Politechnika Rzeszowska
Katedra Informatyki i Automatyki

Sprawozdanie z przedmiotu
Analiza danych w językach R i Python

TEMAT PROJEKTU

Prowadzący:

Krzysztof Kołodziej, L01,
169562@stud.prz.edu.pl
Jakub Duda, L01,
169532@stud.prz.edu.pl

Tutaj znajduje się informacja o tym w jaki sposób należy dostarczyć kolejne etapy sprawozdania prowadzącym.

Kolejne etapy sprawozdania należy dostarczyć prowadzącemu w podanych niżej terminach za pomocą strony **sprawozdania.kia.prz.edu.pl**. Należy tam w pierwszej kolejności założyć konto (jeżeli jeszcze takowego nie ma) i dołączyć do odpowiedniej grupy projektowej z przedmiotu. Każdy kolejny etap sprawozdania należy dopisać do pliku zawierającego poprzedni, już wysłany, tak, aby ostatnie sprawozdanie było kompletne. Jeśli kolejny etap wymaga poprawek w treści poprzedniego są one dozwolone. Przed każdorazowym wysłaniem sprawozdania należy odświeżyć pole spisu treści!

Każde sprawozdanie ma być wysłane w załączniku maila z tematem:

`ADRP<numer_rozdziału>_L<numer_grupy>_<I1>.<Nazwisko1>_<I2>.<Nazwisko2>`

na przykład:

`ADRP_R1_L01_J.Kowalski_M.Nowak`

Ostatnie sprawozdanie powinien zawierać też załącznik z implementacją projektu, plikami danych opisującymi problemy rozważane w sprawozdaniu oraz plikami wyników jeśli takie pliki istnieją.

Każdy wykonany projekt trzeba obronić podczas osobistej rozmowy grupy projektowej z prowadzącym w wyznaczonym terminie. Na tą rozmowę należy być przygotowanym do zaprezentowania działającego projektu. Każdy członek grupy zostanie oceniony osobno na podstawie jego wkładu w wykonanie projektu.

Terminy nadsyłania kolejnych sprawozdania:

Rozdział sprawozdania	**Termin**
Rozdział numer 1.	28 X
Rozdział numer 2.	25 XI
Rozdział numer 3. etapów	23 XII

Pozostałe rozdziały należy wypełnić podczas pracy nad zasadniczymi trzema wymienionymi wyżej. Niewywiązanie się z powyższych terminów będzie skutkowało drastycznym obniżeniem oceny z przedmiotu. Termin oddawania i obrony projektów zostanie ustalona ze starostą roku.

Spis treści

1. Opis problematyki klasyfikacji danych

1.1 Wstęp

Klasyfikacja danych jest kluczowym elementem eksploracji danych, czyli procesu automatycznego odkrywania nieznanych wcześniej zależności i wzorców w zbiorach danych. Proces ten pozwala na przekształcenie surowych danych w wartościowe informacje, które mogą wspierać podejmowanie decyzji w różnych dziedzinach. Informacje uzyskane dzięki klasyfikacji mogą być wykorzystywane m.in. do przewidywania zachowań czy generowania raportów. [2]

Klasyfikację można podzielić na: [2]

- **Klasyfikacja dwuklasowa**: Przypisuje obiekt do jednej z dwóch dostępnych klas (np. "ssak" lub "nie ssak").
- **Klasyfikacja wieloklasowa**: Przyporządkowuje obiekt do jednej z wielu klas (np. rozpoznanie gatunku zwierzęcia).

