COMP 551 – APPLIED MACHINE LEARNING – WINTER 2022

MINIPROJECT 1:
GETTING STARTED WITH MACHINE LEARNING

Frida-Cecilia Acosta-Parenteau – 260870444
Karl Michel Koerich – 260870321
Simon Nakane Marcil – 260926522

McGill University, Montréal, QC, Canada

ABSTRACT

The automation of disease screening and prediction of future outcomes of a disease in health sciences is one of the main subjects of interest in applied machine learning. Large amounts of data are made available publicly to help data scientist find the best applied machine learning models to make the best predictions. This project is about implementing two models of classifier: k-nearest neighbours (K-NN) and decision trees. We run these two algorithms on two distinct datasets: the Diabetic Retinopathy Debrecen dataset where the goal is to predict whether a patient has signs of diabetic retinopathy or not, and the Hepatitis dataset, where the goal is to predict whether a patient with Hepatitis dies or survives. We compare the performance of the two models by estimating their accuracy. Our K-NN model accurately predicts 79% and 68% of the classes for the Hepatitis and Diabetes datasets, respectively, while our decision tree correctly predicts 82% and 68% of the classes for the same datasets. We test different number of neighbours for KNN for the euclidean and manhattan distances. Whereas for decision tree, we test different tree depths for three cost functions: misclassification, entropy, and gini index.

***Index Terms*—** Decision tree, Hepatitis, K-nearest neighbours, Diabetic retinopathy.

1. INTRODUCTION

K-Nearest Neighbour (K-NN) and Decision Tree are two well-known classifiers in machine learning (ML). The project starts by implementing both methods from scratch in order to analyze and compare their performance.

The K-NN model is a lazy learner which classifies based on the closest data points in the training set. Variation of the model depends on different distance functions used to calculate the K closest training points. Additionally, this K hyper-parameter can be modified to indicate how many training points it should consider from the testing point. In comparison, Decision Tree classifies by splitting features into regions. These regions are separated with conditional boundaries which are determined to optimize the cost function. This cost function can vary the model and specifying the maximum depth of the tree determines how many decision boundaries it can have.

The data used for this exercise are acquired from the UCI Machine Learning Repository: the Hepatitis dataset and the Diabetic Retinopathy Debrecen dataset. This data is imported and cleaned before running experiments. Both datasets have two classes to run a binary classification with each model. The objective is not only to observe performance difference between the two models but to also vary model parameters to see changes in accuracy. Parameters include testing different K and maximum tree depth values and different distance/cost functions. Finally, the decision boundaries for each model are shown for some of the key features in the dataset. After testing the K-NN with different hyper-parameters and distance functions, our model accurately predicts 79% and 68% of the classes for the Hepatitis and Diabetes datasets, respectively, while our decision tree correctly predicts 82% and 68% of the classes for the same datasets.

2. DATASETS

2.1. Hepatitis Dataset

The Hepatitis Dataset contains personal information of medical patients with hepatitis. It consists of 155 instances (different patients) with 19 attributes each. The attributes are either binary or decimal numbers, and they correspond to different characteristics of the person's health profile. Each instance is classified according to their outcome after having hepatitis: 'DIE' or 'LIVE'. More information about the dataset can be found in its website: http:archive.ics.uci.edumldatasetsHepatitis.

To clean the dataset, we first verified the frequency of

missing data (rows contains at least one missing value) and found that 75 out of 155 (48.4%) attributes were incomplete. Therefore, eliminating faulty rows would result in a considerable data loss. To solve this issue, we verified the frequency of missing data per attribute (column). We found that several instances were incomplete, but most notably, the attributes ALK PHOSPHATE, ALBUMIN, and PROTIME, had most missing values, with 29, 16, and 67, respectively. With this knowledge, we removed these attributes from the dataset and recounted the rows with missing data. The number dropped from 75 to 18, meaning that 57 (35.8%) instances had data points missing in at least one of the these columns. We then tried removing each of the three columns alone and all possible pairs among them. By calculating the number of data points lost by the removal of each column and subsequently the rows with missing attributes, we concluded that eliminating ALK PHOSPHATE, ALBUMIN, and PROTIME, would yield the lowest data loss. Following the removal, we proceeded with deleting the 18 instances that still had missing values. The dataset was reduced from (155x20) to (137x17), meaning a 25% data loss (including the removal of missing data points). Finally, we transformed all values into numerical attributes and normalized them.

