**Projekt**

**z przedmiotu**

**Język Python w analizie danych**

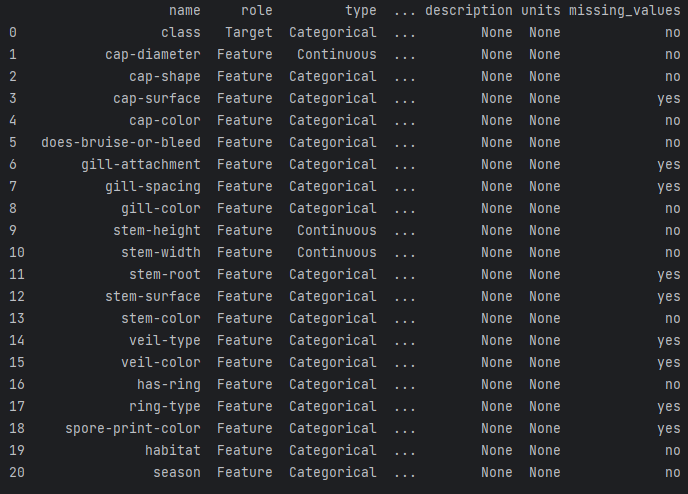
**Temat projektu: Porównanie metod imputacji oraz algorytmów uczenia maszynowego na podstawie klasyfikacji**

**Prowadzący: mgr inż. Patryk Organiściak**

|  |  |
| --- | --- |
| **Imię i nazwisko: inż. Kamil Hansel** |  |
| **Nr albumu: 166728** |  |
| **Rok studiów: 1 FS0-DU** |  |
| **Grupa projektowa: P2** |  |
| **Data oddania projektu: 20.01.2025** |  |

## 1. Opis zbioru danych

Dane wykorzystane w ramach tego projektu zostały pobrane z repozytorium Uniwersytetu Kalifornijskiego w Irvine. Zbiór dotyczy klasyfikacji grzybów decydując czy dany grzyb jest jadalny czy trujący. Zbiór danych zawiera 61068 wierszy oraz 20 kolumn (21 z cechą docelową) Parametry cech wyglądają następująco:



Rysunek - Informacje dotyczące cech: nazwa, rola, typ oraz indeks pustych wartości

Link do dostępu: https://archive.ics.uci.edu/dataset/848/secondary+mushroom+dataset

## 2. Transformacja danych

Dane wstępnie zostały przemapowane za pomocą funkcji preprocessing.OrdinalEncoder() z pakietu sklearn. Aby zbiór mógł zostać poddany procesowi imputacji zmieniono typ kolumn zbioru na float. Następnie obliczono współczynniki pustych wartości w każdej kolumnie, tak aby można było wykluczyć atrybuty mało istotne. Wprowadzony próg dopuszczalności ustalono na 50%. Następnie wyrzucono kolumny, w których ponad połowa danych zawierała puste wartości oraz statycznie wprowadzono wektor\_pominiec który zostanie wykorzystany w funkcji wspomagającej jeden algorytm imputacji. Następnie podzielono zbiór danych na zbiór treningowy oraz zbiór testowy.

## 3. Imputacja danych

W projekcie wykorzystano 4 różne sposoby imputacji danych:

* Imputacja zastępująca puste wartości modą(dominantą) danej kolumny,
* imputacja zastępująca puste wartości medianą danej kolumny,
* imputacja za pomocą algorytmu IterativeImputer za pomocą którego wartości wprowadzone zostały następnie zaokrąglone do liczb całkowitych w kolumnach, które zawierają dane kategoryczne,
* imputacja za pomocą algorytmu KNNImputer.

Ponadto zaprogramowano opcję umożliwiającą pominięcie przeprowadzenia imputacji ze względu na fakt, że dwa z pięciu algorytmów uczenia maszynowego wykorzystanego w tym projekcie jest w stanie obsłużyć braki danych w kolumnach. Wszystkie sposoby imputacji zostały opatrzone w funkcje mierzące czas potrzebny na przetworzenie kodu.

## 4. Dopasowanie modeli uczenia maszynowego oraz przeprowadzanie predykcji

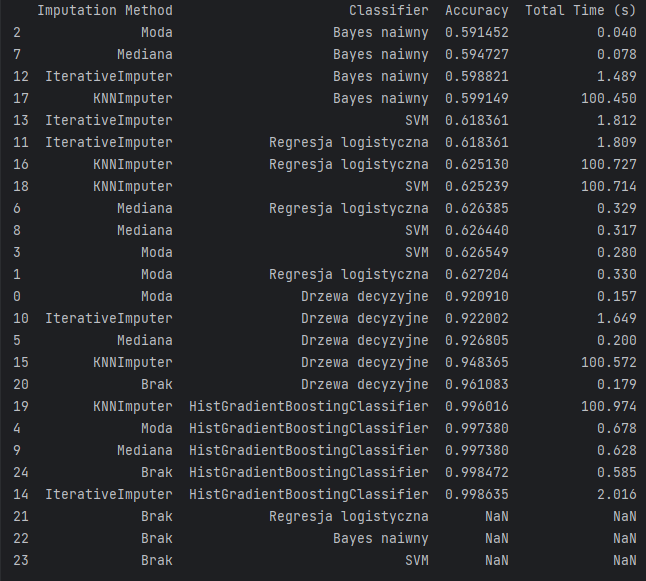
W projekcie wykorzystano 5 różnych algorytmów uczenia maszynowego:

* Drzewo decyzyjne
* Regresja logistyczna
* Naiwny klasyfikator bayesowski
* SVM
* HistGradientBoostingClassifier

Podobnie jak przy imputacji, wszystkie algorytmy uczenia maszynowego zostały opatrzone w funkcje mierzące czas potrzebny na przetworzenie kodu. Ponadto obliczono stopień dopasowania każdego modelu. Na koniec zebrano wszystkie wyniki i przedstawiono je w tabeli poniżej.

## 5. Podsumowanie wyników

Z podanych wyników można wywnioskować, że metody imputacji danych nie wpływają w znacznym stopniu na poziom predykcji, natomiast wpływają one w znaczniejszym stopniu na czas obliczeniowy. Algorytm KNNImputer pomimo parametru neighbours ustawionego na 1 potrzebował aż 100 sekund na przeprowadzenie imputacji. Ciekawym spostrzeżeniem jest fakt, iż największy stopień dopasowania występuje przy dwóch algorytmach które są w stanie obsłużyć braki wartości. Tabela została posortowana wobec parametru ‘Accuracy’.



Rysunek - Tabela podsumowująca wyniki imputacji oraz uczenia maszynowego