COMP 551 - Miniproject 1

Brendon McGuinness - Shubhika Ranjan - Pauline Riviere
Group 63

February 5, 2021

Abstract

The project required us to work on the two benchmark health datasets to investigate the performance of two machine learning models. After data cleaning and exploration, datasets were shuffled before splitting into training and testing datasets. Based on these newly created datasets k-nearest neighbours (KNN) and decision tree algorithms were tested and analysed for different hyperparameters. The efficiency and accuracy of the models were compared to determine the best suited model and parameters for the provided datasets. We found that KNN model performed better than the decision tree approach on both datasets. The handling of missing data and the selection of features had more impact on the accuracy of the decision tree model than on KNN.

1 Introduction

The purpose of this project was to understand the implementation of K-Nearnest-Neighbours and Decision-Tree implementation from scratch for the two provided datasets of Wisconsin breast cancer database and hepatitis database. The Wisconsin breast cancer dataset presents data on cytologic evaluation of breast mass classifed as benign or malign. The objective of the classification is to be able to predict the nature of the breast mass based on features of its cells as done previously in [3]. The hepatitis dataset presents clinical and biochemical characteristics of patients with hepatitis and their survival. The objective of the classification is to predict the survival of the patient based on various clinical and biochemical features as done previously in [2].
The first task was to clean the data from features with missing instances and to explore the distribution of classes and features. After data-exploration, the datasets were divided into training and testing datasets. These training sets were used to build the KNN model and decision tree. Two approaches of data-cleaning were analysed and evaluated based on these two models. Based on numerous iterations for different hyperparameters in each of these models, we can say that for both the datasets, KNN performed better with 97.1% and 85% accuracy rate for breast cancer and hepatitis dataset, respectively, when compared to the decision tree with 94.7% and 75% accuracy.

2 Datasets

The two data-sets analyzed in this project, obtained from UCI Machine learning Repository, are Wisconsin breast cancer Wisconsin(diagnostic) dataset and Hepatitis dataset. As mentioned in the repository, the Cancer data-set describes features from digitized images of fine needle aspiration of breast mass which describes the characteristics of the mass cells. The class label is the nature of the mass, i.e. "benign" or "malign". The dataset provided had 699 instances and 11 attributes with 16 total missing data (for only one attribute). This data was handled and compared in 2 ways, firstly by substituting the missing data by the mean of the corresponding attribute with the remaining data. And secondly, by removing all the instances with missing data, that resulted in 683 instances and 11 attributes.

Hepatitis dataset describes 19 clinical and biochemical numerical or binary features of patients suffering from hepatitis. The class label is whether they survived or not. It 155 instances with around 167 fields missing (spread across the entire matrix). Hepatitis data was again cleaned by two methods, firstly by substituting the missing data by mean (for continuous variables) or by mode (for binary variables). Secondly, by removing the instances with missing data, that resulted in 80 instances for all 19 attributes.

Regarding ethical concerns, these datasets don't contain any identifiying feature like date of birth, or city. In that context, they are suitable for widespread research based on data repository.

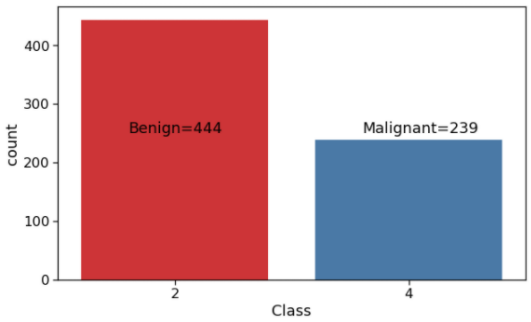


Figure 1: Distribution of benign and malignant classes over the cancer dataset

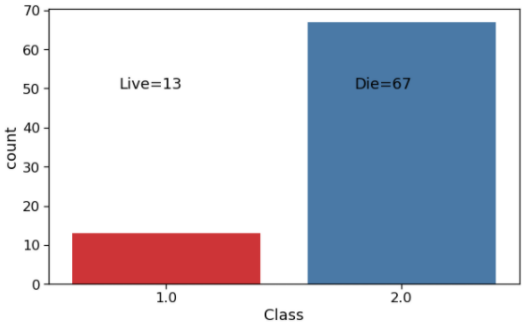


Figure 2: Distribution of die and live class over the hepatitis dataset

These data-sets were separately explored statistically to determine the dependence or the distribution of the class across the fields as described in 1 and 2. Figure A.1 and figure A.2 in Appendix show the distribution of all the attributes on the basis of the two classes in their respective data-sets.

3 Results

3.1 Compared accuracy of the algorithms on the two datasets

As displayed in Table 1, the KNN model performed better than the Decision Tree model in the two datasets. However, the recall rate was slightly higher for the Decision Tree algorithm on the Breast cancer dataset.

Model	Best accuracy achieved (%)	Recall (%)	Precision (%)
Breast cancer dataset			
KNN algorithm	97.1	95.8	100
Decision Tree algorithm	94.7	98.1	95.5
Hepatitis dataset			
KNN algorithm	85.0	83.3	71.4
Decision Tree algorithm	75	83.3	55.5

Table 1: Compared performance of the KNN and Decision Tree algorithms on the two datasets

3.2 Testing different K values

As shown in Figure 3 and 4, with higher k values, the training data accuracy decreased and the test data accuracy initially increased and then decreased. For both datasets, the k value associated with the lowest train error was 1 and with the lowest test error was 3. A k value close to 1 will overfit the data (low train error but high test error) and a high k value (more than 10) will underfit the data (high train and test error).