With the dataset clean, we verified the class distribution, and indeed it was imbalanced: 26 instances were 'DIE,' while 111 were 'LIVE.' The following step was to plot the correlation between attributes and the class. The 4 attributes that showed the highest correlation can be seen in Figure 1. Having an unbalanced dataset with weak correlations can reduce the ability of the model to properly learn features related to the class less represented. This can cause the model to wrongly predict more instances of the predominant class (in this case, 'LIVE').

2.2. Diabetic Retinopathy Dataset

The data set was imported into pandas data frames. In the DR (diabetic retinopathy) dataset, for the Class column we defined no sign of DR (entry " b'0' ") as 0 and sign of DR as 1 (entry " b'1' " in the Class column. We found five duplicated instances which we deleted to avoid them impacting the results. Further, one attribute given in the raw data is the quality assessment of the data. We deleted the four instances which were categorized as bad quality, since these results were not reliable. Since the K-NN model was evaluated with euclidean and manhattan distance, we normalized the data with min-max normalization, which forces the numerical values to be in the range [0,1]. To evaluate the data, we separated the it into both classes and plotted the distributions of the features for each class 2. We also did a heat-map of the cross-correlations between all the features and between features and the class attribute. We can see in the heat-map that attributes 2 to 7 correlate with values between [0.86,1] for each possible pair. Indeed, i =(2 , . . . , 7) stand for the number of MAs found at

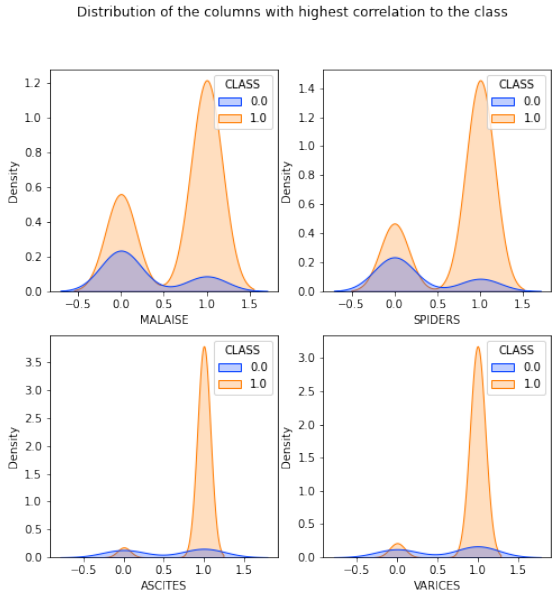

Fig. 1. All four attributes are binary, therefore they are distributed between 0 and 1.

the confidence levels α =(0.5 , . . . , 1) respectively and the same can be seen for attributes that stand for the number of exudates represented by i =(8 , . . . , 15) found at the confidence levels α =(0.1 , . . . , 0.9). This explains the decreasing standard error within each interval of features. Even if these two series of feature correlate within series, we kept them all since they are the ones with the highest correlation with the classes compared to the others i=(1,16,17,18). When working with these kinds of datasets, one could argue that estimating if a patient has diabetic retinopathy with a machine learning algorithm is too risky because a bad diagnostic could lead to dangerous consequences.

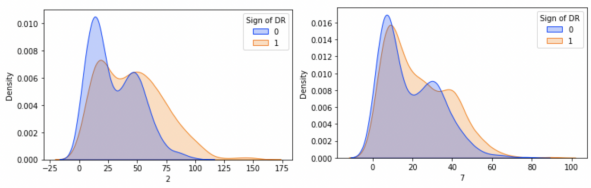

Fig. 2. Distributions for feature 2 and 7 that correspond to the number of macroaneurysms

(Diabetic Retinopathy Debrecen dataset): https://archive.ics.uci.edu/ml/datasets/Diabetic+Retinopathy+Debrecen+Data+Set

3. RESULTS

A set of experiments were conducted with each dataset to show case the performance of the K-NN and the Decision Tree model.