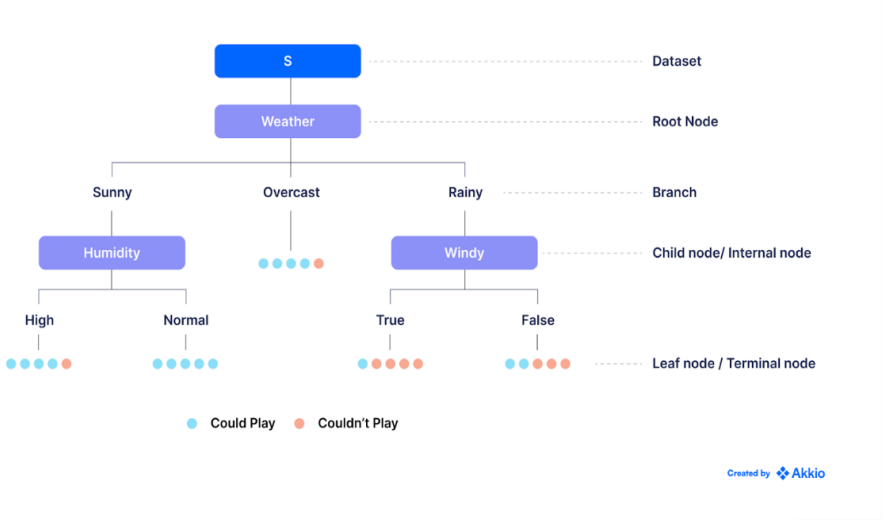
1.2 Etapy procesu klasyfikacji

Proces klasyfikacji można podzielić na dwa główne etapy. Pierwszy to **etap uczenia się**, w którym algorytm analizuje dane treningowe, aby nauczyć się wzorców i reguł, które później będą wykorzystywane do przypisywania nowych danych do odpowiednich klas. W tym etapie model buduje swoją wiedzę na podstawie dostarczonych informacji. Drugi to **etap predykcji**, w którym model, opierając się na wyuczonych wzorcach, dokonuje klasyfikacji nowych danych, przypisując je do odpowiednich kategorii. [2]

Klasyfikację można zaliczyć do metod **nadzorowanych**, ponieważ algorytm uczy się na wcześniej oznaczonych danych. Oznacza to, że każdy obiekt w zbiorze treningowym ma już przypisaną klasę, a zadaniem modelu jest nauczyć się, jak na podstawie cech obiektu przypisać go do odpowiedniej kategorii. W fazie uczenia algorytm otrzymuje dane wejściowe wraz z prawidłowymi wynikami (czyli z etykietami klas), zakładając, że są one prawidłowe, co pozwala mu wyciągać wnioski dotyczące przyszłych klasyfikacji. [2]

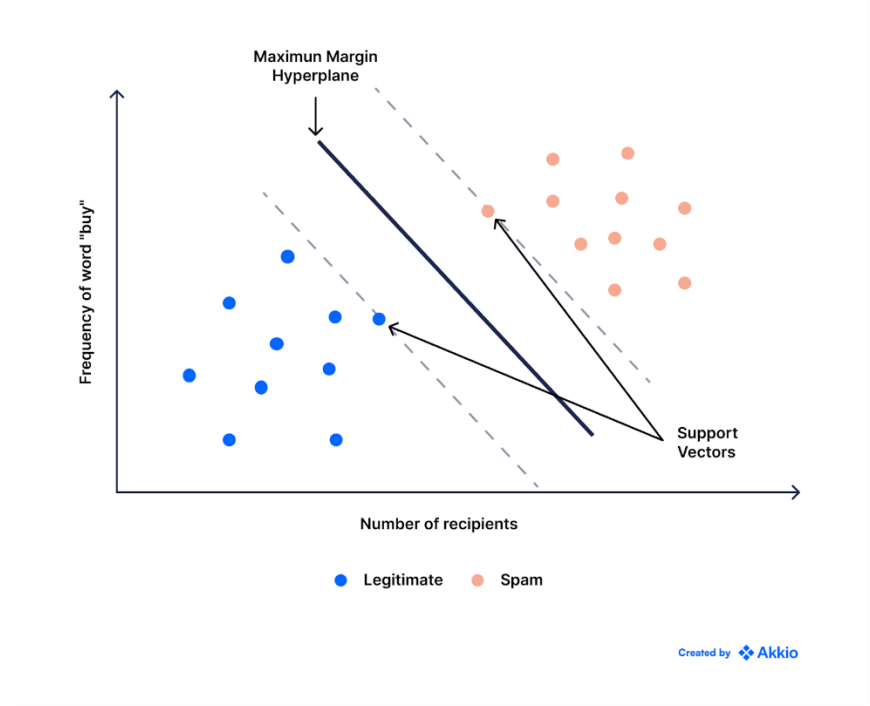
Przykładami algorytmów stosowanych w klasyfikacji nadzorowanej są:

