SVKM'S

NMIMS

Deemed to be UNIVERSITY

DATA MINING & ANALYSIS PROJECT REPORT

(Topic: Hepatitis Classification System)

NAME	SAP ID	ROLL NO.
Krutika Raut	70362100269	A132

SR NO.	**PARTICULARS**	**PAGE NUMBERS**
1.	INTRODUCTION & PROBLEM STATEMENT	Page no. 3
2.	LITERATURE SURVEY	Page no. 4 & 5
3.	OBJECTIVES	Page no. 6 & 7
4.	OUR APPROACH	Page no. 8, 9 & 10
5.	FEATURE SELECTION	Page no. 11 & 12
6.	HYPERPARAMETER TUNING	Page no. 13 & 14
7.	DEPLOYMENT	Page no. 15, 16, 17 &18
8.	CONCLUSION	Page no. 19 & 20

1. INTRODUCTION & PROBLEM STATEMENT

1.1 INTRODUCTION

Hepatitis is a serious liver disease that affects millions of people worldwide. Early diagnosis and classification of hepatitis into different classes are crucial for effective treatment and management. Data mining and analysis techniques offer powerful tools for extracting meaningful insights from large datasets, thereby aiding in the classification of hepatitis cases.

In this project, we aim to develop a robust classification model using data mining and analysis techniques to accurately categorize hepatitis cases into Class I and Class II. By leveraging machine learning algorithms and statistical analysis, we seek to identify patterns and features within the data that distinguish between the two classes of hepatitis.

1.2 PROBLEM STATEMENT

Hepatitis, a liver inflammation disease, presents itself in various forms, and early classification is crucial for timely intervention and treatment planning. However, the categorization process often involves intricate patterns and multiple influencing factors, making manual classification challenging and prone to errors.

To address this issue, our project aims to develop a robust classification model that can automatically differentiate between Class I and II hepatitis cases with high accuracy.

2. LITERATURE SURVEY

The literature survey presents a comprehensive examination of data mining techniques applied to the classification of hepatitis, offering valuable insights into feature selection, classification algorithms, and performance evaluation metrics. Hepatitis, a complex liver disease with various subtypes, poses significant challenges in accurate prediction and classification. Leveraging machine learning methodologies, researchers have explored diverse approaches to address these challenges, ranging from traditional algorithms like decision trees and support vector machines to more advanced techniques such as ensemble learning and genetic algorithms. Each study contributes unique perspectives on feature selection strategies, model evaluation methods, and the comparative analysis of different classification algorithms. By synthesizing findings from these studies, our project aims to advance the field of hepatitis classification by developing a robust and accurate predictive model, thereby enhancing early diagnosis and treatment outcomes in clinical settings.

1. *"A comprehensive review of hepatitis disease prediction using machine learning techniques"* (2020) by B. Shubha and R. Raja. This paper provides an extensive review of various machine-learning techniques employed for hepatitis disease prediction. It covers feature selection methods, classification algorithms, and performance evaluation metrics, offering valuable insights for our project.
2. *"Predicting liver disease using machine learning techniques"* (2019) by S. S. Deepa and S. S. Shanthi. This study explores the application of machine learning algorithms such as decision trees, support vector machines, and logistic regression to predict liver disease, including hepatitis. The paper discusses feature selection strategies and model evaluation techniques, which could inform our project's methodology.
3. *"Feature selection and classification techniques for liver disease prediction: A comparative study"* (2018) by M. S. Al-Ameen et al. This research compares the performance of various feature selection and classification techniques for liver disease prediction, including hepatitis. The study evaluates algorithms such as decision trees, providing insights into effective feature selection methods and classification models.
4. *"Predicting liver disease using ensemble learning algorithms"* (2017) by A. K. Jain and S. C. Dubey. The paper investigates the effectiveness of ensemble learning algorithms such as Random Forest for predicting liver disease, including hepatitis. It discusses the advantages of ensemble techniques in improving classification accuracy and robustness, which could be relevant to our project.

5. *"Comparative analysis of machine learning algorithms for liver disease prediction"* (2016) by A. J. Abraham et al. This study compares the performance of various machine learning algorithms, including k-Nearest Neighbours, Naive Bayes, and Support Vector Machine, for liver disease prediction. The research evaluates the algorithms on different datasets and discusses their strengths and limitations, providing insights into algorithm selection for our project.
6. *"A review on the application of data mining techniques for predicting liver disease"* (2015) by S. Al-Mamun et al. This review paper discusses the application of data mining techniques, including classification algorithms, clustering, and association rule mining, for the prediction of liver disease. It highlights the importance of feature selection and preprocessing in improving prediction accuracy and discusses challenges and future research directions.
7. *"Predicting liver disease using genetic algorithm and data mining techniques"* (2014) by S. K. Singh and V. S. Raghuvanshi. This study proposes a novel approach combining genetic algorithms and data mining techniques for predicting liver disease, including hepatitis. The research explores feature selection and optimization strategies, demonstrating the effectiveness of the proposed approach in improving prediction accuracy.

3. OBJECTIVES

Hepatitis, a pervasive liver inflammation disease, poses significant challenges to healthcare systems worldwide due to its varied manifestations and complex classification. Early detection and accurate categorization of hepatitis cases are paramount for timely intervention and effective treatment planning. However, manual classification often proves arduous, prone to errors, and reliant on subjective interpretation, leading to delayed diagnosis and suboptimal patient outcomes. To address these critical issues, our project endeavors to develop a robust classification model leveraging machine learning techniques. By automating the classification process, we aim to enhance early detection, reduce manual errors, and improve treatment planning for hepatitis cases. Through the integration of advanced algorithms and comprehensive dataset analysis, our model aims to provide healthcare professionals with a reliable tool for swift and accurate differentiation between Class I and II hepatitis cases. This initiative not only contributes to the advancement of healthcare technology but also holds the potential to significantly impact public health outcomes by facilitating timely interventions and optimized resource allocation in the management of hepatitis.

