Uczenie Maszynowe - Projekt Wstępny

Bartosz Psik 325211

Alicja Jonczyk 325165

1. Opis projektu

Celem projektu jest implementacja oraz porównanie technik oceny klasyfikacji na przykładzie dwóch rzeczywistych zestawów danych, dostępnych w repozytorium UCI Machine Learning Repository. Pierwszy zestaw dotyczy klasyfikacji guzów piersi jako złośliwych lub łagodnych (*Breast Cancer Wisconsin Dataset*), natomiast drugi — dotyczy klasyfikacji ocen studentów w skali 0 - 20 (*Student Performance*)

Projekt koncentruje się nie tylko na budowie klasyfikatorów, lecz przede wszystkim na **dokładnej ocenie ich jakości** przy użyciu popularnych metryk oraz wizualizacji.

2. Opis wykorzystanych algorytmów

Metody oceny klasyfikacji

W projekcie skoncentrujemy się na implementacji i analizie kilku popularnych metryk oceny skuteczności klasyfikatorów. Każda z nich pozwala spojrzeć na działanie modelu z nieco innej perspektywy:

- **Accuracy (dokładność)**

  To odsetek poprawnie sklasyfikowanych przypadków względem całkowitej liczby obserwacji. Jest prostą miarą skuteczności, ale bywa myląca w przypadku niezrównoważonych klas.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

- **Precision (precyzja)**

  Mierzy, jaki procent wszystkich przypadków zaklasyfikowanych jako pozytywne był rzeczywiście pozytywny.

$$\text{Precision} = \frac{TP}{TP + FP}$$

- **Recall (czułość, TPR)**

  Określa, jaki procent rzeczywistych pozytywnych przypadków został poprawnie wykryty. Jest szczególnie ważna w dziedzinach, gdzie istotne jest wychwycenie wszystkich przypadków np. choroby.

$$\text{Recall} = \frac{TP}{TP + FN}$$

- **F1 Score**

  Jest średnią harmoniczną precyzji i czułości. Balansuje obie miary i daje bardziej zrównoważony obraz skuteczności klasyfikatora, szczególnie przy niezbalansowanych danych.

$$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

- **Macierz pomyłek (Confusion Matrix)**

  Graficzna reprezentacja skuteczności klasyfikatora. Pokazuje liczbę poprawnych i błędnych klasyfikacji dla każdej klasy (TP, TN, FP, FN), co umożliwia dokładną analizę błędów modelu.

- **Krzywa ROC (Receiver Operating Characteristic)**

  Pokazuje zależność między czułością (TPR) a odsetkiem fałszywych alarmów (FPR) przy różnych progach decyzyjnych. Powierzchnia pod krzywą (AUC) jest syntetyczną miarą skuteczności modelu – im bliżej 1, tym lepszy klasyfikator.

- **Walidacja krzyżowa (Cross Validation)**

Technika oceny jakości modelu, która polega na wielokrotnym dzieleniu danych na zbiór treningowy i testowy. Najpopularniejsza forma to **K-fold cross validation**, gdzie dane dzielone są na *K* równych części (foldów). W każdej iteracji jeden fold służy jako zbiór testowy, a pozostałe jako treningowe — aż każdy fold zostanie raz użyty jako testowy.

Na końcu obliczamy średnią wartość metryk (accuracy, F1 itp.) ze wszystkich foldów, co daje bardziej stabilną i rzetelną ocenę modelu niż pojedynczy podział danych.

- **Własna metryka**

  Na chwilę obecną jeszcze nie mamy pomysłu na stworzenie własnej metryki, aczkolwiek wraz z pracą nad projektem może ulec to zmianie.

Klasyfikatory

Aby uzyskać pełniejszy obraz skuteczności metryk i ich zachowania w różnych warunkach, planujemy przetestować różne klasyfikatory, dostępne w popularnych bibliotekach uczenia maszynowego (np. `scikit-learn`). Wśród rozważanych algorytmów znajdą się między innymi:

- **Regresja logistyczna (Logistic Regression)** – prosty, liniowy model dobrze sprawdzający się w klasyfikacji binarnej
- **Drzewa decyzyjne (Decision Trees)** – modele oparte na regułach decyzyjnych, intuicyjne i łatwe do wizualizacji,
- **Las losowy (Random Forest)** – złożony model oparty na wielu drzewach, oferujący większą stabilność i dokładność,
- **SVM (Support Vector Machine)** – model maksymalizujący margines pomiędzy klasami, skuteczny przy małych, dobrze przygotowanych danych,
- S**ieci neuronowe (MLPClassifier)** – umożliwiają modelowanie nieliniowych zależności przy większych zbiorach danych.