3.1. KNN

For the K-NN model, its accuracy was measured over a range of K values using two different distance functions. The two distance functions are the Euclidean distance and the Manhattan distance function.

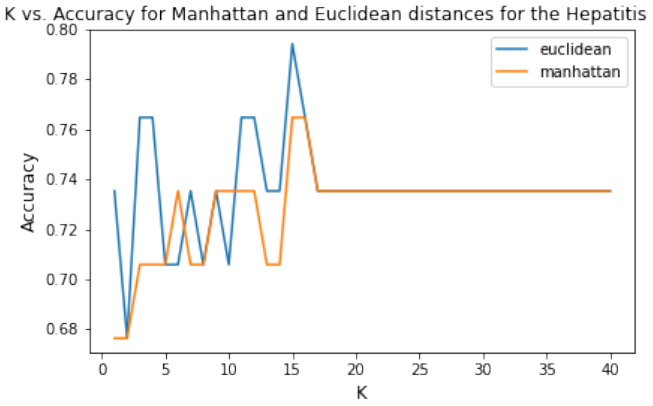

Fig. 3. Distributions of the values for two features with highest correlations with the class

First using the Hepatitis dataset, the graph in figure 3 shows the accuracy for K values in the range of 1 to 40. The best model has a high accuracy of 79.4% using a K value of 15 and the Euclidean distance function. Otherwise, the Manhattan distance function reaches its highest accuracy of 76.5% also with a K value of 15. Interestingly, the accuracy flattens out roughly at 74% after the maximum accuracy. This indicates that the hyper-parameter is overfitting. Since the parameter is overfitting, all test data is being classified to the same class and giving a constant accuracy corresponding to the distribution of that class in the test set. With this issue in consideration, the accuracy is not a preferable measure of performance for this dataset.

In the case of the Diabetic Retinopathy dataset, accuracy measurements are taken for K values between 1 and 200. The maximum accuracy obtained in figure 4 is 68.7%. This accuracy was achieved by using the Manhattan distance function with a K value of 29. The Euclidean distance performs closely with its highest accuracy of 68.4%, but requires a much higher K value of 83. Furthermore, the plot of the Euclidean function very well demonstrates the overall change in accuracy as the K parameter overfits and underfits the training set.

In the next experiment, two features with high correlation to the class were selected to plot out the decision boundary

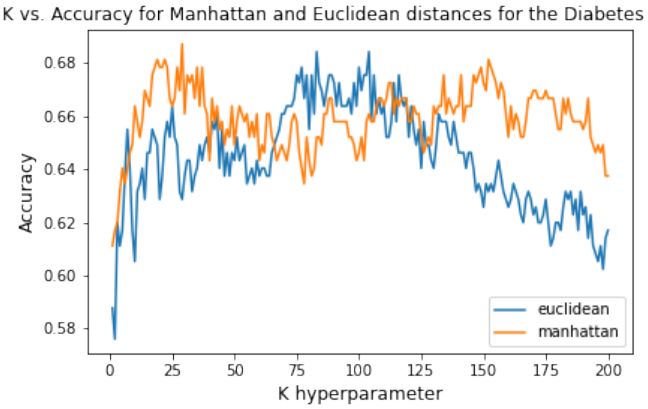

Fig. 4. Accuracy plot with different K values for Diabetic Retinopathy dataset.

for each dataset. Each model used K-NN parameters found in the previous experiment which gave the highest accuracy.