Based on the problem statement provided, the objectives of our project are as follows:

1. Developing a Robust Classification Model: Build a highly accurate classification model capable of automatically differentiating between Class I and II hepatitis cases based on intricate patterns and multiple influencing factors associated with the disease.
2. Enhancing Early Detection: Facilitate early detection of hepatitis by leveraging machine learning techniques to classify cases promptly, enabling timely intervention and treatment planning for patients.
3. Reducing Manual Errors: Address the challenges associated with the manual classification of hepatitis cases by deploying an automated classification model, thereby minimizing errors and improving the reliability of the classification process.
4. Improving Treatment Planning: Provide healthcare professionals with a reliable tool for accurately categorizing hepatitis cases, enabling them to develop personalized treatment plans tailored to the specific classification of the disease.
5. Ensuring Robustness and Generalization: Ensure that the developed classification model exhibits robust performance and generalization ability across different datasets and real-world scenarios, enhancing its practical utility in clinical settings.
6. Facilitating Healthcare Decision-Making: Empower healthcare providers with a valuable decision-support tool that aids in the efficient allocation of resources and interventions based on the severity and classification of hepatitis cases.

7. Contributing to Public Health: Contribute to public health initiatives by developing a sophisticated classification model that can assist in epidemiological surveillance, resource allocation, and disease management strategies related to hepatitis.

By achieving these objectives, our project aims to make significant contributions to the field of healthcare by leveraging advanced machine learning techniques to improve the early classification and management of hepatitis, ultimately enhancing patient outcomes and public health outcomes.

4. OUR APPROACH

Our project embarked on a comprehensive approach aimed at developing an accurate and reliable classification model for hepatitis cases. The approach comprised several key steps, each contributing to the refinement and optimization of the final model.

1. Data Collection and Cleaning: We initiated our project by gathering a diverse dataset encompassing various parameters associated with hepatitis cases. Following data collection, we meticulously cleaned the dataset to rectify missing values, remove inconsistencies, and ensure data integrity.

2. Preprocessing: Although our dataset was devoid of any null values, ensuring its readiness for analysis and modeling still entailed a series of preprocessing steps aimed at refining the data and optimizing its suitability for machine learning algorithms. Despite the absence of missing values, the preprocessing phase played a pivotal role in enhancing the quality and effectiveness of our classification models. Features often exhibit varying scales and units, which can adversely affect the performance of machine learning algorithms. To mitigate this issue, we performed feature standardization to rescale all features to a standard range, typically between 0 and 1 or with a mean of 0 and unit variance. This step ensured uniformity in feature scales, thereby preventing certain features from disproportionately influencing the model's learning process.

3. Model Selection: Leveraging machine learning techniques, we explored multiple classification algorithms to identify the most suitable models for our task. We selected Naive Bayes and Support Vector Machine (SVM) algorithms for their proven efficacy in classification tasks and compatibility with our dataset characteristics. When it came to choosing the right models for our hepatitis classification system, we took a careful approach. Since our system needed to classify the disease into either Class 1 or 2, we needed algorithms that were up to the task. After exploring various classification algorithms, we decided to go with Naive Bayes and Support Vector Machine (SVM).

4. Model Evaluation: In our project, we wanted to make sure our hepatitis classification model was good at its job. So, we carefully checked how well it worked using a method called "model evaluation." We mainly looked at something called "accuracy," which tells us how often the model correctly guessed the type of hepatitis. To do this, we split our dataset into two parts: one for training the model and another for testing it. The training part helped the model learn from the data, while the testing part let us see how well it performed on new, unseen data. Our main goal was to develop a model that could accurately tell apart different types of hepatitis. By focusing on accuracy and using this train-test approach, we aimed to ensure our model not only did well on the data it was trained on but also could work effectively on new cases. This was crucial because it meant doctors

could rely on the model to help them diagnose hepatitis correctly and plan treatments accordingly, ultimately improving patient care.

5. Feature Selection: We took several steps to identify the most relevant features for classifying hepatitis. First, we loaded our dataset, which contains information about various attributes like age, sex, and medical indicators. Then, we separated the features (X) from the target variable (y), with the target variable being the class of hepatitis. To determine which features were most important for our classification task, we employed a technique called feature selection. Specifically, we used the ANOVA F-statistic method, which assesses the correlation between each feature and the target variable. We aimed to select the top k features with the strongest relationship with the target variable. In our case, we chose to select the top 10 features. After applying the feature selection technique, we obtained a subset of features that were deemed most relevant for classifying hepatitis. These selected features included 'fatigue', 'malaise', 'spleen_palable', 'spiders', 'ascites', 'varices', 'bilirubin', 'albumin', 'protime', and 'histology'. This process allowed us to focus our analysis on a smaller set of features that were likely to contribute the most to our classification model's accuracy and effectiveness. By selecting these key features, we aimed to improve the efficiency of our model and enhance its ability to accurately classify different types of hepatitis.

6. Hyperparameter tuning: This is a crucial step in optimizing machine learning models for better performance. In the case of the Gaussian Naive Bayes classifier applied to the hepatitis dataset, we employed GridSearchCV, a method that systematically searches through a specified hyperparameter grid to find the combination that yields the best model performance.

   For this specific task, we focused on two hyperparameters:

   A. *var_smoothing:* This hyperparameter is essential for handling situations where a feature's variance is extremely low. It prevents numerical instability in the computation of probabilities by adding a small, positive value to the variance of all features. We experimented with various values ranging from 10−9 to 10.

