Instytut Informatyki
Wydział Nauk Ścisłych i Technicznych
Uniwersytet Rzeszowski

Przedmiot:

Hurtownie danych

Dokumentacja projektu:

Hurtownia bazy grzybów

Wykonał: Mateusz Hołyszko

Prowadzący: mgr inż. Adam Szczur

Rzeszów 2025

Spis treści

1. Temat i cel projektu

Tematem projektu jest zaprojektowanie i wdrożenie hurtowni danych dla zbioru danych „Mushroom", opisującego cechy 23 gatunków grzybów wraz z informacją o ich jadalności. W ramach realizacji projektu zostanie zbudowany proces ETL (Extract–Transform–Load), czyli sekwencja działań polegająca na wyodrębnieniu danych ze źródła (plik CSV), ich przekształceniu (m.in. standaryzacja wartości, kodowanie kolumn symbolicznych, uzupełnianie braków) oraz załadowaniu do właściwej struktury hurtowni danych. Dzięki temu wszystkie etapy od surowego pliku po gotową bazę analityczną będą zautomatyzowane i powtarzalne.

Celem projektu jest stworzenie wielowymiarowego modelu danych, który umożliwi szybkie wykonywanie analiz statystycznych oraz wspomoże zadania klasyfikacji (rozdzielenie grzybów jadalnych od trujących), eksploracji danych oraz generowanie intuicyjnych raportów i wykresów. W rezultacie użytkownicy otrzymają czytelne GUI, pozwalające na definiowanie parametrów zapytań OLAP i prezentację wyników w formie tabelarycznej lub graficznej, co ułatwi formułowanie wniosków na temat zależności między cechami morfologicznymi grzybów a ich jadalnością.

Techniczne aspekty projektu

1.1. Funkcjonalności aplikacji

1. **Wczytywanie i wyświetlanie danych**
   - Wczytywanie danych o grzybach z plików .data
   - Wyświetlanie danych w formie tabeli z opisowymi etykietami zamiast kodów
2. **Edycja danych**
   - Ręczna edycja pojedynczych komórek poprzez menu kontekstowe
   - Dodawanie nowych wierszy
   - Usuwanie istniejących wierszy
   - Automatyczna walidacja wprowadzanych wartości zgodnie ze schematem
3. **Statystyki (zakładka Statistics)**
   - Wyświetlanie podstawowych statystyk dla wybranej kolumny
   - Pokazywanie rozkładu wartości
   - Wizualizacja danych w formie wykresów
4. **Operacje na podzbiorach (zakładka Subset)**
   - Wybieranie wierszy poprzez indeksy (slice)
   - Filtrowanie po wartościach w kolumnach
   - Wybór konkretnych kolumn
   - Możliwość łączenia operacji (np. filtrowanie po wartościach, a następnie wybór kolumn)
   - Zapisywanie utworzonych podzbiorów do plików
5. **Klasyfikacja (zakładka Classify)**
   - Trenowanie klasyfikatora k-NN na wybranych cechach
   - Podział danych na zbiór treningowy i testowy
   - Wyświetlanie dokładności klasyfikacji
   - Możliwość zapisania i wczytania wytrenowanego modelu
   - Klasyfikacja pojedynczych grzybów poprzez ręczne wprowadzenie cech
6. **Integracja z bazą danych**
   - Centralne zarządzanie danymi przez DataWarehouse

- Spójny system walidacji danych oparty na schemacie
- Możliwość zapisywania zmian do plików

7. **Interfejs użytkownika**
   - Przyjazny interfejs z zakładkami dla różnych funkcjonalności
   - Menu kontekstowe dla szybkich operacji
   - Automatyczna aktualizacja wszystkich widoków po zmianach
   - Komunikaty o błędach i statusie operacji
8. **Obsługa etykiet i kodów**
   - Automatyczna konwersja między kodami a etykietami opisowymi
   - Spójne wyświetlanie etykiet w całej aplikacji
   - Zachowanie kodów w warstwie danych

1.2. Wykorzystane technologie

Podstawowe Technologie:

Python - główny język programowania

- Wykorzystanie typowania (type hints)
- Wykorzystanie klas i dziedziczenia
- Obsługa wyjątków

PyQt5 - framework GUI

- QMainWindow - główne okno aplikacji
- QWidget - podstawowe widgety
- QComboBox, QListWidget, QPushButton - elementy interfejsu
- Delegaty do wyświetlania danych (ComboBoxDelegate)
- System sygnałów i slotów do komunikacji między komponentami

pandas - biblioteka do analizy danych

- DataFrame - przechowywanie i manipulacja danymi
- Operacje na danych (filtrowanie, wybieranie kolumn)

scikit-learn - biblioteka uczenia maszynowego

- Klasyfikator k-NN (w KNNClassifier)
- Podział danych na zbiory treningowy i testowy