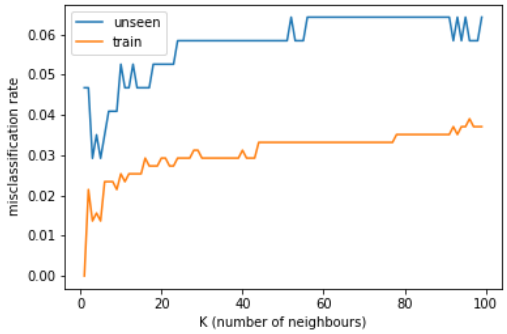


Figure 3: Performance of KNN algorithm on breast cancer dataset over a range of K values

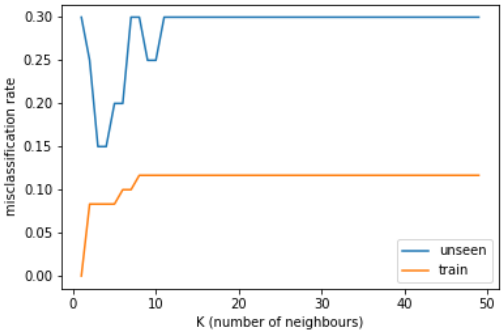
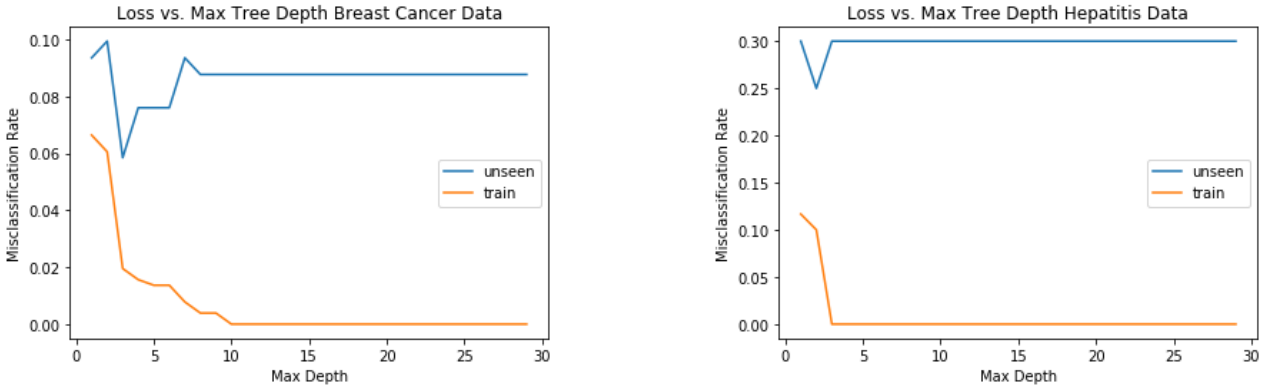


Figure 4: Performance of KNN algorithm on hepatitis dataset over a range of K values

3.3 Testing different maximum tree depths

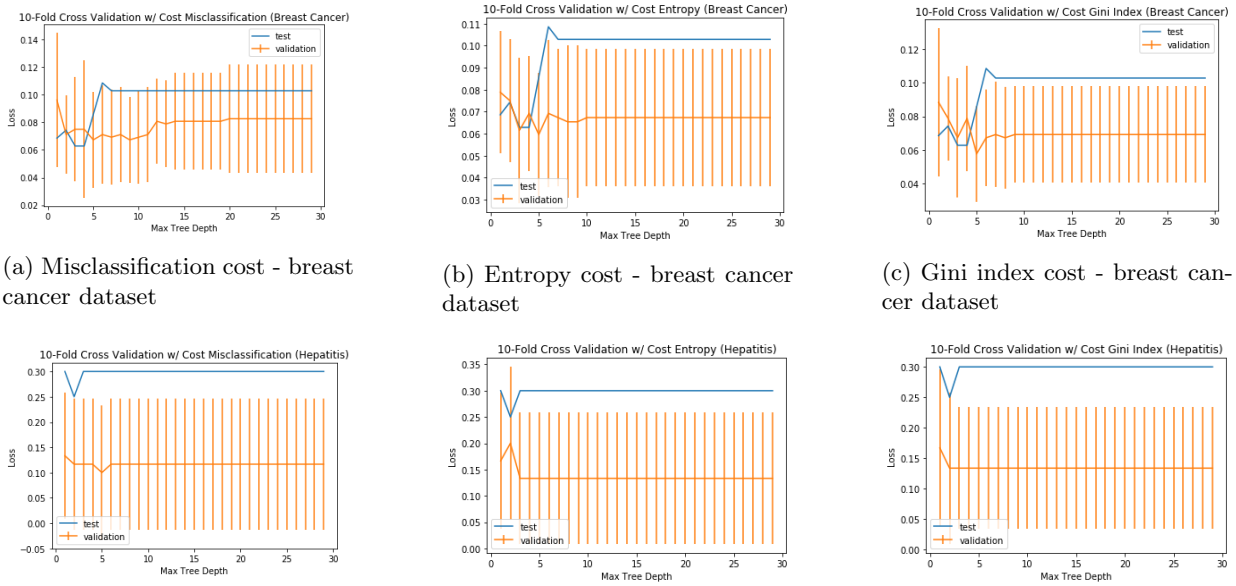
As shown in Figure 5 the optimal maximum tree depths are a depth of three for the breast cancer dataset and two for the hepatitis dataset. A higher maximum tree depth would then overfit the data suggested by the decrease in error rate on training data with an increase in error rate on testing data. Additionally, to check if our hyperparameter tuning was biased towards the testing data, we performed a 10-fold cross-validation (Figure 6). Minimum error rates are similar for the average validation set as for the testing data although there is slight variation. The maximum tree depth that minimizes the average validation error is what should be used when testing additional unseen data.