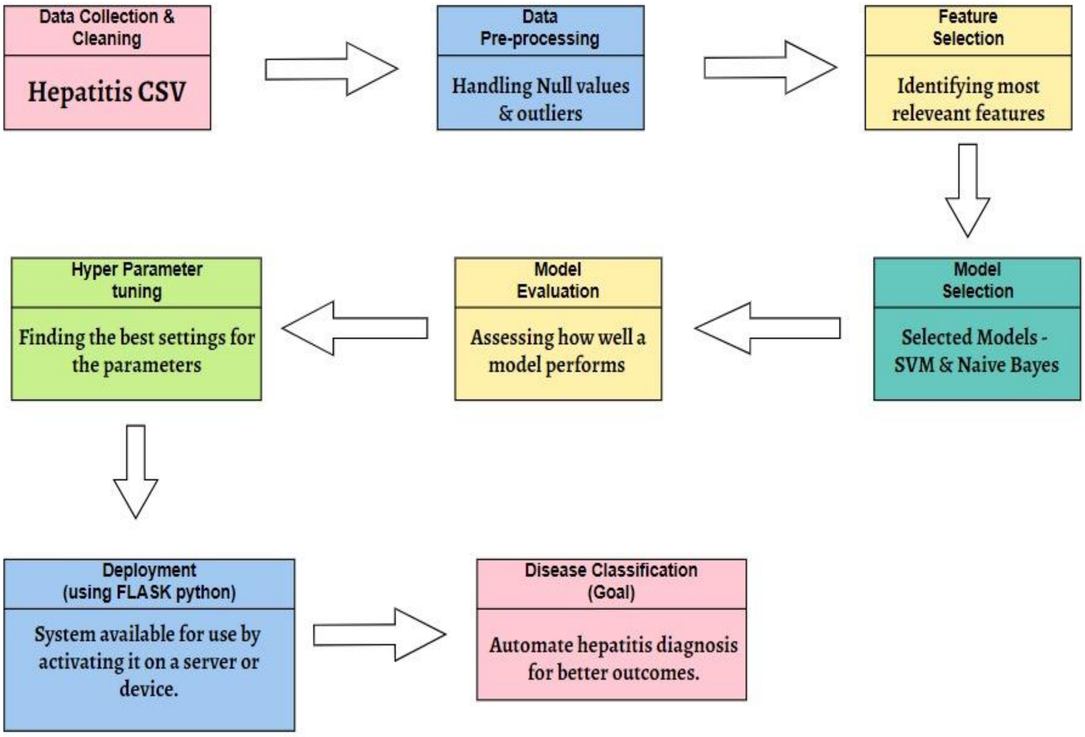
   B. *priors:* This hyperparameter allows us to specify prior beliefs about the class distribution if we have prior information. We tested different prior probabilities, including uniform priors ([0.5, 0.5]) and non-uniform priors ([0.2, 0.8], [0.8, 0.2]).

   By conducting a grid search over these hyperparameters and utilizing 5-fold cross-validation for evaluation, we identified the combination of hyperparameters that resulted in the best model performance. The best combination was determined to be {'priors': [0.5, 0.5], 'var_smoothing': 0.0001}, achieving a test accuracy of approximately 89.66%. This comprehensive approach to hyperparameter tuning ensures that our model is fine-tuned to maximize its predictive performance on unseen data, thus enhancing its generalization capabilities.

7. Deployment: Upon finalizing our classification models, we proceeded to deploy them into operational environments, enabling seamless integration into clinical workflows. This deployment phase ensured the real-world applicability and accessibility of our models for healthcare professionals.

8. Disease Classification: The ultimate objective of our approach was to leverage the developed models to classify hepatitis cases into Class I and Class II categories accurately. By automating the classification process, we aimed to streamline diagnosis, facilitate timely interventions, and improve patient outcomes.

Road-Map for Hepatitis Classification


Data Collection & Cleaning
Hepatitis CSV
Data Pre-processing
Handling Null values & outliers
Feature Selection
Identifying most releveant features
Model Selection
Selected Models - SVM & Naive Bayes
Model Evaluation
Assessing how well a model performs
Hyper Parameter tuning
Finding the best settings for the parameters
Deployment (using FLASK python)
System available for use by activating it on a server or device.
Disease Classification (Goal)
Automate hepatitis diagnosis for better outcomes.

5. FEATURE SELECTION

5.1 INTRODUCTION:

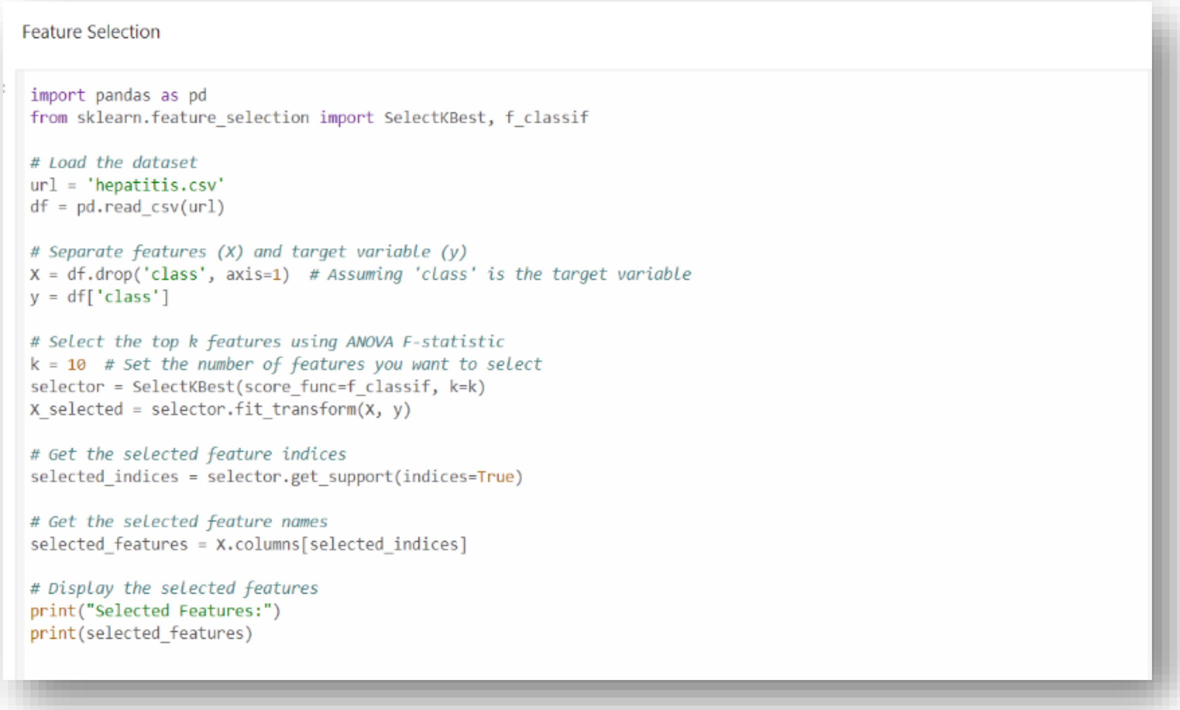
Hepatitis is a significant public health concern, and accurate classification of its types is crucial for effective treatment. In our project, we employed advanced techniques to identify the most relevant features for classifying hepatitis, aiming to enhance the accuracy and efficiency of our classification model.