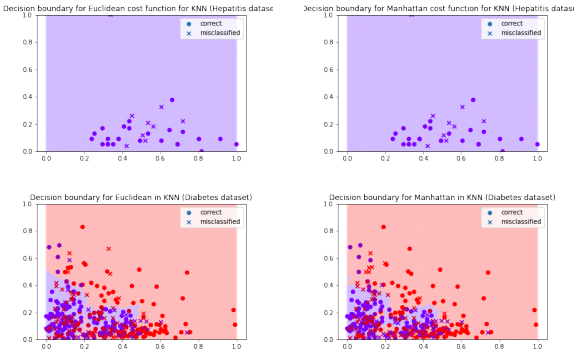

Fig. 5. Decision boundary plot for best K-NN models.

For the Hepatitis dataset, the age and the bilirubin features were used to plot out the boundaries. It was decided to disregard higher correlated features as they were binary. Consequently, it can be seen in the top two plots of figure 5 that this decision affected our boundary as it only classifies to only one class. This indicates that these features are not key features affecting the prediction. In contrast, a better decision boundary is obtained for the bottom two plots in figure 5. The Diabetic Retinopathy dataset used the number of microaneurysm at two different confidence level as features to plot its decision boundary. In the plot, it can be seen that the blue class is at the bottom left preferring lower value of microaneurysm. Otherwise, it can also be observed that there are no major difference in the boundary depending on the distance function.

3.2. Decision Tree

Similar to the K parameter, an accuracy test is performed over a range of maximum tree depth for the Decision Tree model. There are three different cost functions, namely the misclassification, the entropy, and the Gini index cost function, to run the experiments for.

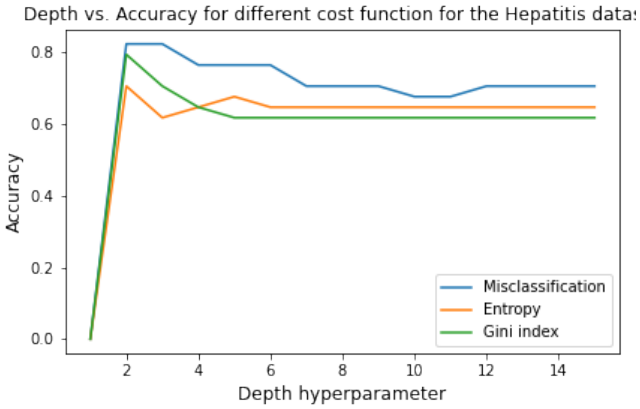

Fig. 6. Accuracy plot with different maximum tree depth for Hepatitis dataset.

Figure 6 shows the accuracy of the Decision Tree with a tree depth from 1 to 15 for the Hepatitis dataset. The highest accuracy is achieved with the misclassification cost function resulting at 82.4%. Interestingly, all cost functions reaches its best performance with a maximum depth value of 2. The highest accuracy for the entropy cost function is 70.6% and for the Gini index cost function it is 79.4

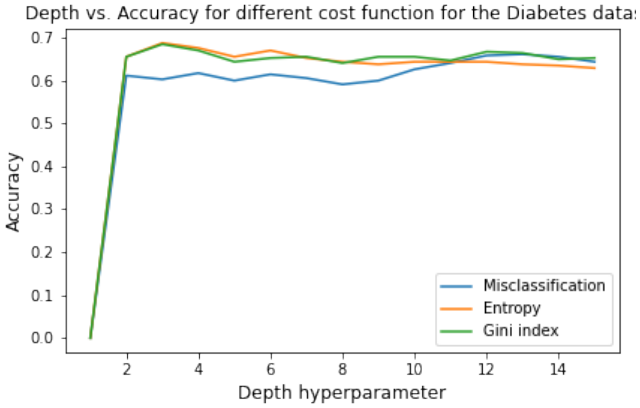

Fig. 7. Accuracy plot with different maximum tree depth for Diabetic Retinopathy dataset.

Next is the Diabetic Retinopathy dataset also measuring its accuracy for the depth range of 1 to 15. The highest accuracy obtained in figure 7 is 68.7% using the entropy cost function with a maximum depth value of 3. At the same depth value, the Gini index cost function obtains its maximum accuracy of 68.4%. Lastly, the misclassification cost function performs best with a depth of 13 with a 66.1% accuracy.

