Machine Learning & Data-Mining : Research Project

Predictions of breast cancer using decision trees

Introduction - Short outline of the problematic

Breast cancer is a prevalent and life-threatening disease that affects individuals worldwide. Timely and accurate diagnosis is critical for improving patient outcomes, making the development of effective diagnostic tools a priority in healthcare. In this research project, I focus on the implementation of decision trees as a predictive model for distinguishing between benign and malignant breast tumors.

Breast cancer is characterized by various clinical and pathological features, and its early detection is crucial for successful treatment. Decision trees offer an interpretable approach to classifying breast cancer cases, making them an attractive option for this task.

In this report, I will outline the steps taken to implement decision trees for breast cancer prediction. This includes data collection, preprocessing, model development, evaluation, and interpretation of the results. The aim is to provide a comprehensive understanding of this approach and its potential impact on the field of breast cancer diagnosis.

Methodology - Steps taken to tackle the problem

In my implemention of decision trees for breast cancer prediction, I went through the following steps :

- **Data collection :**

My research relied on the dataset « Diagnostic Winsconsin Breast Cancer Database » obtained on Kaggle : https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data/data

- **Data Preprocessing :**

Before starting to develop our decision tree model, I did some preprocessing steps to ensure the dataset's suitability for my implementation.

First of all, I started to check if there are any missing values to ensure that the data remained complete and usable for the analysis. After that, I removed unnecessary columns and retained only those features that were relevant to our research. These steps where crucial for model efficiency and reducing the dimensionality of the data.

Then I extracted the features and the target variable from the dataset. The features represent the clinical and pathological attributes, while the target variable distinguishes between benign and malignant breast tumors.

In this step, I tried to see what features of the dataset have the biggest impact on results in order to reduce the number of features (more than 30). By retaining only features that are not correlated, the model can improve in interpretability. This function helped me in obtaining a correlation map of the Breast Cancer dataset’s features and in choosing the relevant features in the final implementation of the decision tree model.

```python
import seaborn as sns

def feature_correlation_map(dataframe):
    # Calculate the correlation matrix
    corr_matrix = dataframe.corr()

    # Create a mask for the upper triangle
    mask = np.triu(np.ones_like(corr_matrix, dtype=bool))

    # Set up the matplotlib figure
    plt.figure(figsize=(12, 10))

    # Draw the heatmap with the mask
    sns.heatmap(corr_matrix, mask=mask, cmap='coolwarm', vmax=1, center=0,
                square=True, linewidths=.5, cbar_kws={"shrink": 0.5})

    plt.show()
```

- **Before Decision Tree Model Implementation :**

In order to implement this model, I got helped from the first assignment of Machine Learning & Data-Mining course about decision trees.

Multiple functions have been implemented :

- The prior function calculates the prior probability of each class type in a classification problem such as this research. It represents the distribution of classes in the dataset before considering any features.
- The split_train_test function shuffles and splits the dataset into training and testing sets based on a specified ratio. I am using the same split_train_test function from the tools of the assignment.
- The split_data function splits the dataset and targets into two separate datasets based on a given feature and threshold.
- The gini_impurity function calculates the Gini impurity for a set of targets.
- The weighted_impurity function calculates the weighted sum of Gini impurities for two branches.
- The total_gini_impurity function calculates the total Gini impurity which returns the weighted impurity given the dataset and threshold to split on.
- The brute_best_split function finds the best split for the given data by iterating over feature dimensions and thresholds.

- **Decision Tree Model Implementation :**

The heart of my research involved the implementation of a decision tree classifier. I utilized the machine learning library « sklearn » to build my decision tree model. Then I created a class which handles the training and evaluation of the Decision Tree model of the Breast Cancer Winsconsin dataset. It utilizes the DecisionTreeClassifier from scikit-learn and provides methods for training, accuracy calculation, plotting Decision Tree map, making predictions, and generating a confusion matrix.