Wybór różnych klasyfikatorów pozwoli nie tylko porównać ich skuteczność, ale również zaobserwować, jak różne metryki mogą lepiej lub gorzej oddawać rzeczywistą jakość klasyfikacji w zależności od charakterystyki modelu.

3. Szacunkowy plan eksperymentów

1. Walidacja własnych implementacji metryk dla klasyfikacji binarnej

Cel: Sprawdzić poprawność zaimplementowanych metryk przez porównanie ich z funkcjami z `scikit-learn` :

- Wytrenowanie różnych modeli klasyfikatorów dla obu zbiorów danych
- Obliczenie `accuracy`, `precision`, `recall`, `f1-score`:
  - własnoręcznie, przy pomocy naszej implementacji metryk do oceny
  - oraz za pomocą `sklearn.metrics`.

2. Walidacja graficznych metryk implementacji

Cel: Zweryfikować poprawność oraz jakość działania własnych implementacji **macierzy pomyłek** oraz **krzywej ROC**, poprzez porównanie z odpowiadającymi im funkcjami z biblioteki `scikit-learn`.

- Wytrenowanie różnych klasyfikatorów (np. Logistic Regression, Random Forest, SVM) na obu zbiorach danych.
- **Macierz pomyłek:**
  - Obliczenie liczby TP, TN, FP, FN z przewidywań modelu przy pomocy własnej implementacji macierzy.
  - Narysowanie graficznej wersji macierzy pomyłek
  - Porównanie z `sklearn.metrics.confusion_matrix`
- **Krzywa ROC:**
  - Dla modeli zwracających prawdopodobieństwo (np. `predict_proba()`), obliczenie punktów (FPR, TPR) dla różnych progów decyzyjnych.
  - Narysowanie krzywej ROC na podstawie własnych obliczeń oraz policzenie pola pod krzywą (AUC) metodą trapezów.
  - Porównanie z `sklearn.metrics.roc_curve` i `roc_auc_score`.

- Graficzna prezentacja obu krzywych (własnej i z `scikit-learn`) na jednym wykresie – analiza rozbieżności i potwierdzenie poprawności implementacji.

3. Czas obliczeń własnych metryk vs scikit-learn

Cel: Sprawdzić wydajność własnoręcznie zaimplementowanych metryk

- Użycie `time` lub `timeit` do zmierzenia czasu:
  - naszych funkcji do precision / recall / F1,
  - odpowiedników z `sklearn`.
- Porównanie wyników

4. Walidacja krzyżowa (Cross Validation)

Cel: Ocenić stabilność oraz wiarygodność metryk przy użyciu wielokrotnego podziału danych (K-fold cross validation), a także porównać zachowanie klasyfikatorów na różnych foldach.

- Porównać własną implementację, z gotową implementacją z `sklearn`
- Zastosowanie K-fold cross-validation (np. K=5) dla wybranych klasyfikatorów.
- Porównanie metryk (`accuracy`, `precision`, `recall`, `f1-score`) uzyskanych:
  - przy pojedynczym podziale danych (train/test split),
  - oraz w ramach K-fold cross-validation.
- Analiza różnic i stabilności wyników: czy metryki znacząco się różnią między foldami?
- Porównanie skuteczności różnych klasyfikatorów w ujęciu przekrojowym (średnie wartości metryk z CV).
- Wizualizacja rozrzutu wyników przy pomocy wykresów pudełkowych (boxplotów) dla wybranych metryk (np. F1-score), w celu oceny równomierności działania modeli.

5. Eksperyment: mikro vs makro vs weighted dla klasyfikacji wieloklasowej

Cel: Pokazać różnicę w agregacji metryk przy więcej niż dwóch klasach

- Użycie danych wieloklasowych (Student Performance).
- Porównanie `precision`, `recall`, `F1` gotowych implementacji oraz własnoręcznj implementacji w wersjach:
  - `macro` – każda klasa traktowana równo,
  - `micro` – globalna suma TP, FP, FN,
  - `weighted` – ważone proporcją występowania klas.