Wzorce Projektowe:

MVC (Model-View-Controller)

- Model: DataTable, TableModel
- View: Panele (StatsPanel, SubsetPanel, ClassPanel)
- Controller: SubsetController, ClassController

Singleton

- TableStore - centralne zarządzanie danymi

Observer

- Wykorzystany w TableStore do powiadamiania o zmianach

Strategy

- Różne strategie ekstrakcji podzbiorów w SubsetExtractor

Narzędzia i Biblioteki Pomocnicze:

- NumPy - operacje na macierzach i obliczenia numeryczne
- Pickle - serializacja modeli uczenia maszynowego
- pytest - testy jednostkowe
- typing - adnotacje typów

Architektura:

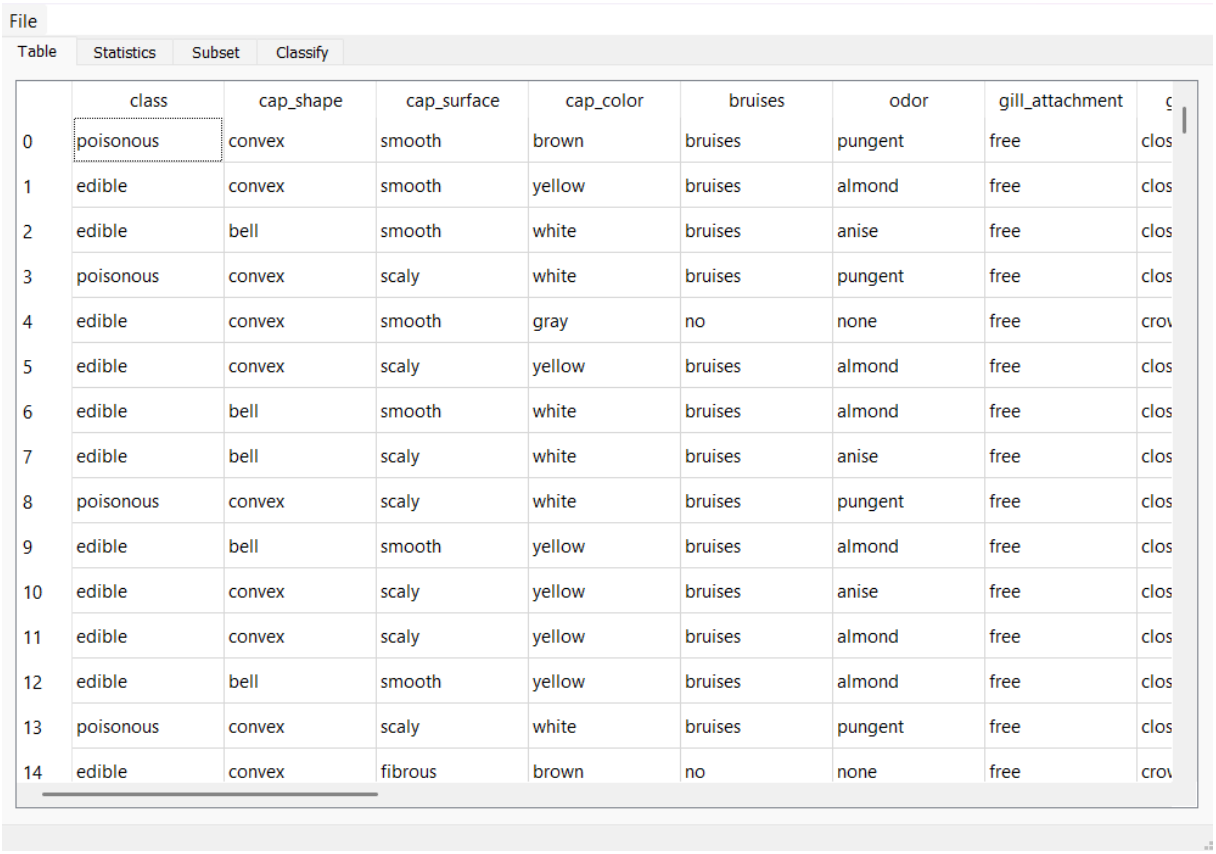
Modułowa struktura projektu:

- core/ - podstawowe klasy i funkcjonalności
- gui/ - interfejs użytkownika
- classification/ - komponenty uczenia maszynowego
- extractor/ - ekstrakcja podzbiorów
- editor/ - edycja danych
- preprocessing/ - przetwarzanie danych

Abstrakcje systemowe:

- Schema - walidacja i mapowanie danych
- DataWarehouse - zarządzanie danymi
- TableStore - zarządzanie stanem aplikacji

1.3. Projekt GUI



	class	cap_shape	cap_surface	cap_color	bruises	odor	gill_attachment	g
0	poisonous	convex	smooth	brown	bruises	pungent	free	clos
1	edible	convex	smooth	yellow	bruises	almond	free	clos
2	edible	bell	smooth	white	bruises	anise	free	clos
3	poisonous	convex	scaly	white	bruises	pungent	free	clos
4	edible	convex	smooth	gray	no	none	free	crov
5	edible	convex	scaly	yellow	bruises	almond	free	clos
6	edible	bell	smooth	white	bruises	almond	free	clos
7	edible	bell	scaly	white	bruises	anise	free	clos
8	poisonous	convex	scaly	white	bruises	pungent	free	clos
9	edible	bell	smooth	yellow	bruises	almond	free	clos
10	edible	convex	scaly	yellow	bruises	anise	free	clos
11	edible	convex	scaly	yellow	bruises	almond	free	clos
12	edible	bell	smooth	yellow	bruises	almond	free	clos
13	poisonous	convex	scaly	white	bruises	pungent	free	clos
14	edible	convex	fibrous	brown	no	none	free	crov

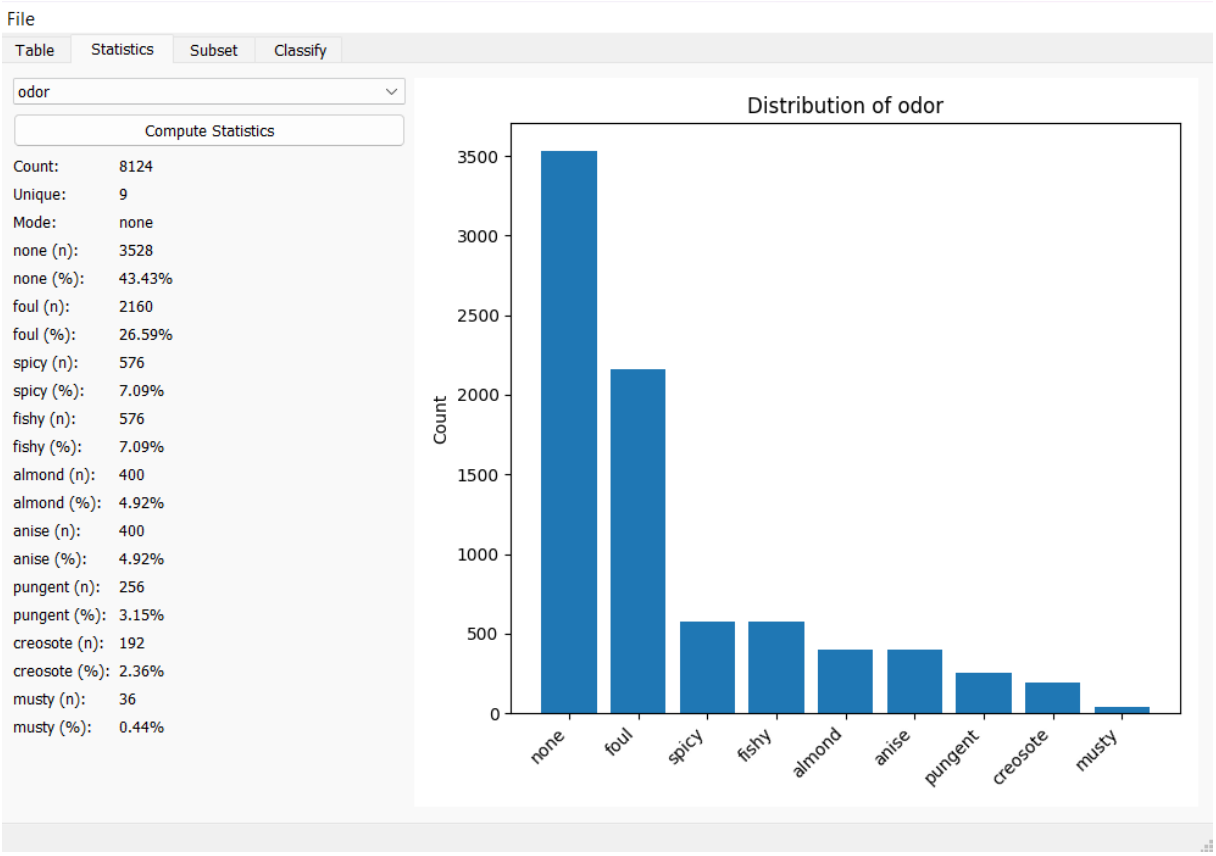
Rysunek 1 Główne okno i zakładka widoku tabeli