- **Model Evaluation :**

To gauge the predictive capabilities of my decision tree model, I calculated the accuracy and the confusion matrix :

- Accuracy, the proportion of correctly predicted cases. It provides an overall assessment of the model's performance.

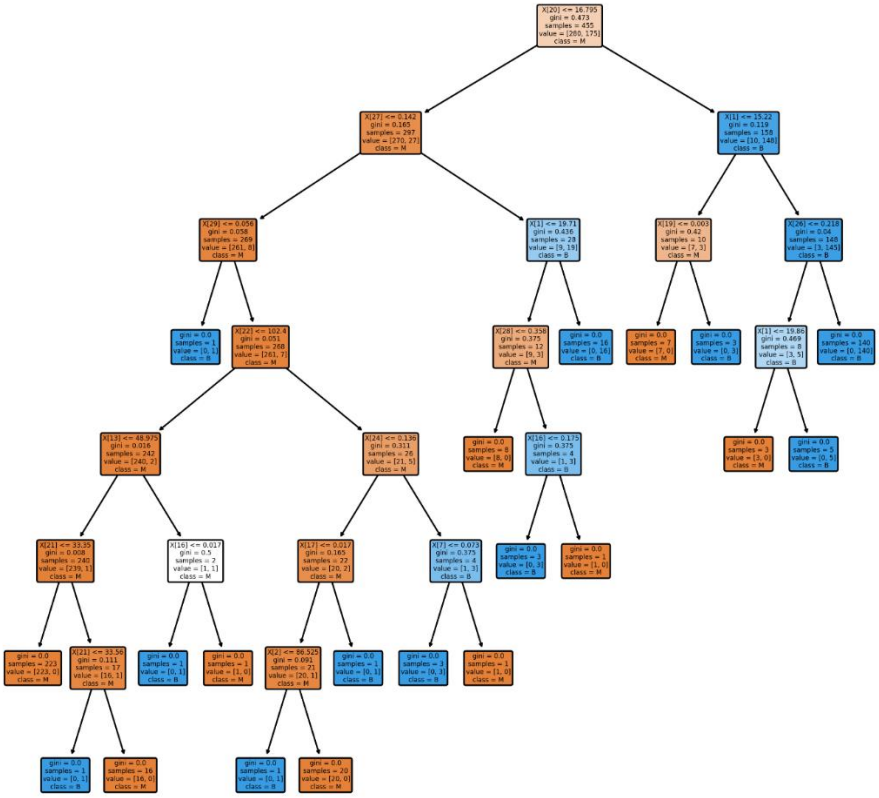
- Confusion matrix is a critical tool for evaluating classification models. It quantifies the number of true positive, true negative, false positive, and false negative predictions.

These metrics quantify the effectiveness of the model in distinguishing between benign and malignant breast tumors.

Results - Raw results

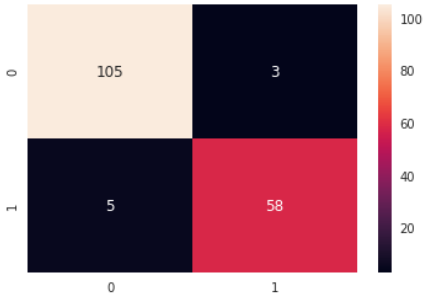
I trained a Decision Tree model in order to do prediction and classify if the patient's tumor is benigne or malignant. If it is malignant, it means that the patient has breast cancer.