4. Wykorzystane zbiory danych

W naszym projekcie wykorzystamy dwa zbiory danych. Pierwszy, zaproponowany w poleceniu zadania - dane dotyczące raka piersi oraz drugi - dane dotyczące wyników w nauce uczniów licealnych.

Analiza zbiorów

Przy wyborze drugiego zbioru danych wzięliśmy pod uwagę, ile instancji ma dany zbiór, ile cech (features) oraz czy zakłada klasyfikację binarną czy wieloklasową. Zależało nam, by przetestować nasze klasyfikatory dla różnych zbiorów. Następnie przeanalizowaliśmy, czy wybrane zbiory są zbalansowane.

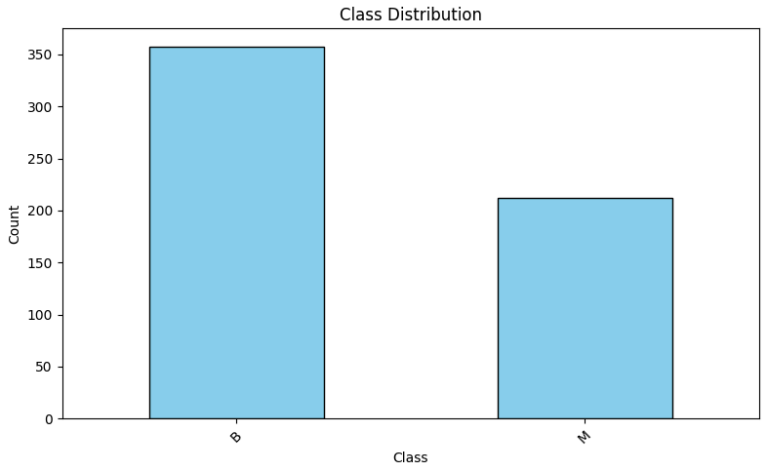
1. **Breast Cancer Wisconsin (Diagnostic)**

http://archive.ics.uci.edu/dataset/17/breast+cancer+wisconsin+diagnostic

Pierwszy zbiór danych pozwala na klasyfikację binarną. Po wczytaniu danych zauważyliśmy, że nie mają wiersza z opisem atrybutów, więc dodaliśmy pierwszy wiersz, korzystając z nazw i opisów z powyżej podlinkowanej strony. Kolumna “Diagnosis” zawiera informację, czy rak piersi jest M = malignant - złośliwy, czy B

= benign - łagodny. Zbiór zawiera 569 instancji i 30 cech, opisujących właściwości komórek guza, takie jak promień, obwód, powierzchnia, symetria czy nieregularność kształtu, wyodrębnione z obrazów mikroskopowych. Cechy te występują jako statystyki średnie, błędy standardowe i wartości skrajne, umożliwiając analizę struktury i tekstury komórek nowotworowych.

Przejrzeliśmy zbalansowanie klas:



2. **Student Performance**

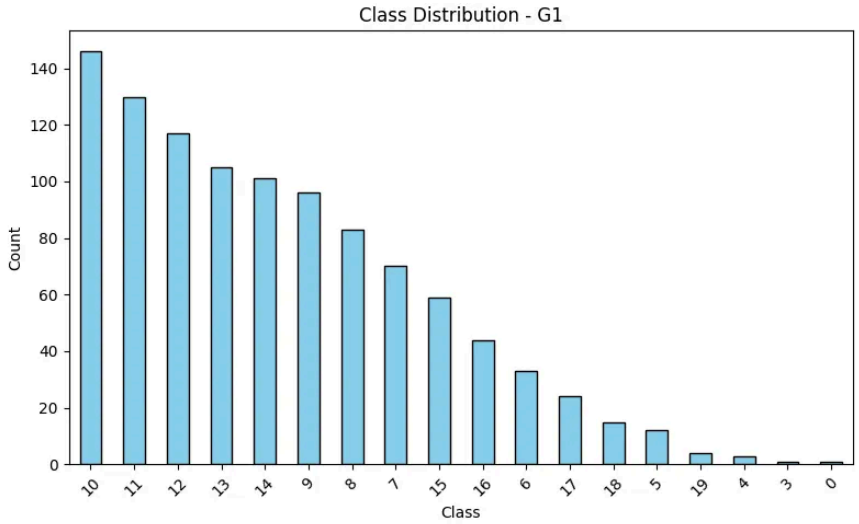
http://archive.ics.uci.edu/dataset/320/student+performance