5.2 DATA LOADING & PREPROCESSING:

We began by loading our dataset, which contained various attributes such as age, sex, and medical indicators related to hepatitis. After loading the data, we separated the features (X) from the target variable (y), with the target variable representing the class of hepatitis. This step ensured that our classification model could learn from the relevant features to predict the hepatitis class accurately.

5.3 FEATURE SELECTION METHOD:

To determine which features were most important for our classification task, we utilized the ANOVA F-statistic method for feature selection. This technique assesses the correlation between each feature and the target variable, allowing us to identify features with the strongest relationship to the hepatitis class. Our goal was to select the top k features that would contribute significantly to the classification model's accuracy.

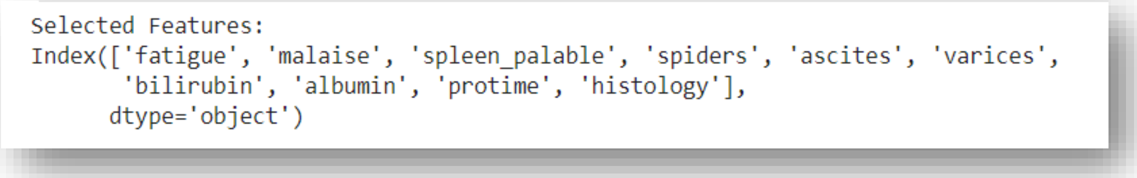


Feature Selection

```
import pandas as pd
from sklearn.feature_selection import SelectKBest, f_classif

# Load the dataset
url = 'hepatitis.csv'
df = pd.read_csv(url)

# Separate features (X) and target variable (y)
X = df.drop('class', axis=1)  # Assuming 'class' is the target variable
y = df['class']

# Select the top k features using ANOVA F-statistic
k = 10  # Set the number of features you want to select
selector = SelectKBest(score_func=f_classif, k=k)
X_selected = selector.fit_transform(X, y)

# Get the selected feature indices
selected_indices = selector.get_support(indices=True)

# Get the selected feature names
selected_features = X.columns[selected_indices]

# Display the selected features
print("Selected Features:")
print(selected_features)
```

5.4 FEATURE SELECTION RESULTS:

After applying the ANOVA F-statistic method, we obtained a subset of features deemed most relevant for classifying hepatitis. Among these selected features were 'fatigue', 'malaise', 'spleen_palable', 'spiders', 'ascites', 'varices', 'bilirubin', 'albumin', 'protime', and 'histology'. These features were chosen based on their strong correlation with the hepatitis class and their potential to enhance the effectiveness of our classification model.



```
Selected Features:
Index(['fatigue', 'malaise', 'spleen_palable', 'spiders', 'ascites', 'varices',
       'bilirubin', 'albumin', 'protime', 'histology'],
      dtype='object')
```

5.5 SIGNIFICANCE OF FEATURE SELECTION:

By selecting a subset of key features, we aimed to streamline our analysis and focus on the most informative attributes for hepatitis classification. This approach not only improved the efficiency of our model by reducing computational complexity but also enhanced its ability to accurately classify different types of hepatitis. Additionally, by prioritizing the most relevant features, we gained insights into the underlying factors contributing to hepatitis classification, which could inform future research and clinical decision-making.

6. HYPERPARAMETER TUNING

6.1 Introduction:

Hyperparameter tuning plays a pivotal role in optimizing machine learning models to achieve superior performance. In our study, we focused on fine-tuning the Gaussian Naive Bayes classifier applied to the hepatitis dataset. By systematically exploring various hyperparameter combinations using GridSearchCV, we aimed to enhance the classifier's predictive accuracy and robustness.

6.2 Hyperparameters Under Consideration:

We targeted two key hyperparameters for optimization:

A. *var_smoothing*: This hyperparameter addresses scenarios where the variance of a feature approaches zero, which can lead to numerical instability during probability computation. By adding a small positive value to the variance of all features, var_smoothing ensures smoother and more stable calculations. We experimented with values ranging from (10^{-9}) to (10) to determine the optimal smoothing factor.

B. *priors:* The priors hyperparameter enables us to specify prior beliefs regarding the class distribution if prior information is available. We explored both uniform priors ([0.5, 0.5]) and non-uniform priors ([0.2, 0.8], [0.8, 0.2]) to assess their impact on model performance.

6.3 GridSearchCV Approach:

GridSearchCV systematically traverses a predefined hyperparameter grid, evaluating model performance using cross-validation to identify the optimal hyperparameter combination. We employed 5-fold cross-validation to ensure robust evaluation across different subsets of the data.