(a) Breast cancer dataset

(b) Hepatitis dataset.

Figure 5: A comparison of how maximum tree depth affects the performance of our decision tree on both the breast cancer (a) and hepatitis datasets (b).



(a) Misclassification cost - breast cancer dataset

(b) Entropy cost - breast cancer dataset

(c) Gini index cost - breast cancer dataset

(d) Misclassification cost - hepatitis dataset

(e) Entropy cost - hepatitis dataset

(f) Gini index cost - hepatitis dataset

Figure 6: 10-fold cross validation with different cost functions.

3.4 Trying different distance/cost functions

KNN algorithm

Breast cancer dataset: the use of Manhattan distance was associated with a lower accuracy of the model compared to the use of Euclidean distance (96.5% versus 97.1%, respectively).

Hepatitis dataset: after min-max normalization of the features, the use of Euclidean distance for categorical and continuous features or the use of Manhattan distance for categorical and continuous features did not affect the accuracy of the model.

Decision Tree algorithm
We used misclassification error, entropy, and gini index as cost functions when running our decision tree algorithm. For the breast cancer dataset, the misclassification cost function gave the highest accuracy (94.7%). However for the hepatitis dataset, the entropy cost function gave the highest accuracy (75%). These results are before any feature selection methods.

3.5 Decision boundary

KNN algorithm
Based on class correlation analysis (see figures A.5 and A.6 in Appendix), for each dataset, we selected 2 features among the most correlated with class labels to be displayed on the decision boundary plot (the uniformity of cell size and uniformity of cell shape for the breast cancer dataset and albumin and bilirubin for the hepatitis dataset). For the Breast cancer dataset, we can see that the decision boundary is well defined at the bottom-left corner. However, the lace-like appareance of the decision boundary may suggest overfitting of the data. Only one patient is misclassified because of values more similar to the other class patients for theses features. For the hepatitis dataset, the boundary between die and live patients is less linear. It seems that the model will predict the "die" label mostly for the bottom-left and the upper-right corner.

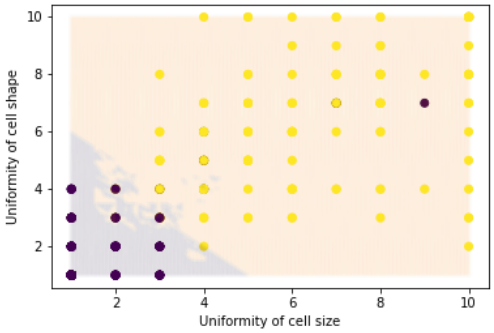


Figure 7: Decision boundary plot of KNN algorithm for breast cancer dataset

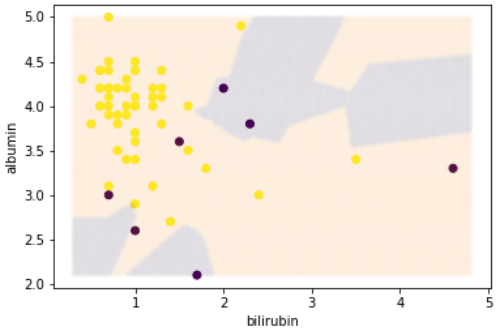
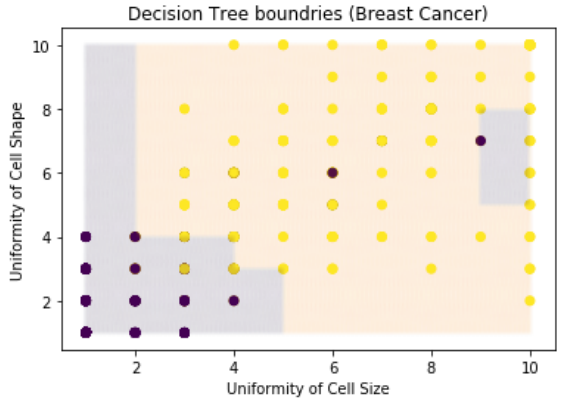
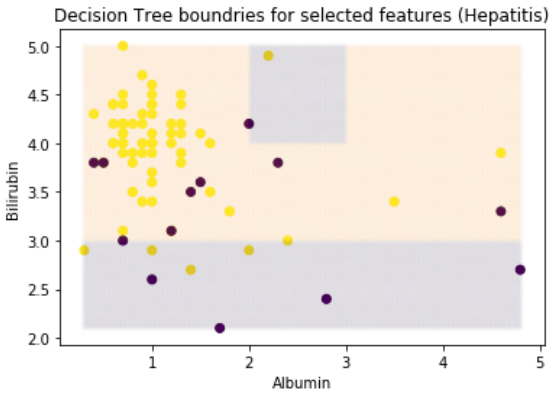


Figure 8: Decision boundary plot of KNN algorithm for hepatitis dataset

Decision tree algorithm
Based on same class correlation analysis ran for the KNN section (see figures A.5 and A.6 in Appendix), we plotted the decision boundaries for the decision tree algorithm. We see that the decision boundaries are all rectangular which works alright with the breast cancer dataset. However, for the hepatitis dataset it does not partition the heterogenous data that well for the selected features (albumin and bilirubin).