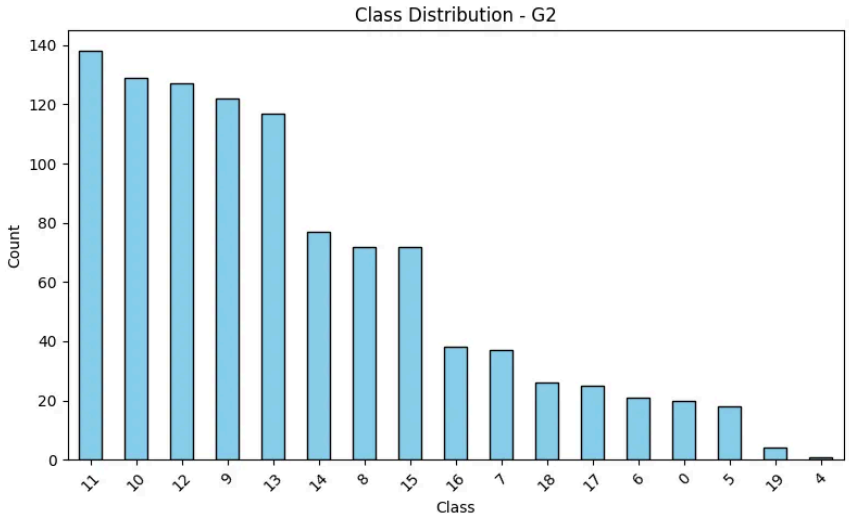
Drugi zbiór zakłada klasyfikację wieloklasową, gdzie wyniki studenta mieszczą się w skali 0-20. Zbiór ten składa się z dwóch podzbiorów: student-mat i student-por, które zawierają oceny studentów odpowiednio z matematyki i z języka portugalskiego. Ponadto, zbiór zawiera informacje demograficzne, edukacyjne i społeczne, takie jak wiek, płeć, status rodzinny, czas nauki, frekwencja na zajęciach czy nawyki alkoholowe. Pracę nad tymi zbiorami rozpoczęliśmy poprzez połączenie ich w jeden większy zbiór, który ostatecznie zawierał 1044 instancje. Ten zbiór pozwala również na klasyfikację różnych stadiów oceny uczniów: po pierwszym semestrze, po drugim oraz oceny końcowej.

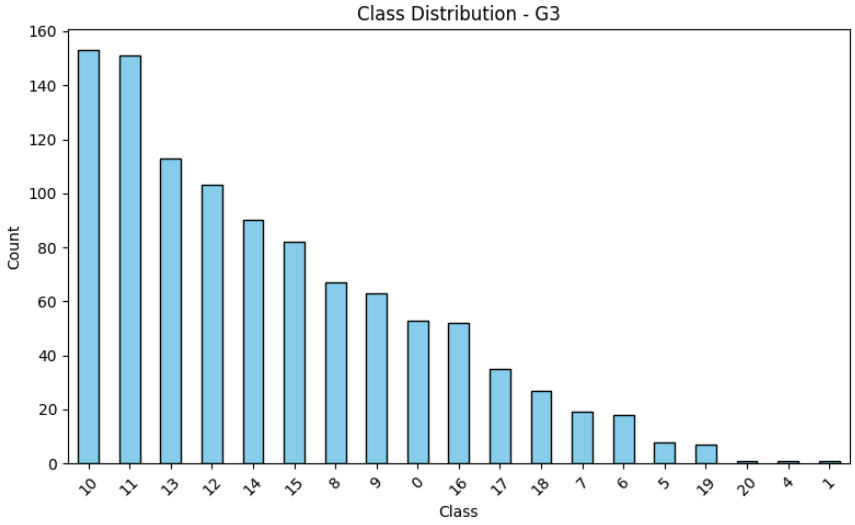
G1	first period grade

G2	second period grade
G3	final grade

Przejrzeliśmy zbalansowanie klas dla G1, G2 i G3.







Pierwsze przetwarzanie i testy naszych zbiorów danych umieściliśmy na naszym GitHubie:

https://github.com/psiku/UMA---Techniki-oceny-klasyfikacji.git