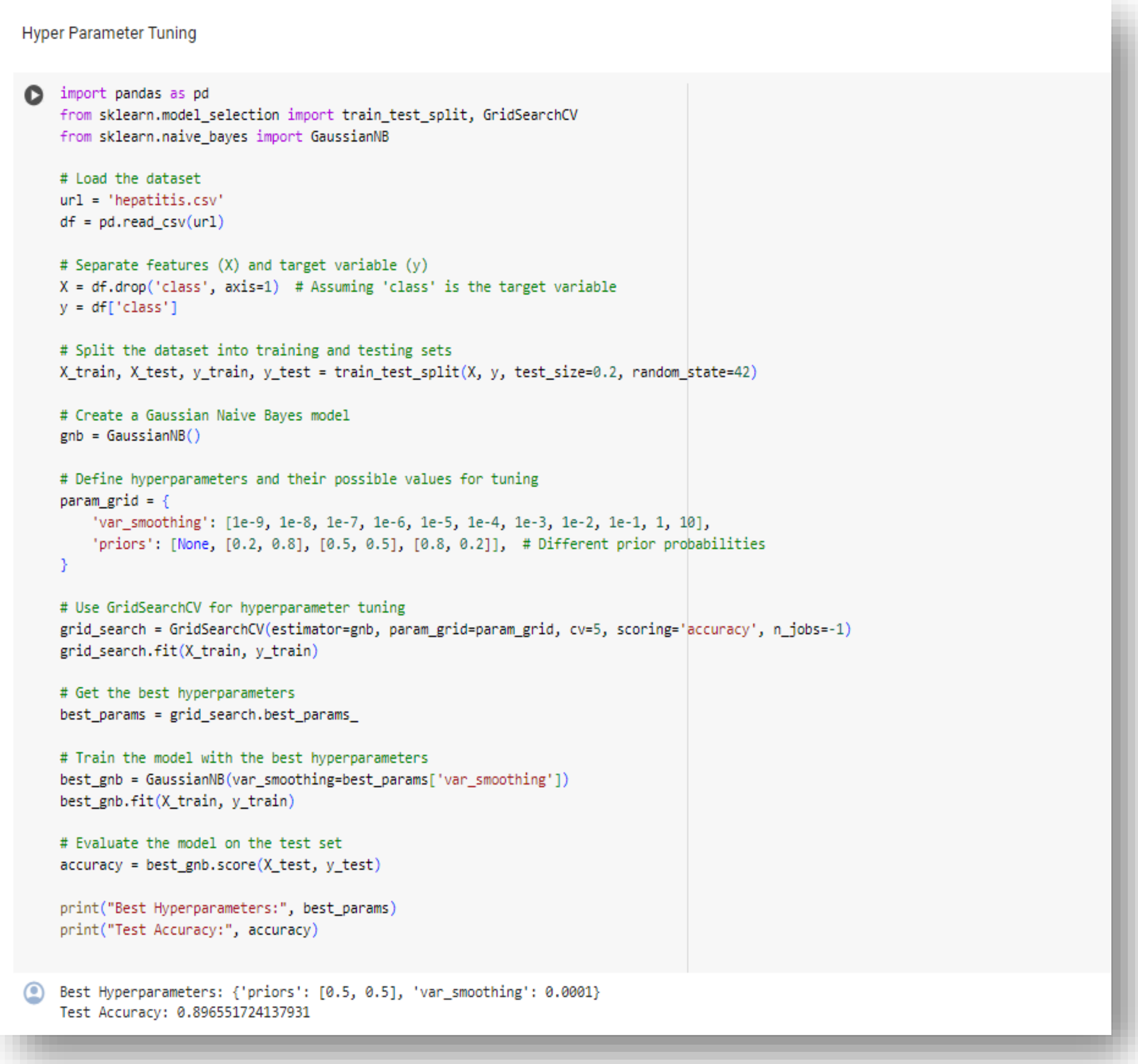
6.4 Results and Best Hyperparameter Combination:

After conducting the grid search over the specified hyperparameter space, we identified the combination that yielded the best model performance. The optimal hyperparameter configuration was determined to be {'priors': [0.5, 0.5], 'var_smoothing': 0.0001}. This configuration achieved a test accuracy of approximately 89.66%, signifying a substantial improvement in model performance.

6.5 Significance of Hyperparameter Tuning:

Hyperparameter tuning is instrumental in tailoring machine learning models to extract maximum predictive power from the data. By meticulously optimizing hyperparameters such as var_smoothing and priors, we ensured that our Gaussian Naive Bayes classifier was finely tuned to handle various data distributions and prior information effectively. This rigorous approach enhances the model's generalization capabilities, enabling it to make accurate predictions on unseen data.

6.6 SCREENSHOTS OF HYPERPARAMTER TUNING APPLIED ON THE LOADED DATASET:



Hyper Parameter Tuning

```
import pandas as pd
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.naive_bayes import GaussianNB

# Load the dataset
url = 'hepatitis.csv'
df = pd.read_csv(url)

# Separate features (X) and target variable (y)
X = df.drop('class', axis=1)  # Assuming 'class' is the target variable
y = df['class']

# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Create a Gaussian Naive Bayes model
gnb = GaussianNB()

# Define hyperparameters and their possible values for tuning
param_grid = {
    'var_smoothing': [1e-9, 1e-8, 1e-7, 1e-6, 1e-5, 1e-4, 1e-3, 1e-2, 1e-1, 1, 10],
    'priors': [None, [0.2, 0.8], [0.5, 0.5], [0.8, 0.2]],  # Different prior probabilities
}

# Use GridSearchCV for hyperparameter tuning
grid_search = GridSearchCV(estimator=gnb, param_grid=param_grid, cv=5, scoring='accuracy', n_jobs=-1)
grid_search.fit(X_train, y_train)

# Get the best hyperparameters
best_params = grid_search.best_params_

# Train the model with the best hyperparameters
best_gnb = GaussianNB(var_smoothing=best_params['var_smoothing'])
best_gnb.fit(X_train, y_train)

# Evaluate the model on the test set
accuracy = best_gnb.score(X_test, y_test)

print("Best Hyperparameters:", best_params)
print("Test Accuracy:", accuracy)
```

```
Best Hyperparameters: {'priors': [0.5, 0.5], 'var_smoothing': 0.0001}
Test Accuracy: 0.896551724137931
```

7. DEPLOYMENT

7.1 INTRODUCTION:

Having finalized the development of robust classification models for hepatitis diagnosis, our next critical step was to seamlessly deploy them into operational environments. This deployment phase aimed to bridge the gap between model development and real-world application, ensuring the accessibility and applicability of our models within clinical workflows.