Główne Okno Aplikacji

- Menu Plik z opcjami Otwórz/Zapisz/Wyjdź
- Pasek Statusu: Wyświetla aktualny status i wyniki operacji
- Widget Zakładek: Zawiera wszystkie główne panele

Zakładka Widoku Tabeli

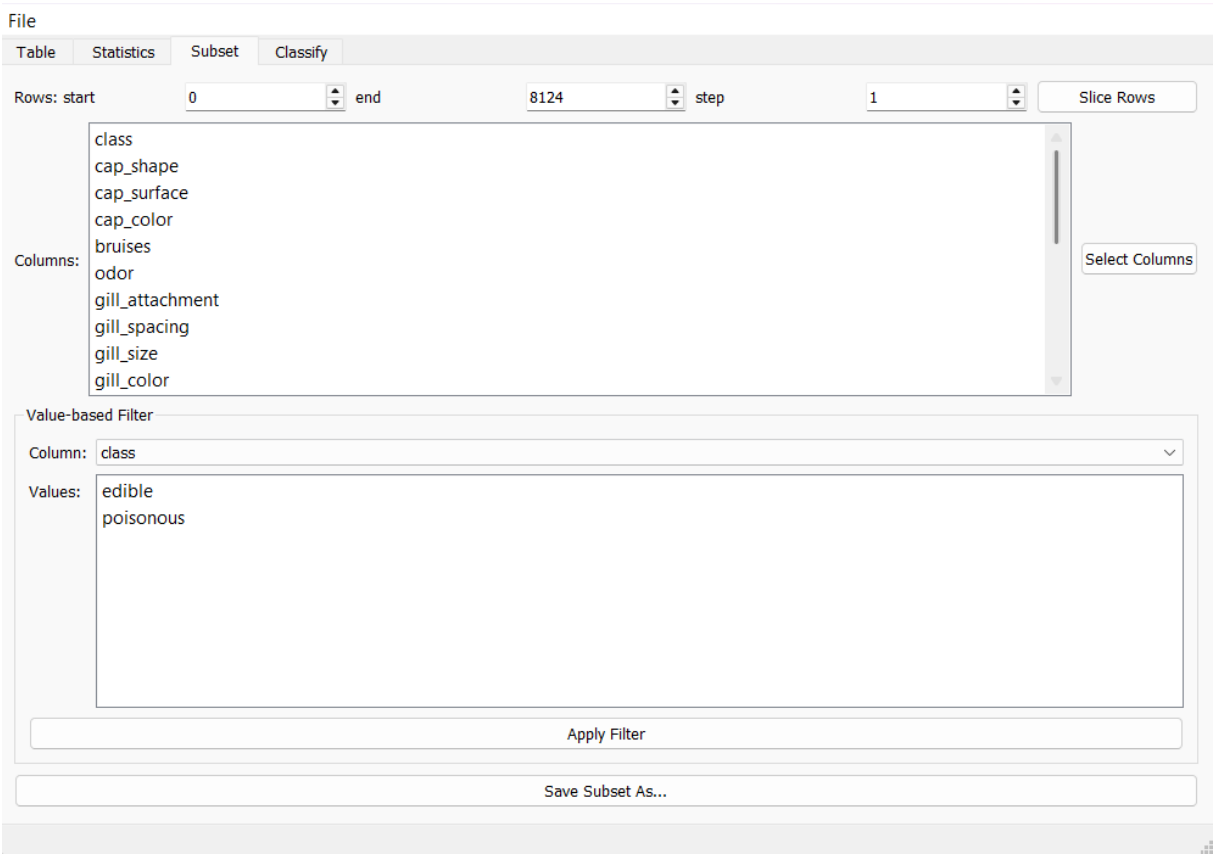
- QTableView: Główny widok danych
    - Wyświetla wszystkie dane w formacie tabelarycznym
    - Używa ComboBoxDelegate do edycji komórek
    - Pokazuje czytelne etykiety zamiast kodów
- Menu Kontekstowe:
    - Wstaw Wiersz
    - Usuń Wiersz(e)
    - Zastąp Wartość w Kolumnie