(a) Breast cancer dataset



(b) Hepatitis dataset.

Figure 9: Decision boundaries plotted for both datasets. The features uniformity of cell size and shape were used in the plot for the breast cancer data (a). The features albumin and bilirubin were used for the hepatitis dataset (b).

3.6 Additional experiments

KNN algorithm
We tried some features manipulations in order to optimize the model performance:

- replacing missing data by mode in case of categorical feature or mean in case of continuous feature;
- remove from the model features that had a lower correlation with the class, that is "mitoses" for breast cancer dataset and "SGOT" and "liver firm" for hepatitis dataset (see figures A.5 and A.6 in Appendix);
- for the hepatitis dataset where features have different scales:
  - normalize all the features with min-max normalization
  - rescale the albumin feature which showed one of the best correlation with the class (see figures A.5 in Appendix) but had small scale compared to others

Results are presented in the table A.1 in Appendix. None of them improved the accuracy beyond 97.1%.

Decision Tree algorithm
We tried different data preprocessing methods before inputting the data into our decision tree, removing missing data and imputation (i.e. filling the missing data based off of statistical properties of the dataset). Removing the data gave the best accuracy for both datasets as using mean/mode to refill missing data likely biased the training dataset. Additionally we removed the features that had a lower correlation with the class in the same fashion as for the KNN algorithm. This selection of features led to higher accuracy for the decision tree algorithm on both datasets as described in Table A.2.

4 Discussion and Conclusion

Overall, the KNN model performed better that the Decision Tree model for the two datasets, as shown in Table 1. However, the recall rate was 2.3% higher with the Decision Tree model in the Breast cancer dataset which will lead to less false negatives. In a diagnostic test for cancer, you would want to keep your false negative rate as low as possible. The accuracy rates we obtained are consistent with previously described work by Ashraf et al. [1] and Potdar et al [4]. This suggests that our testing set was not unrepresentative of our total data, which we validated by running a cross-validation (Figure 6).

For preprocessing of the data, we did use two methods: removing rows that were missing data and filling in missing data with the average of that feature (or mode if the feature was binary). Our accuracy was better when we removed missing rows (Tables A.1 and A.2), due to likely biasing the data when filling in missing data points with average. Drawing from a distribution representative of the specific feature could potentially fill in missing data points with less bias. For instance, if it was a binary feature we could fill in that data point with a Bernoulli distribution representative of that feature. Feature selection could be a way to increase the model accuracy, potentially reduce overfitting, and lower a model's complexity. However, using various feature selection methods for the KNN model on these two datasets, Ashraf et al. [1] didn't obtain an higher accuracy than what we demonstrated for the hepatitis dataset. Noteworthy, they obtained an higher accuracy for the breast cancer dataset using consistency-based subset evaluation. When feature selection was used for the decision tree model, the accuracy did increase a little bit for both datasets.

Overall, this work shows that KNN and decision tree models are useful in regards to analyzing medical records to make predictions on patients outcomes. However, when using such techniques we should be wary of biases in the data that may not reflect the true distributions as seen in the hepatitis dataset likely due to small size of the dataset thus leading to lower predictive accuracy.

5 Statement of contributions

Shubhika performed Task 1. Pauline performed Task 2 and 3 regarding KNN algorithm and class correlation analysis and Brendon Task 2 and 3 regarding Decision Tree Algorithm. All of us contributed to the writing of the report and approved the final version of the manuscript.

References

[1] Mohammad Ashraf, Girija Chetty, and Dat Tran. Feature selection techniques on thyroid, hepatitis, and breast cancer datasets. *International Journal on Data Mining and Intelligent Information Technology Applications*, 3(1):1, 2013.

[2] B. Diaconis, P. Efron. Computer-intensive methods in statistics. *Scientific American*, 248, 1983.

[3] O. L. Mangasarian and W. H. Wolberg. Cancer diagnosis via linear programming. *SIAM News*, 23(5):1–18, 1990.

[4] Kedar Potdar and Rishab Kinnerkar. A comparative study of machine learning algorithms applied to predictive breast cancer data. *International Journal of Science and Research*, 5(9):1550–1553, 2016.