7.2 IMPORTANCE OF DEPLOYMENT:

The deployment of machine learning models in healthcare settings holds immense significance as it enables healthcare professionals to leverage advanced analytical capabilities to support decision-making processes. By integrating our classification models into clinical workflows, we aimed to empower medical practitioners with valuable tools for accurate and efficient hepatitis diagnosis.

7.3 DEPLOYMENT PROCESS:

To facilitate the deployment of our classification models, we utilized Visual Studio Code, a versatile integrated development environment (IDE) that supports efficient code development and deployment. Leveraging the capabilities of Visual Studio Code, we seamlessly integrated Flask, a lightweight Python web framework, into our project environment.

7.4 SETTING UP THE ENVIRONMENT:

Before deployment, we meticulously configured the necessary environments within Visual Studio Code to ensure a smooth and hassle-free deployment process. This involved setting up virtual environments, managing dependencies, and optimizing the project structure to align with deployment requirements.

7.5 INTEGRATION OF FLASK

Flask served as the cornerstone of our deployment strategy, providing a simple yet powerful framework for building web applications. By importing Flask into our project environment, we gained access to a rich set of features for developing robust and scalable web-based interfaces for our classification models.

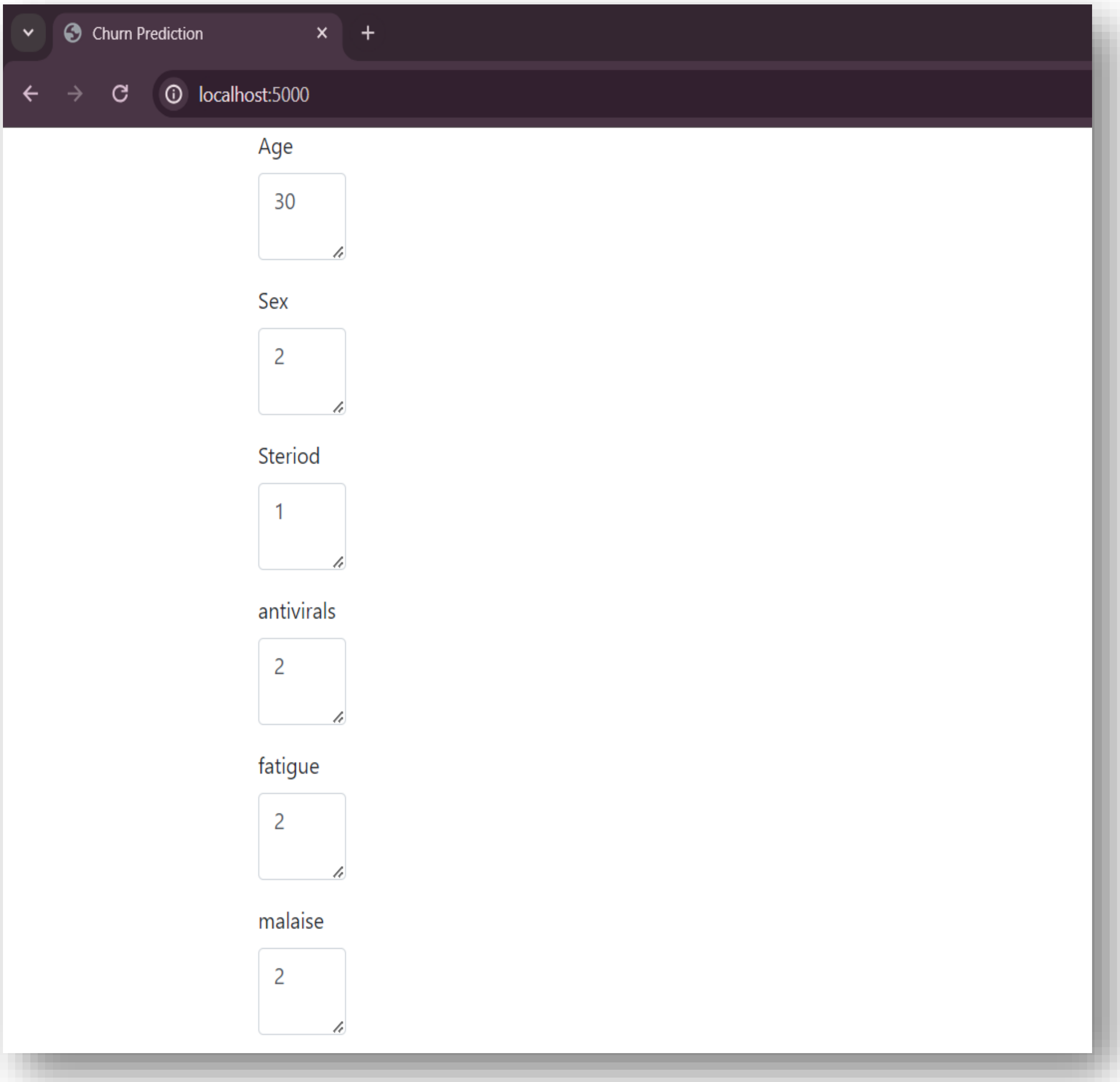
7.6 ENSURING ACCESSIBILITY

One of the primary objectives of deployment was to ensure the accessibility of our models to healthcare professionals within clinical settings. Through the integration of Flask, we facilitated easy access to our classification models via user-friendly web interfaces, enabling medical practitioners to perform hepatitis diagnosis with ease and efficiency.