First implementation result :

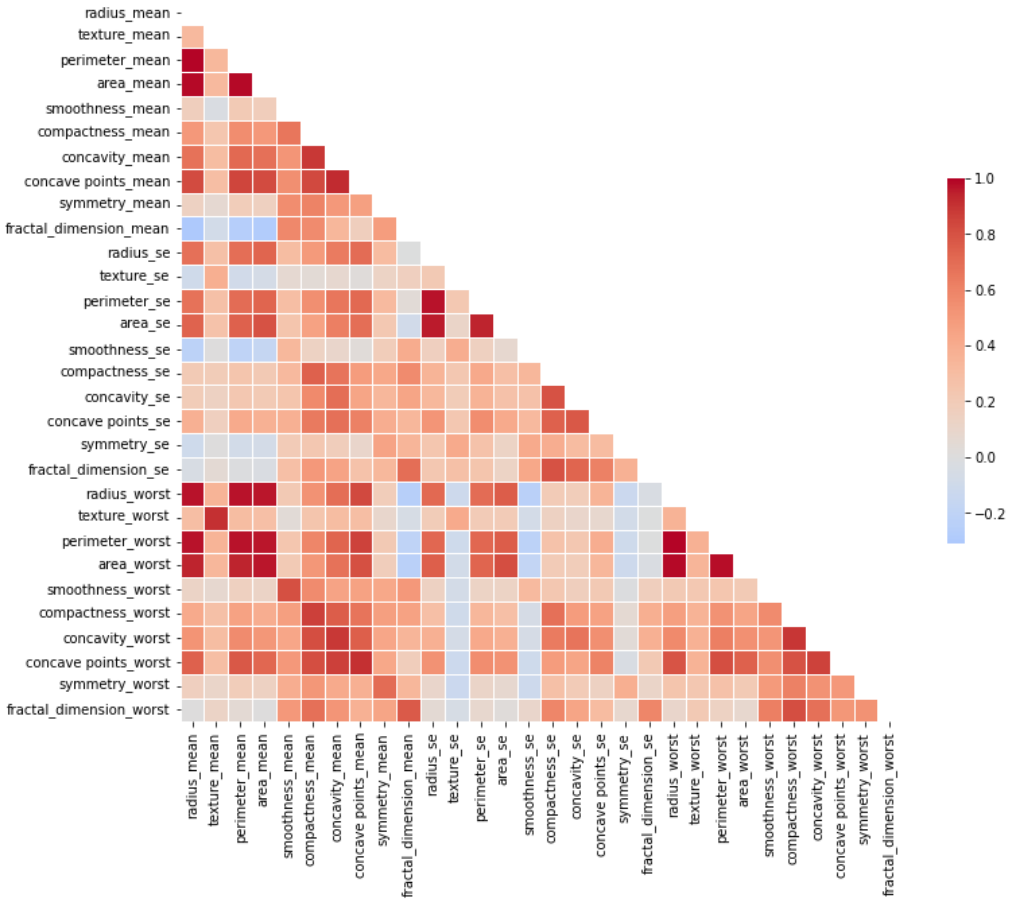


The accuracy achieved with this first implementation is 95.58% which is almost perfect.

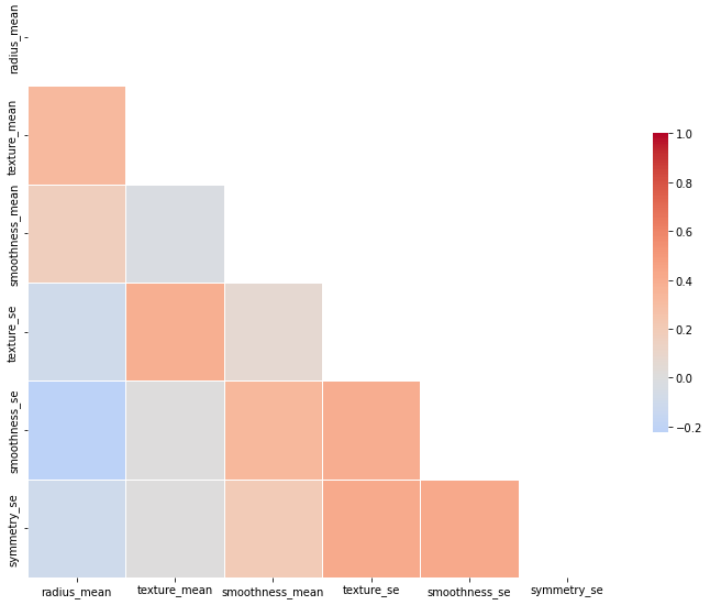
The confusion matrix :