6 Appendix

6.1 Figures

Statistical exploration of all the attributes in breast-cancer-wisconsin dataset

```
1. Clump_Thickness
        Training Set: Minimum: 1.0 Maximum: 10.0 Mean: 4.55078125 Median: 4.0
        Testing Set: Minimum: 1.0 Maximum: 10.0 Mean: 4.116959064327485 Median: 4.0
2. Uniformity_of_Cell_Size
        Training Set: Minimum: 1.0 Maximum: 10.0 Mean: 3.228515625 Median: 1.0
        Testing Set: Minimum: 1.0 Maximum: 10.0 Mean: 2.91812865497076 Median: 1.0
3. Uniformity_of_Cell_Shape
        Training Set: Minimum: 1.0 Maximum: 10.0 Mean: 3.31640625 Median: 2.0
        Testing Set: Minimum: 1.0 Maximum: 10.0 Mean: 2.912280701754386 Median: 1.0
4. Marginal_Adhesion
        Training Set: Minimum: 1.0 Maximum: 10.0 Mean: 2.912109375 Median: 1.0
        Testing Set: Minimum: 1.0 Maximum: 10.0 Mean: 2.584795321637427 Median: 1.0
5. Single_Epithelial_Cell_Size
        Training Set: Minimum: 1.0 Maximum: 10.0 Mean: 3.291015625 Median: 2.0
        Testing Set: Minimum: 1.0 Maximum: 10.0 Mean: 3.064327485380117 Median: 2.0
6. Bare_Nuclei
        Training Set: Minimum: 1.0 Maximum: 10.0 Mean: 3.609375 Median: 1.0
        Testing Set: Minimum: 1.0 Maximum: 10.0 Mean: 3.3508771929824563 Median: 1.0
7. Bland_Chromatin
        Training Set: Minimum: 1.0 Maximum: 10.0 Mean: 3.486328125 Median: 3.0
        Testing Set: Minimum: 1.0 Maximum: 10.0 Mean: 3.3216374269005846 Median: 3.0
8. Normal_Nucleoli
        Training Set: Minimum: 1.0 Maximum: 10.0 Mean: 2.95703125 Median: 1.0
        Testing Set: Minimum: 1.0 Maximum: 10.0 Mean: 2.608187134502924 Median: 1.0
9. Mitoses
        Training Set: Minimum: 1.0 Maximum: 10.0 Mean: 1.671875 Median: 1.0
        Testing Set: Minimum: 1.0 Maximum: 10.0 Mean: 1.3976608187134503 Median: 1.0
```

Figure A.1: Statistical analysis of the cancer dataset

Statistical exploration of all the attributes in hepatitis dataset

```
1. AGE
        Training Set: Minimum: 20.0 Maximum: 72.0 Mean: 39.266666666666666 Median: 38.0
        Testing Set: Minimum: 25.0 Maximum: 65.0 Mean: 44.85 Median: 45.5
2. SEX
        Training Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.1333333333333333 Median: 1.0
        Testing Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.15 Median: 1.0
3. STEROID
        Training Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.5166666666666666 Median: 2.0
        Testing Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.55 Median: 2.0
4. ANTIVIRALS
        Training Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.7 Median: 2.0
        Testing Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.85 Median: 2.0
5. FATIGUE
        Training Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.3833333333333333 Median: 1.0
        Testing Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.25 Median: 1.0
6. MALAISE
        Training Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.65 Median: 2.0
        Testing Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.5 Median: 1.5
7. ANOREXIA
        Training Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.85 Median: 2.0
        Testing Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.85 Median: 2.0
8. LIVER_BIG
        Training Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.8333333333333333 Median: 2.0
        Testing Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.85 Median: 2.0
9. LIVER_FIRM
        Training Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.5166666666666666 Median: 2.0
        Testing Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.55 Median: 2.0
10. SPLEEN_PALPABLE
        Training Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.8666666666666667 Median: 2.0
        Testing Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.65 Median: 2.0
11. SPIDERS
        Training Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.75 Median: 2.0
        Testing Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.5 Median: 1.5
12. ASCITES
        Training Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.9 Median: 2.0
        Testing Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.7 Median: 2.0
13. VARICES
        Training Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.9166666666666667 Median: 2.0
        Testing Set: Minimum: 1.0 Maximum: 2.0 Mean: 1.75 Median: 2.0
14. BILIRUBIN
        Training Set: Minimum: 0.4 Maximum: 4.6 Mean: 1.1483333333333332 Median: 1.0
        Testing Set: Minimum: 0.3 Maximum: 4.8 Mean: 1.44 Median: 1.0
15. ALK_PHOSPHATE
        Training Set: Minimum: 26.0 Maximum: 280.0 Mean: 98.6 Median: 85.0
        Testing Set: Minimum: 50.0 Maximum: 243.0 Mean: 115.85 Median: 100.0
16. SGOT
        Training Set: Minimum: 14.0 Maximum: 420.0 Mean: 82.0 Median: 54.5
        Testing Set: Minimum: 19.0 Maximum: 269.0 Mean: 82.1 Median: 66.0
17. ALBUMIN
        Training Set: Minimum: 2.1 Maximum: 5.0 Mean: 3.8699999999999997 Median: 4.0
        Testing Set: Minimum: 2.4 Maximum: 4.7 Mean: 3.7649999999999997 Median: 3.9
18. PROTIME
        Training Set: Minimum: 0.0 Maximum: 100.0 Mean: 63.483333333333334 Median: 65.0
        Testing Set: Minimum: 29.0 Maximum: 100.0 Mean: 59.6 Median: 54.5
```

Figure A.2: Statistical analysis of the hepatitis dataset

ViolinPlot to demonstrate the distribution of all the attributes for benign and malignant classes in breast-cancer-wisconsin dataset