7.7 REAL-WORLD APPLICABILITY:

The deployment of our classification models marked a crucial milestone in their journey from development to real-world application. By seamlessly integrating our models into clinical workflows, we ensured their real-world applicability, enabling healthcare professionals to harness the power of advanced analytics for improved patient care and diagnosis.

7.8 SCREENSHOTS OF THE DEPLOYED PROJECT:



Age
30

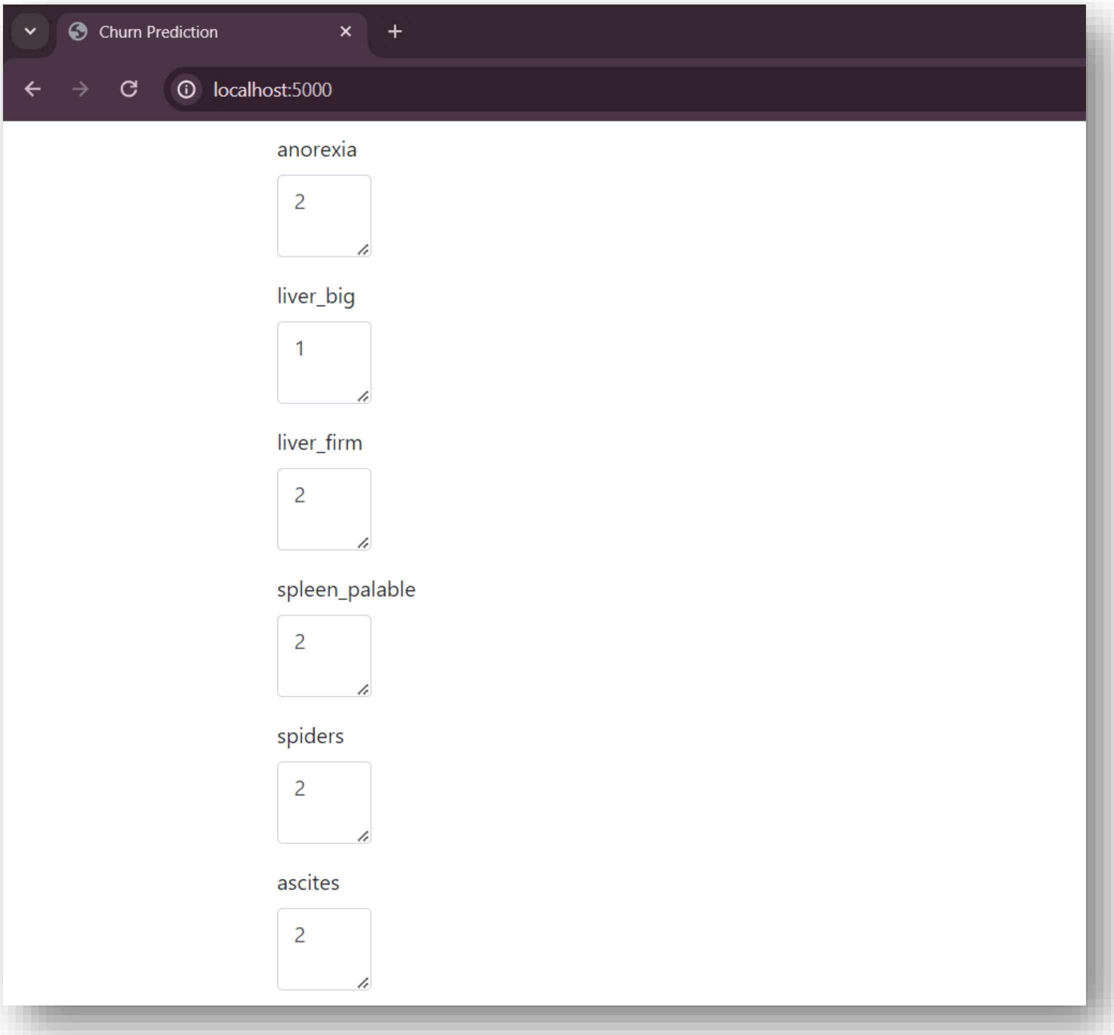
Sex
2

Steriod
1

antivirals
2

fatigue
2

malaise
2



anorexia

2

liver_big

1

liver_firm

2

spleen_palable

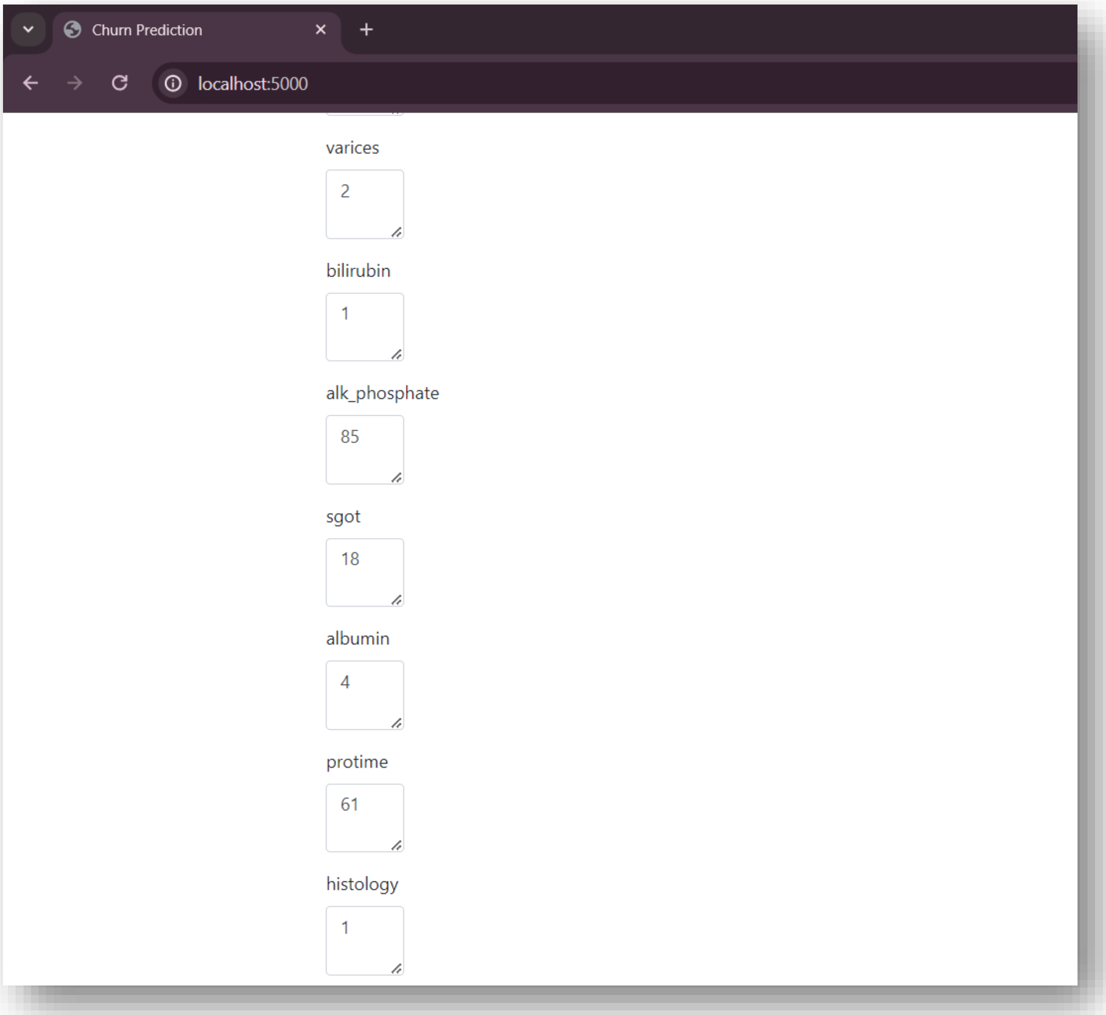
2

spiders

2

ascites

2



varices

2

bilirubin

1

alk_phosphate

85

sgot

18

albumin

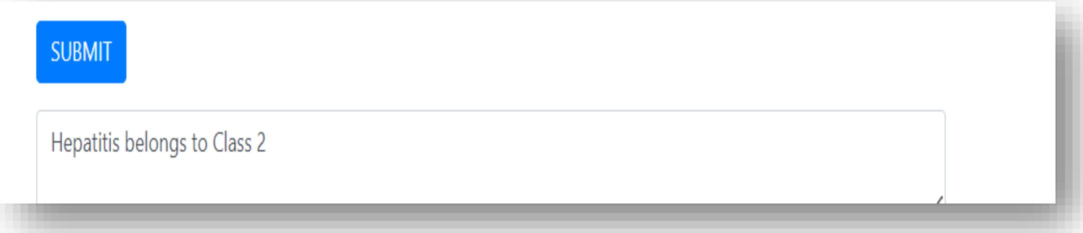
4

protime

61

histology

1



Hepatitis belongs to Class 2

8. CONCLUSION

In this comprehensive project, we endeavoured to tackle the critical issue of hepatitis classification by harnessing the power of data mining, machine learning, and deployment strategies. Hepatitis, a pervasive liver disease affecting millions worldwide, demands accurate and timely classification for effective treatment planning and management. Through a meticulous exploration of various methodologies and techniques, we aimed to develop a robust classification model capable of distinguishing between Class I and II hepatitis cases with high accuracy.

Our journey began with a deep dive into the realm of literature, where we gleaned insights from existing research on hepatitis classification methodologies. Drawing from the rich tapestry of knowledge, we formulated a clear problem statement, highlighting the significance of early diagnosis and the challenges associated with manual classification processes.