A second implementation has been done. The function **feature_correlation_map** applied to the Breast Cancer Dataset gave this map :

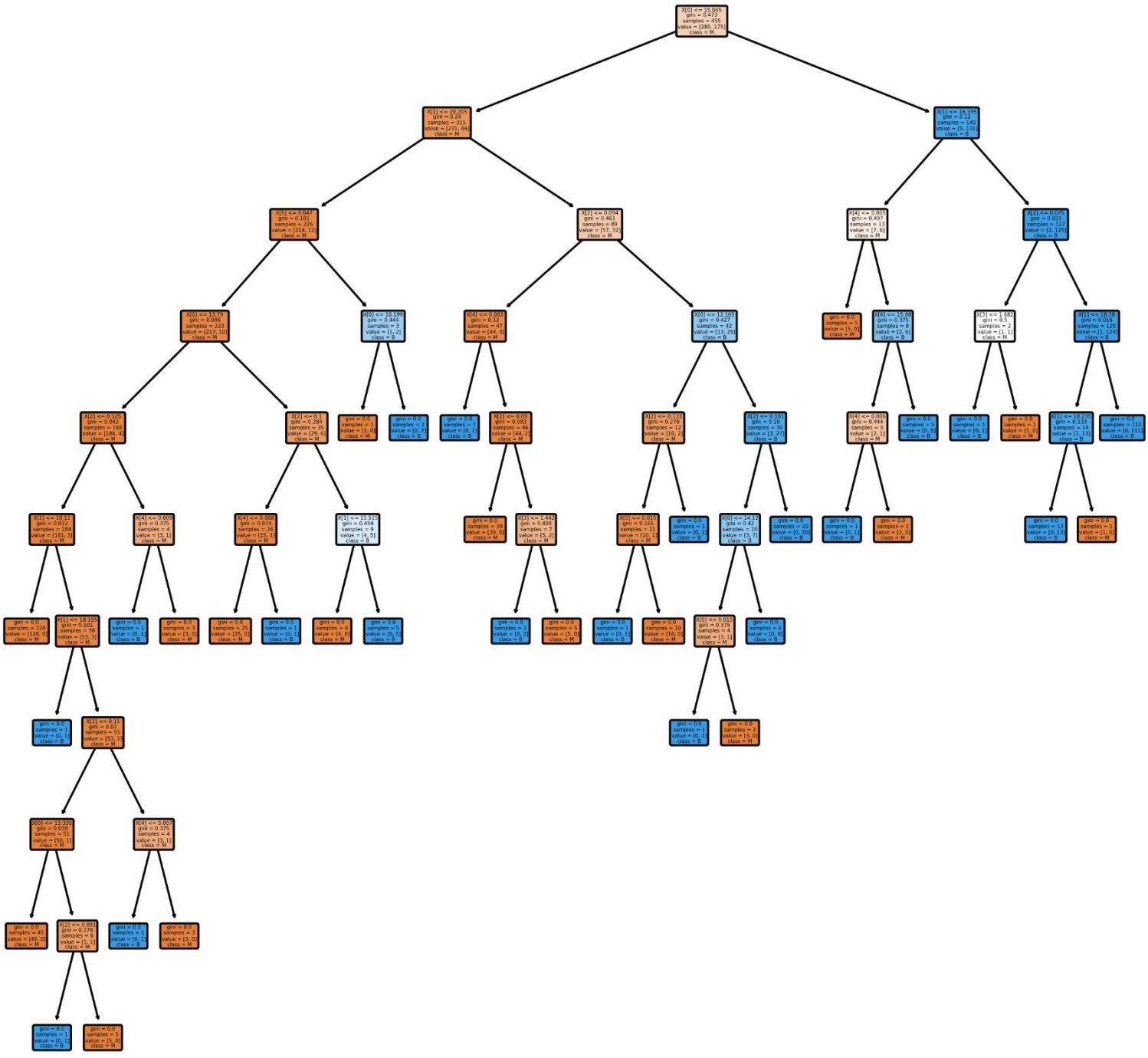


After retaining only the features that has a correlation higher than the threshold of 0.5, we obtain a new correlation map :



Radius_mean, texture_mean, smoothness_mean, texture_se, smoothness_se and symmetry_se are the features which are not correlated.

