кожен вузол у дереві відповідає певній ознаці (характеристиці), за якою здійснюється розподіл даних. Процес прийняття рішень триває доти, поки не досягнуть листового вузла, який відображає остаточний клас або значення. Вони можуть обробляти як числові, так і категоріальні дані та добре справляються з нетиповими розподілами даних. Однак, дерева рішень можуть бути схильні до перенавчання, тому їх часто комбінують з іншими методами.

1. Розкажіть про метрики якості для задачі бінарної класифікації, про які ви знаєте

Основною метрикою є accuracy, яка показує відсоток правильних прогнозів серед усіх випадків.   
Classification error — це метрика, яка вказує на частку помилкових класифікацій у моделі. Вона вимірює, наскільки часто модель неправильно класифікує дані.  
Precision вимірює частку правильних позитивних прогнозів серед усіх позитивних прогнозів, вказуючи, наскільки точно модель визначає позитивні випадки. Specificity — навпаки, метрика, яка вимірює здатність моделі правильно визначати негативні випадки.  
Recall (чутливість) оцінює частку правильних позитивних прогнозів серед усіх реальних позитивних випадків, що важливо, коли критично виявляти позитиви.   
F1-score є гармонійним середнім між precision та recall, забезпечуючи збалансовану оцінку, коли потрібно враховувати обидві метрики.   
ROC-AUC (площа під кривою ROC) відображає здатність моделі відрізняти класи, враховуючи всі можливі пороги, і корисна для оцінки продуктивності моделі в умовах незбалансованих класів.