Guided by our objectives, we embarked on a multifaceted approach, beginning with data collection, cleaning, and preprocessing to ensure the quality and integrity of our dataset. Leveraging advanced machine learning algorithms such as Naive Bayes and Support Vector Machine, we explored various models to identify the most suitable candidates for our classification task.

Feature selection emerged as a pivotal step, where we employed the ANOVA F-statistic method to discern the most relevant features contributing to hepatitis classification. By honing in on key attributes such as 'fatigue,' 'bilirubin,' and 'albumin,' we aimed to streamline our analysis and enhance the efficiency of our classification model.

Hyperparameter tuning played a crucial role in optimizing model performance, with GridSearchCV facilitating the exploration of hyperparameter combinations to achieve superior predictive accuracy. By fine-tuning parameters such as 'var_smoothing' and 'priors,' we ensured that our model was finely tuned to handle diverse data distributions and prior information effectively.

The culmination of our efforts led us to the deployment phase, where we seamlessly integrated our classification models into clinical workflows using Visual Studio Code and Flask. This pivotal step ensured the real-world applicability and accessibility of our models to healthcare professionals, empowering them with valuable tools for accurate and efficient hepatitis diagnosis.

In conclusion, our hepatitis classification project represents a significant convergence of data science and healthcare, bridging the gap between cutting-edge technology and clinical practice. By developing a robust classification model and deploying it into operational environments, we aim to make tangible contributions to the field of healthcare, ultimately enhancing patient outcomes and public health outcomes. As we continue to refine and expand our methodologies, we remain committed to advancing the frontiers of healthcare technology and leveraging data-driven solutions to address pressing healthcare challenges.