- **Drzewa decyzyjne**: Bardzo powszechny sposób przedstawiania i wizualizacji możliwych wyników decyzji lub działania na podstawie prawdopodobieństw. Każde drzewo jest hierarchiczną reprezentacją przestrzeni wyników, w której każdy węzeł reprezentuje działanie lub wybór, a liście reprezentują stany wyniku. Są łatwe w zaimplementowanie i szczególnie przydatne w sytuacjach, gdy istotna jest przejrzystość procesu decyzyjnego oraz łatwość interpretacji wyników [1]



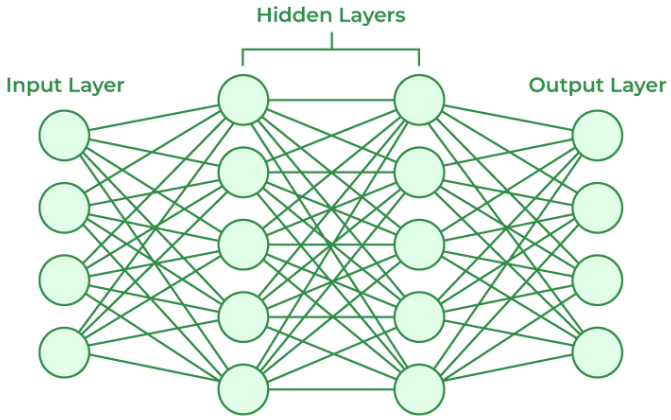
Rysunek 1 - algorytm drzewa decyzyjnego [1]

- **Maszyna wektorów nośnych (SVM)**: Algorytm, który tworzy granicę decyzyjną między klasami, tak aby maksymalizować odległość między najbliższymi przykładami różnych klas. Jest szczególnie dobry w oddzielaniu podobnych danych. Jest ceniony za skuteczność w sytuacjach i stosowany w sytuacjach, gdy dane są trudne do rozdzielenia, ponieważ tworzy hiperplan maksymalnie oddzielający różne klasy, co zmniejsza ryzyko błędnej klasyfikacji. [1]



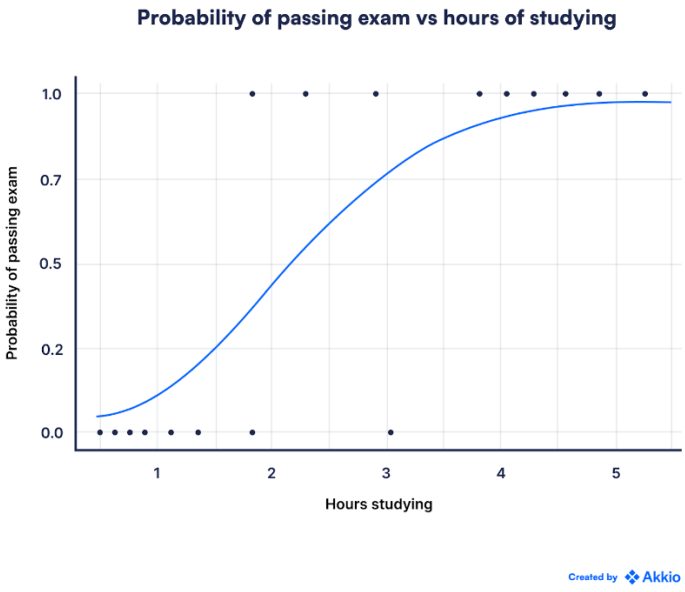
Rysunek 1 - algorytm SVM [1]

- **Sztuczne sieci neuronowe**: sama nazwa wskazuje, emulują biologiczne sieci neuronowe w komputerach. Sieci neuronowe działają poprzez trenowanie zestawu parametrów na danych. Parametry te są następnie używane do określania wyników modelu. Sieci mogą być jednokierunkowe lub rekurencyjne w których występuje sprzężenie zwrotne. Sztuczne sieci neuronowe mają szerokie zastosowanie w wielu dziedzinach, zwłaszcza tam, gdzie występuje duża złożoność danych lub trudność w wykrywaniu wzorców. Dzięki swojej zdolności do przetwarzania dużych ilości danych i "uczenia się" z nich, sieci neuronowe znajdują zastosowanie w takich obszarach, jak: przetwarzanie języka naturalnego, rozpoznawanie obrazów i wideo itp. [1]