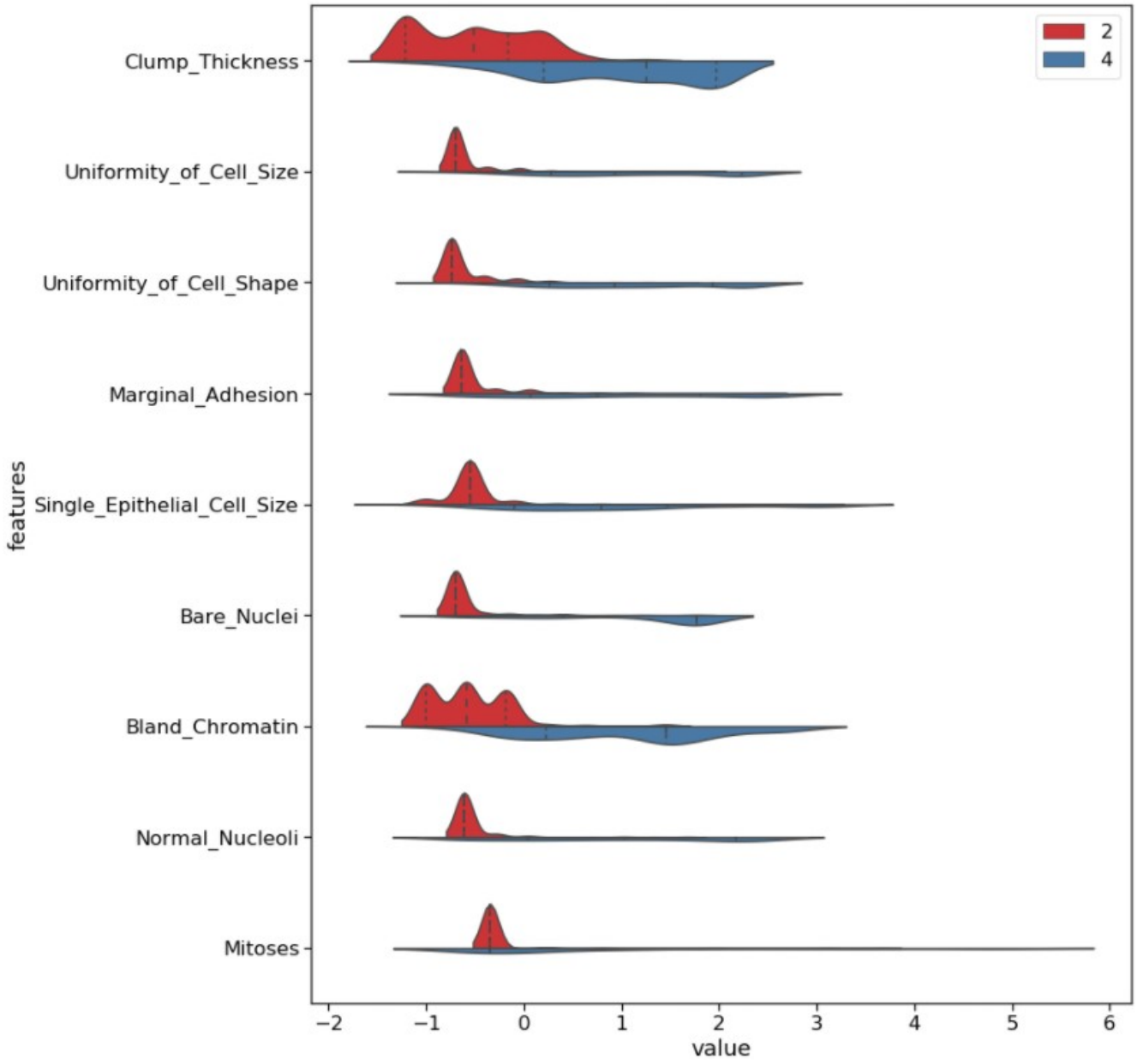


Figure A.3: Distribution of attributes for benign and malignant classes over the cancer dataset

ViolinPlot to demonstrate the distribution of all the attributes for cells that die or live in hepatitis dataset