Rysunek 2 Zakładka statystyki

Panel Statystyk

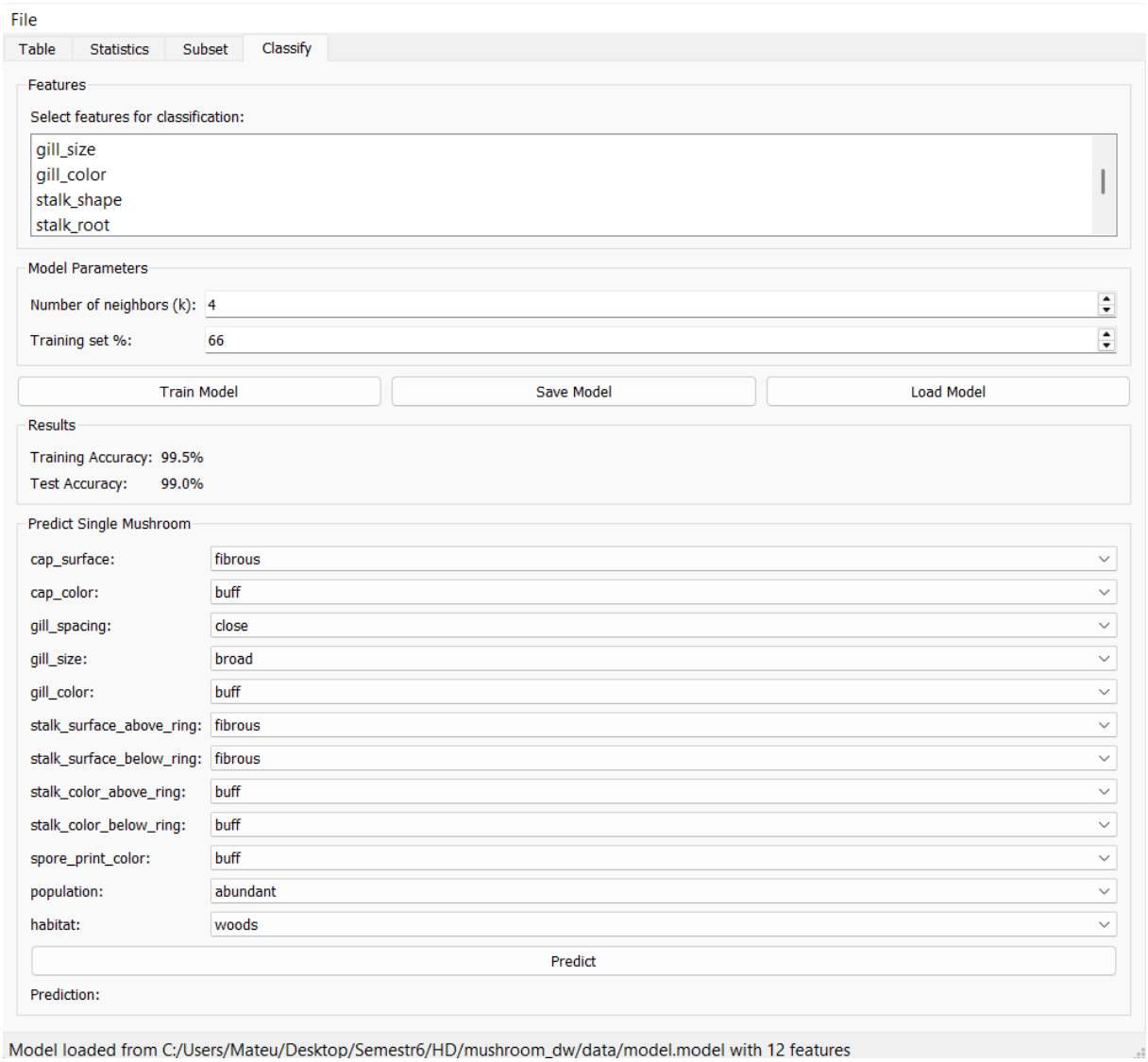
- Lewa Sekcja:
  - Selektor Kolumn (QComboBox): Wybór kolumny do analizy
  - Przycisk Oblicz: Wylicza statystyki
  - Formularz Statystyk: Pokazuje obliczone wartości
    - Podstawowe liczniki
    - Częstotliwości
    - Wartości procentowe
- Prawa Sekcja:
  - Płótno Matplotlib: Pokazuje wizualizację histogramu
  - Dynamiczne skalowanie
  - Obrócone etykiety dla lepszej czytelności