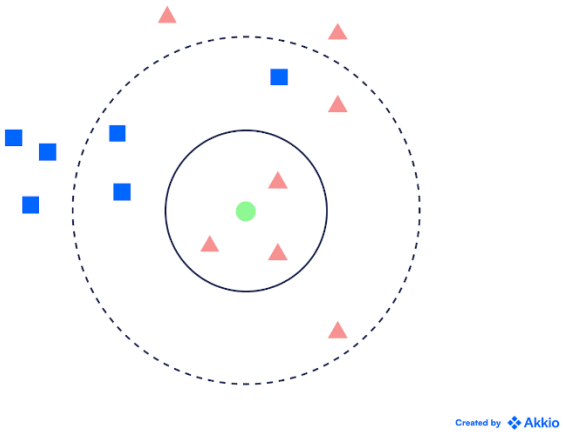
Rysunek 2 - sztuczne sieci neuronowe [3]

- **Regresja logistyczna**: W odróżnieniu od regresji liniowej, która prognozuje wartości ciągłe, regresja logistyczna stosuje funkcję sigmoidalną (logistyczną), aby wyniki mieściły się w zakresie od 0 do 1, czyli odzwierciedlały prawdopodobieństwo wystąpienia danej klasy. Jest często stosowana w klasyfikacji binarnej, np. do diagnozy chorób (zdrowy vs chory), klasyfikacji wiadomości (spam vs nie-spam) czy przewidywania wyników (sukces vs porażka). [1]



Rysunek 3 - regresja logistyczna [1]

- **K-Najbliższych sąsiadów (KNN)**: Jego zasada działania opiera się na założeniu, że obiekty o podobnych cechach będą należeć do podobnych klas lub mieć zbliżone wartości. W klasyfikacji KNN, aby przewidzieć klasę nowego punktu, algorytm wybiera k najbliższych sąsiadów tego punktu (na podstawie metryki odległości, np. euklidesowej) i przypisuje mu klasę, która występuje najczęściej w tej grupie sąsiadów. Algorytm KNN znajduje szerokie zastosowanie w różnych dziedzinach, szczególnie tam, gdzie ważne jest dopasowanie nowych danych do wzorców na podstawie istniejących danych historycznych np. systemy rekomendacji, rozpoznawanie obrazów i wideo itd. [1]



Rysunek 4 - algorytm K-Najbliższych sąsiadów [1]

- **Naiwny klasyfikator Bayesa**: Algorytm probabilistyczny, który zakłada niezależność cech. Klasyfikatory Naiwnego Bayesa są często używane w klasyfikacji tekstów, ponieważ łatwo jest obliczyć prawdopodobieństwo na podstawie częstości, a tekst zwykle ma dużą liczbę cech (np. pojedyncze tokeny w słowach). [1]

Metody te wykorzystują wiedzę z fazy uczenia, aby skutecznie dokonywać predykcji w fazie wyznaczania wyniku, przypisując nowe dane do właściwych klas.

1.3 Metody stosowane do poprawności wiarygodności i dokładności budowanych modeli

Istnieje kilka kluczowych metod stosowanych w uczeniu maszynowym do poprawy wiarygodności i dokładności modeli. Te techniki pozwalają na ocenę skuteczności modelu i minimalizację błędów związanych z przeuczeniem lub niedouczeniem. Oto kilka z nich:

- **Walidacja krzyżowa z pozostawieniem jednej wartości (LOOCV)**: W tej metodzie przeprowadzamy trening na całym zestawie danych, ale pozostawiamy tylko jeden punkt danych z dostępnego zestawu danych, a następnie iterujemy dla każdego punktu danych. W LOOCV model jest trenowany na $N-1$ próbki i testowane na jednej pominiętej próbce, powtarzając ten proces dla każdego punktu danych w zestawie danych. Wykorzystujemy wszystkie punkty danych, co sprawia, że jest ona mało obciążona błędem. Testujemy jednocześnie tylko jeden punkt danych. Jeśli punkt danych jest wartością odstającą, może to prowadzić do większej zmienności. Metoda ta zajmuje również dużo czasu wykonania, ponieważ iteruje się przez „liczbę punktów danych" razy. [4]
- **Walidacja krzyżowa K-Fold:** Polega na dzieleniu zbioru danych na k podzbiorów (znanych jako folds), a następnie przeprowadzeniu treningu na wszystkich podzbiorach, pozostawiając jeden (k-1) podzbiór do oceny wytrenowanego modelu. Jej zaletą jest możliwość uzyskania bardziej wiarygodnych i stabilnych wyników oceny modelu, zwłaszcza na ograniczonym zbiorze danych [4]
- **Stratyfikowana walidacja krzyżowa:** Jest to technika stosowana w uczeniu maszynowym, aby zapewnić, że każdy etap procesu walidacji krzyżowej zachowuje ten sam rozkład klas, co cały zestaw danych. Jest to szczególnie ważne w przypadku niezrównoważonych zestawów danych, w których pewne klasy mogą być niedoreprezentowane. W tej metodzie:
  1. Zbiór danych podzielony jest na k części, przy zachowaniu proporcji klas w każdej części.
  2. Podczas każdej iteracji jeden z elementów jest używany do testowania, a pozostałe do trenowania.
  3. Proces powtarza się k razy, przy czym każdy podzbiór służy jako zbiór testowy dokładnie raz.

  Stratyfikowana walidacja krzyżowa jest niezbędna w przypadku problemów klasyfikacji, w których zachowanie równowagi rozkładu klas ma kluczowe znaczenie dla prawidłowego uogólniania modelu na niewidziane wcześniej dane. [4]
- **Walidacja holdout**: W tej metodzie przeprowadzamy trening na 50% danego zestawu danych, a pozostałe 50% jest używane do celów testowych. To prosty i szybki sposób na ocenę modelu. Główną wadą tej metody jest to, że przeprowadzamy trening na 50% zestawu danych, może się zdarzyć, że pozostałe 50% danych zawiera pewne ważne informacje, które pomijamy podczas trenowania naszego modelu. [4]

2. Charakterystyka danych

Ten rozdział ma zawierać opis danych dla których będą przeprowadzane eksperymenty. Należy wyszukać dane do procesu klasyfikacji (dobrze by było, aby ich objętość w postaci liczby rekordów była jak największa - rzędu od kilkuset do kilku tysięcy rekordów) i wskazać źródło tych danych. Należy przedstawić informacje charakteryzujące dane (np. średnia, minimum, maksimum, mediana itp.). Przygotować stosowne wykresy (histogramy, wykresy pudełkowe) prezentujące rozkład wartości poszczególnych atrybutów. Jeżeli istnieje konieczność, należy wykonać proces przygotowania danych do budowy modeli klasyfikacji np. czyszczenie, normalizacja, usuwanie wartości pustych itp. Całość należy udokumentować fragmentami kodu w języku Python lub R.

3. Opis eksperymentów i opracowanie uzyskanych wyników

Ten rozdział powinien zawierać wyniki eksperymentów wykonanych dla pozyskanych danych za pomocą przynajmniej 3 wybranych metod klasyfikacji. Powinny zostać przedstawione wyniki budowy modeli klasyfikacji dla różnych parametrów wybranych algorytmów oraz porównanie uzyskanych wyników w formie tabelarycznej i graficznej. Kolejne prezentowane eksperymenty powinny prowadzić do poprawienia parametrów algorytmów użytych w następnych eksperymentach. Uzyskane wyniki oraz dyskusję ich poprawności, dokładności, czasu uzyskania oraz parametrów i ich wpływu na wyniki należy umieścić w tym rozdziale. Całość należy udokumentować fragmentami kodu w języku Python lub R.

Podsumowanie

Tutaj należy umieścić podsumowanie wykonanych prac, własne wnioski dotyczące rozwiązywanego problemu klasyfikacji, metod oraz ich parametrów, zastosowanych narzędzi.

Bibliografia

[1] Akkio.com. (2024, 1 8). Pobrano z lokalizacji https://www.akkio.com/post/5-types-of-machine-learning-classification-algorithms
[2] Fatyga Piotr, P. R. (2023, 11 14). Klasyfikacja danych - przegląd wybranych metod, .
[3] geekforgeeks.org. (2024, 8 7). Pobrano z lokalizacji https://www.geeksforgeeks.org/artificial-neural-networks-and-its-applications/
[4] geeksforgeeks.org. (2024, 8 7). Pobrano z lokalizacji https://www.geeksforgeeks.org/cross-validation-machine-learning/