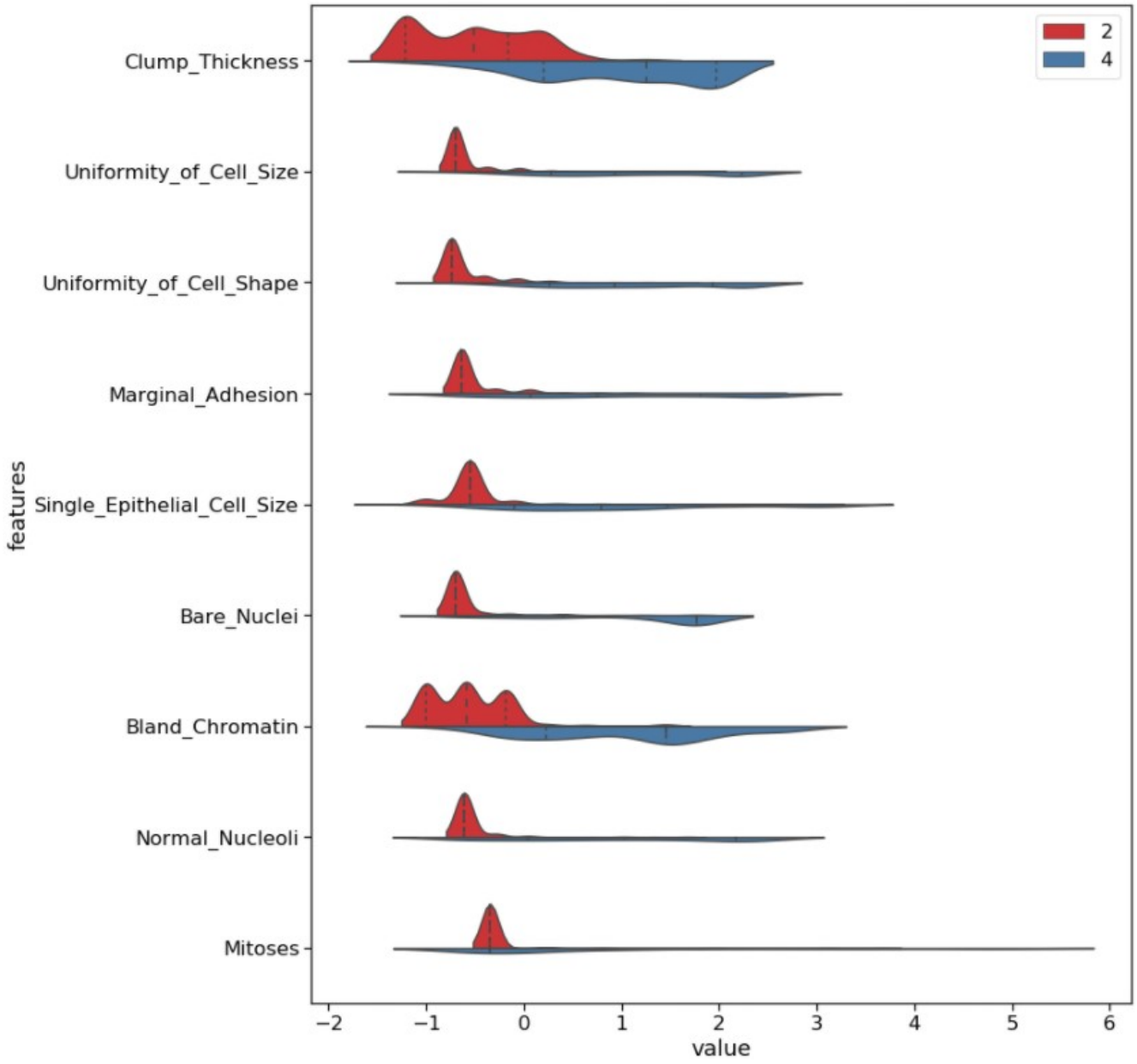


Figure A.4: Distribution of attributes for benign and malignant classes over the cancer dataset

Correlation analysis of the features and the class

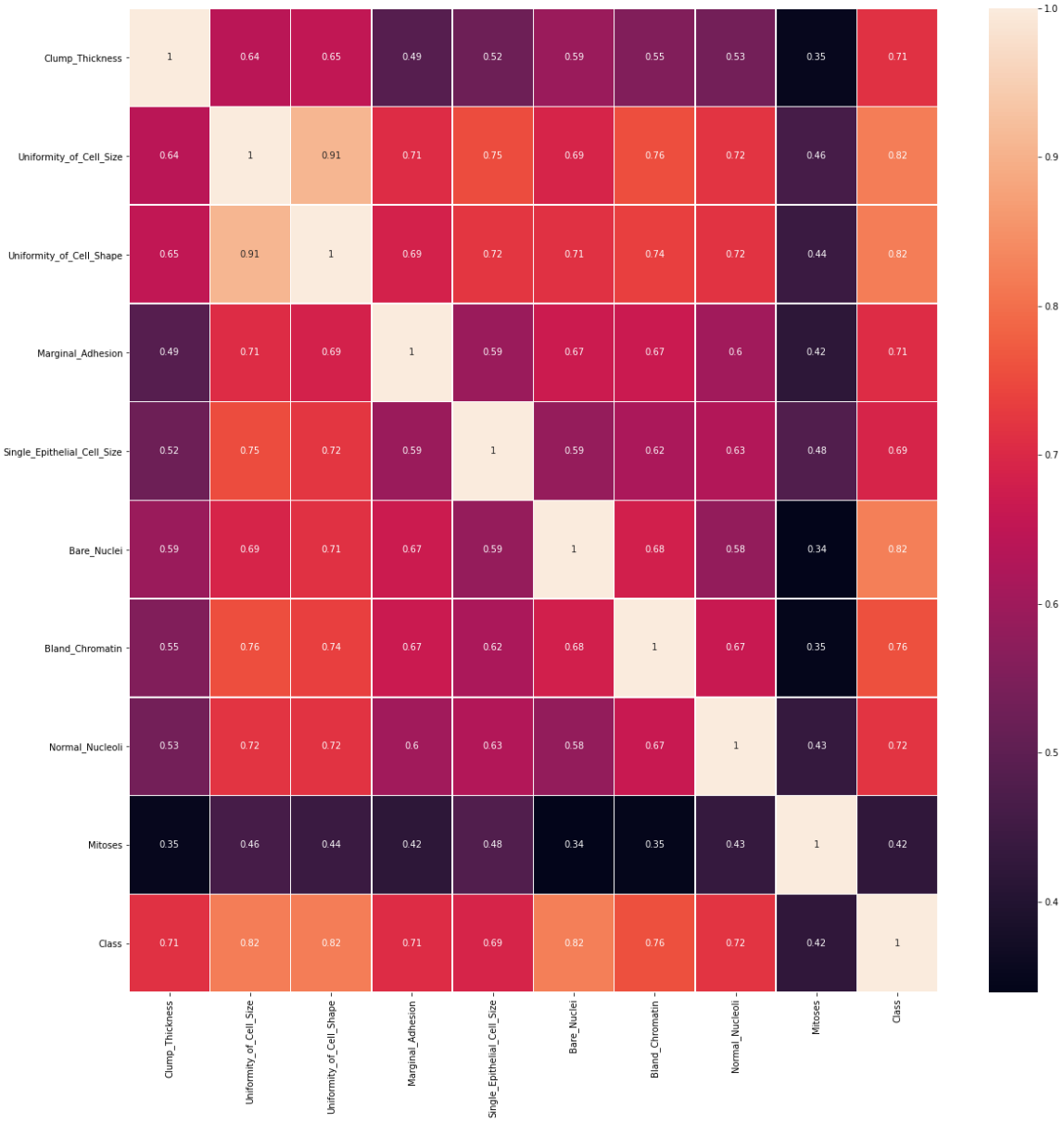


Figure A.5: Heatmap of the correlation between all features and the class in the breast cancer dataset