Second implementation of the decision tree model with the features « filtered » :



The accuracy of this implementation is 88.4%.

Discussion - Interpretation of the results

The accuracy achieved by my first implementation is 95.58%, slightly surpassing the 94.3% reported in the referenced article (https://ieeexplore.ieee.org/document/9442043/).

Despite this high accuracy, it's crucial to acknowledge the presence of bugs in my code, potentially leading to inaccurate results. One identified issue lies in the prior function, which can require modification. The breast cancer dataset's classes, denoted as 'B' for benign and 'M' for malignant, are of string type. This contrasts with the integer type classes in the iris dataset used in the first assignement on decision trees. Consequently, the prior function needs adjustments to handle string-type classes.

As we can see in the correlation map, **Radius_mean, texture_mean, smoothness_mean, texture_se, smoothness_se and symmetry_se** are the features which are not correlated which is nice for our implementation. When features are highly correlated, the model may encounter multicollinearity issues, making it difficult to discern their individual contributions to predictions. In this case, uncorrelated features can potentially lead to a more robust and interpretable model. In the second implementation, I obtained an accuracy of 88.4% which is lower than the first implementation without features selection.

This result can be due to a loss of information through the feature selection process. It's possible that the features removed during feature selection contained some relevant information for the prediction task, even if they were not highly correlated with other features. The retained features may not capture all the nuances present in the data, leading to a decrease in accuracy.

Then, thanks to the decision tree plot, I can visualize the resulting decision tree structure. This visualization gives an understanding of the model but also facilitates its use by medical professionals who may need to interpret the model's predictions in a clinical context.

Conclusion - Conclude on the problematic

In this research project, I implemented decision trees as a predictive model to distinguish the difference between benign and malignant breast tumors. The goal was to develop an accurate and interpretable tool that could help in early breast cancer detection and improve patient outcomes.

Therefore, the project revealed the following insights :

- Certain features had a significant impact on the model's predictions, providing insights into the clinical and pathological factors that are most indicative of breast cancer (such as **Radius_mean, texture_mean, smoothness_mean, texture_se, smoothness_se and symmetry_se**).
- The decision tree's structure, as visualized, can be used as a valuable tool for medical practitioners. They can follow the tree's branches to understand how the model arrives at its classifications, providing a level of transparency and trust that is essential in a healthcare setting.

In summary, the project contributes to the ongoing efforts to improve the accuracy and interpretability of breast cancer prediction. This type of project aim to empower healthcare professionals to make more informed decisions and enhance patient care.

References

- Chronic kidney disease diagnosis usin decision trees algorithms (2021) (https://bmcnephrol.biomedcentral.com/articles/10.1186/s12882-021-02474-z)
- Simple Prediction of Type 2 Diabetes Mellitus via Decision Tree Modeling (2017) (https://brieflands.com/articles/ircrj-10657.pdf)
- Decision tree model in the diagnosis of breast cancer (2017) (https://ieeexplore.ieee.org/document/8789297)
- Early Prediction of Heart Disease Using Decision Tree Algorithm (2017) (https://www.researchgate.net/profile/Safish-Mary/publication/315023624_Early_Prediction_of_Heart_Disease_Using_Decision_Tree_Algorithm/links/58c84b57aca2723ab16eba60/Early-Prediction-of-Heart-Disease-Using-Decision-Tree-Algorithm.pdf)
- First assignment on decision trees
- Dataset « Breast cancer Wisconsin » : https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data