Rysunek 3 Zakładka pozbiorów

Panel Podzbiorów

- Kontrolki Wycinania Wierszy:
  - SpinBox Początek: Pierwszy wiersz do uwzględnienia
  - SpinBox Koniec: Ostatni wiersz do uwzględnienia
  - SpinBox Krok: Interwał między wierszami
  - Przycisk Wytnij: Zastosuj wybór wierszy
- Wybór Kolumn:
  - Lista Kolumn (QListWidget): Lista wielokrotnego wyboru kolumn
  - Przycisk Wybierz Kolumny: Zastosuj wybór kolumn
- Filtr Wartości:
  - Selektor Kolumn (QComboBox): Wybór kolumny do filtrowania
  - Lista Wartości (QListWidget): Wielokrotny wybór dostępnych wartości
  - Przycisk Zastosuj Filtr: Filtruj według wybranych wartości
- Przycisk Zapisz Podzbiór: Eksportuj podzbiór do pliku



Rysunek 4 Zakładka algorytmu Knn

Panel Klasyfikacji

- Wybór Cech:
  - Lista Cech (QListWidget): Wielokrotny wybór kolumn do treningu
- Parametry Modelu:
  - SpinBox k-NN Sąsiadów: Ustaw wartość k (1-10)
  - SpinBox Rozmiar Zbioru Treningowego: Ustaw proporcję podziału (10-90%)
- Kontrolki Treningu:
  - Przycisk Trenuj Model: Rozpocznij trening
  - Przycisk Zapisz Model: Zapisz wytrenowany model
  - Przycisk Wczytaj Model: Wczytaj istniejący model
  - Pasek Postępu: Pokazuje postęp treningu
- Wyświetlanie Wyników:
  - Dokładność Treningu
  - Dokładność Testu
  - Liczby Próbek
- Pojedyncza Predykcja:

- Pola Wprowadzania Cech: ComboBox dla każdej wybranej cechy
- Przycisk Przewiduj: Wykonaj pojedynczą predykcję
- Etykieta Wyniku: Pokazuje przewidywanie z kodowaniem kolorów
  - Zielony dla jadalnych
  - Czerwony dla trujących

2. Wygląd i użytkowanie aplikacji

2.1. Wymagania do uruchomienia aplikacji

Wymagania sprzętowe

- Procesor: dwurdzeniowy CPU (zalecany Intel i3/Ryzen 3 lub lepszy)
- Pamięć RAM: minimum 4 GB (zalecane 8 GB lub więcej)
- Dysk twardy: co najmniej 500 MB wolnego miejsca na pliki aplikacji oraz dodatkowe 1–2 GB na dane i logi
- Karta graficzna: obsługa rozdzielczości co najmniej 1024×768 (w przypadku bardziej rozbudowanych wykresów – obsługa OpenGL 2.0+)

Wymagania programowe

- System operacyjny:
- Windows 10/11 (64-bit)
- macOS 10.14+
- Linux (kernel 4.x+, dystrybucje Debian/Ubuntu, Fedora, CentOS itp.)
- Python: wersja 3.8.x lub wyższa

Biblioteki Pythona:

- PyQt5
- pandas (operacje na tabelach danych)
- numpy (obliczenia numeryczne)
- scikit-learn (analizy statystyczne i klasyfikatory)
- matplotlib (tworzenie wykresów)
- mysql-connector-python (sterownik do komunikacji z bazą MySQL)

Serwer bazy danych:

- MySQL Community Server 5.7+ lub kompatybilny (np. MariaDB 10.2+)
- Uprawnienia do tworzenia schematu i tabel oraz do wykonywania operacji DML (SELECT, INSERT, UPDATE, DELETE)

2.2. Obsługa aplikacji

2.2.1. Wczytanie zbioru danych

Uruchomienie aplikacji

1. Uruchom plik app.py z głównego katalogu projektu
2. Aplikacja wyświetli główne okno z zakładkami: Table, Statistics, Subset i Classify

Import pliku CSV

1. Z menu File wybierz "Open Data File..."
2. Wybierz plik w formacie .data lub .csv zawierający dane o grzybach
3. Po wczytaniu dane zostaną wyświetlone w tabeli w pierwszej zakładce
4. Status wczytania zostanie pokazany na pasku statusu

Podgląd i podstawowa walidacja

- Zakładka "Table" pokazuje pełny zestaw danych
- Kolumny można sortować klikając na nagłówki
- Walidacja odbywa się automatycznie przy:
    - Wczytywaniu pliku
    - Edycji wartości
    - Dodawaniu nowych wierszy