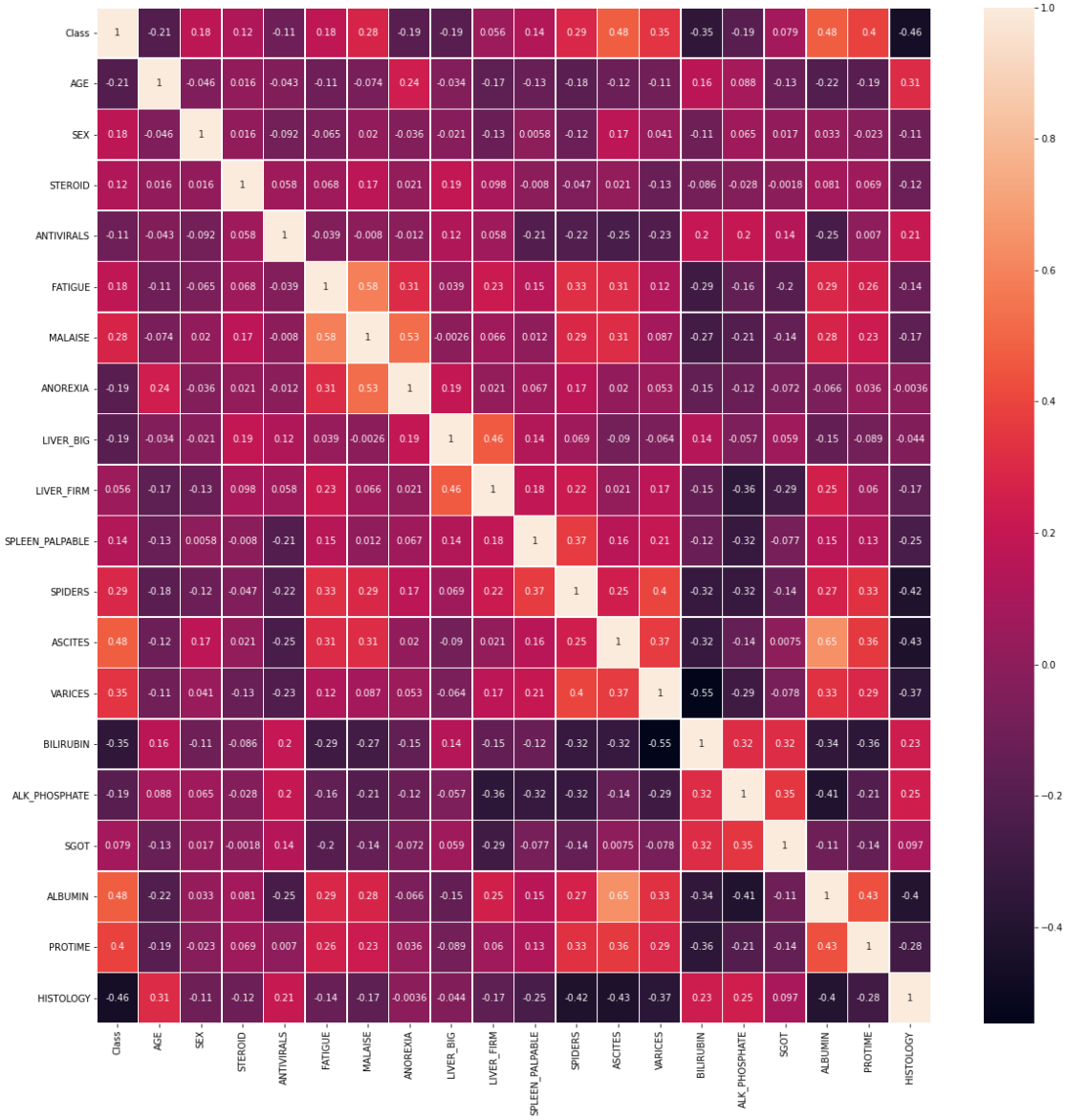


Figure A.6: Heatmap of the correlation between all features and the class in the breast cancer dataset

6.2 Tables

Model	Breast cancer dataset, accuracy (%)	Hepatitis dataset, accuracy (%)
Standard	97.1	85.0
Replacing missing data	96.0	74.4
Removing features with low correlation	97.1	85.0
Rescaling feature with high correlation	Na	85.0

Table A.1: Compared performance of the KNN algorithm with different settings

Model	Breast cancer dataset, accuracy (%)	Hepatitis dataset, accuracy (%)
Standard	94.7	75.0
Replacing missing data	93.7	74.4
Removing features with low correlation	96	76.9

Table A.2: Compared performance of the KNN algorithm with different settings