2.2.2. Przekształcanie i czyszczenie danych

Usuwanie braków i duplikatów

1. Po wczytaniu danych aplikacja automatycznie waliduje wartości według schematu
2. Braki danych oznaczone są symbolem '?'
3. Błędne wartości można edytować:
    a. Kliknij prawym przyciskiem myszy na komórkę
    b. Wybierz "Replace Value in Column"
    c. Wybierz poprawną wartość z listy rozwijanej

Kodowanie kolumn symbolicznych

- Aplikacja automatycznie konwertuje kody na etykiety i odwrotnie
- W widoku tabeli wyświetlane są czytelne etykiety (np. "edible" zamiast "e")
- Przy edycji można wybierać spośród dostępnych etykiet
- Przy zapisie do pliku wartości są konwertowane z powrotem na kody

2.2.3. Analiza i eksploracja danych

Statystyki opisowe

1. Wybierz kolumnę z rozwijanej listy
2. Kliknij "Compute Statistics"
3. Wyświetlone zostaną:
    a. Liczba wszystkich i unikalnych wartości
    b. Dominanta
    c. Częstości występowania poszczególnych wartości
    d. Procenty dla każdej wartości

Wizualizacje

W zakładce "Statistics", po obliczeniu statystyk:

- Po prawej stronie wyświetla się wykres słupkowy
- Pokazuje rozkład wartości w wybranej kolumnie

- Etykiety są automatycznie rotowane dla lepszej czytelności
- Wykres automatycznie dostosowuje skalę

Klasyfikacja i eksploracja zaawansowana

W zakładce "Classify":

1. **Wybór cech:**
   - Zaznacz kolumny do wykorzystania w klasyfikacji
   - Kolumna 'class' jest automatycznie używana jako zmienna celu
2. **Parametry modelu:**
   - Ustaw liczbę sąsiadów (k) dla algorytmu k-NN
   - Wybierz procent danych treningowych
3. **Trenowanie:**
   - Kliknij "Train Model"
   - Postęp pokazywany jest na pasku postępu
   - Po zakończeniu wyświetlane są dokładności dla zbioru treningowego i testowego
4. **Predykcja:**
   - Po wytrenowaniu modelu pojawia się sekcja "Predict Single Mushroom"
   - Wybierz wartości cech dla nowego grzyba
   - Kliknij "Predict"
   - Wynik (jadalny/trujący) wyświetlany jest w odpowiednim kolorze
5. **Zapisywanie/wczytywanie modelu:**
   - Wytrenowany model można zapisać przyciskiem "Save Model"
   - Wcześniej zapisany model można wczytać przyciskiem "Load Model"

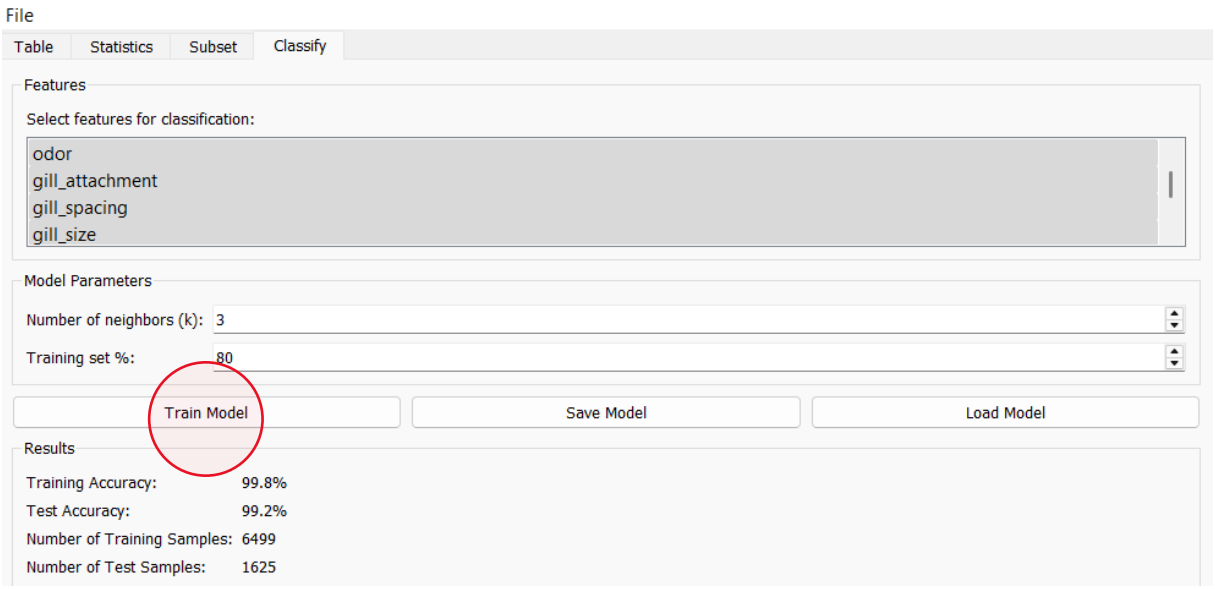
3. Eksperymenty na danych

3.1. Wykorzystane zbiory danych

Z przewodnika terenowego Towarzystwa Audubon; zbiór danych zawiera grzyby opisane pod względem cech fizycznych, wraz z ich klasyfikacją. Ten zbiór danych zawiera opisy hipotetycznych próbek odpowiadających 23 gatunkom grzybów blaszkowych z rodziny pieczarkowatych (Agaricus i Lepiota) (strony 500–525). Każdy gatunek został zaklasyfikowany jako zdecydowanie jadalny, zdecydowanie trujący lub o nieznanej jadalności i niewskazany do spożycia. Ta ostatnia kategoria została połączona z kategorią trujących.

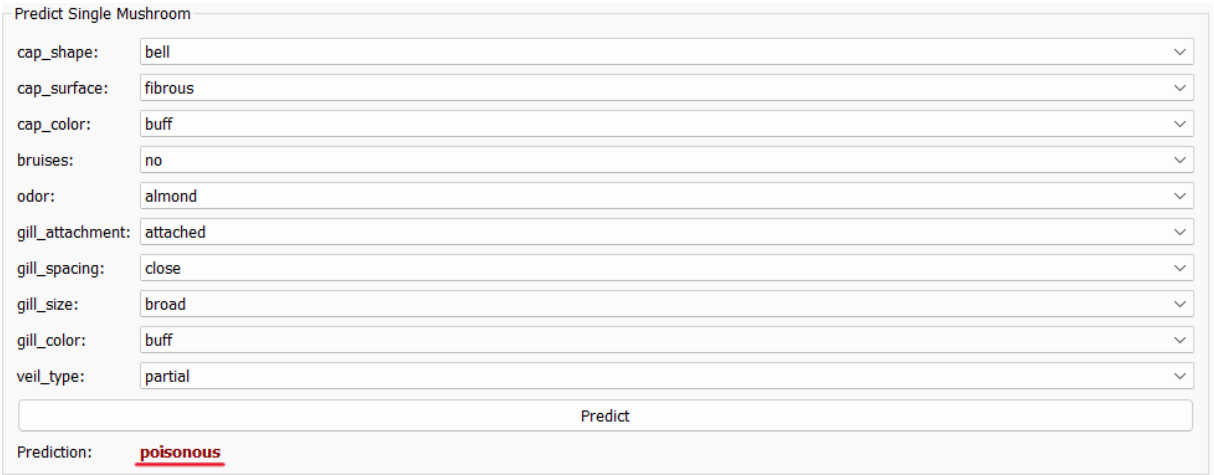
3.2. Przebieg eksperymentu i wyniki

Trenowanie modelu knn na całym zbiorze, z wybranymi cechami



Zastosowanie modelu dla klasyfikacji konkretnego grzyba

3.3. Analiza uzyskanych wyników i wnioski



Algorytm Knn wskazuje, że podany grzyb jest prawdopodobnie trujący. Warto naznaczyć, że poradnik z którego zaczerpnięty jest zbiór danych wyraźnie naznacza, że nie istnieją jednoznaczne reguły pozwalające zidentyfikować jadalność grzyba, zatem algorytm klasyfikacji nie powinien być używany w celu identyfikacji grzybów w celu spożycia.

4. Literatura

1. **Dokumentacja techniczna**
   - PyQt5 Documentation - oficjalna dokumentacja biblioteki Qt5
   - pandas Documentation - dokumentacja biblioteki pandas do analizy danych
   - scikit-learn Documentation - dokumentacja biblioteki uczenia maszynowego
2. **Zbiór danych**
   - Schlimmer, J.C. (1987). Mushroom Database. Retrieved from UCI Machine Learning Repository.
3. **Zasoby online**
   - Real Python: PyQt Tutorials
   - Machine Learning Mastery: k-NN Algorithm
   - Towards Data Science: Mushroom Classification