This next experiment plots the decision boundary for the best Decision Tree models. The same features as in the K-NN section were used to plot out the decision boundaries.

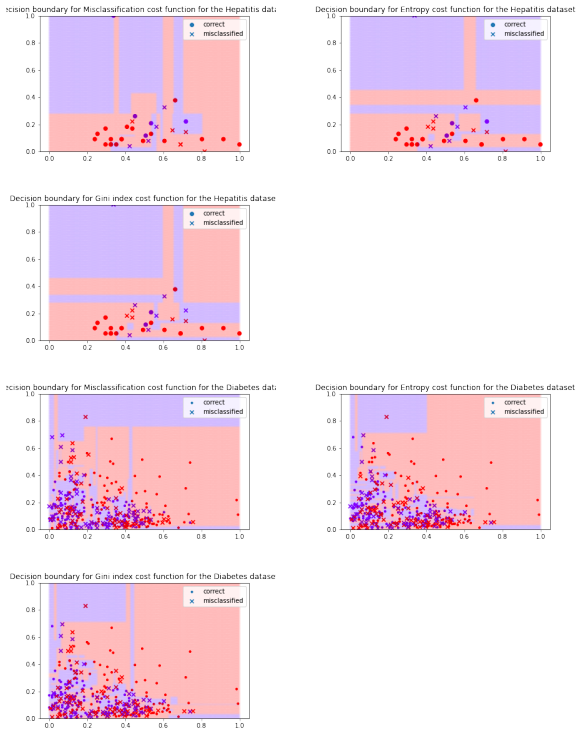

Fig. 8. Decision boundary plot for best Decision Tree models.

Here in figure 8, the top three plots are the resulting boundaries for the Hepatitis dataset and the bottom 3 plots are for the Diabetic Retinopathy dataset.

4. DISCUSSION AND CONCLUSION

For K-NN, we see a poor performance in the DR dataset in particular. This could be due to the nature of the data. For example, the highest correlation with the classes amongst all features is 0.29. This impacts particularly K-NN because it is sensitive to the quality of the data, it is more impacted more by irrelevant features and noise than the decision tree. Furthermore, K-NN is subject to the curse of dimensionality, and since we are working with 17 features in DR dataset, the neighbours could be at similar distances than any random point [1]. One interesting future route to take for the prediction of diabetic retinopathy would be to use Deep convolutional neural network image classification [2], which would remove the time and effort to be spent on extracting and selecting classification features like it was done for our dataset by Antal et al. [3]. Unfortunately, we can't compare both

performance of models as well on the Hepatitis dataset because the accuracy metric is not useful. This is because there is a class imbalance, the "Die" class has 26 instances and the "Live" class has 111 instances. Therefore, the accuracy could hide the performance of the model on predicting how many false negatives are present (i.e. true "Die" instances labeled as "Live") [4]. We therefore verified with the F1-score for the K-NN model. This score uses the precision and recall scores which do not take into account the true negatives that inflate the accuracy. These are the results for the optimal K's: for manhattan we have (K=3, F1-score= 44.4) and for euclidean we have (K=1, F1-score: 52.6). For higher K's the F1-score only decreased. This means that the K-NN model didn't perform well in predicting death, the same was observed for the Decision tree model. To remedy this problem we could only have balanced classes. Decision trees are sensitive to noise in input data. Indeed, the whole model could change if the training set is slightly modified. This affects the interpretability of the model [5].

5. STATEMENT OF CONTRIBUTIONS

Dataset cleaning: Karl and Frida-Cecilia Write-up: Everyone Implementation of models: Simon Experiments: Everyone

6. REFERENCES

[1] Kilian Weinberger, "Lecture 2: k-nearest neighbors," in *https://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote02*$_k NN.html.Cornell.$

[2] Samir S Yadav and Shivajirao M Jadhav, "Deep convolutional neural network based medical image classification for disease diagnosis," *Journal of Big Data*, vol. 6, no. 1, pp. 1–18, 2019.

[3] Bálint Antal and András Hajdu, "An ensemble-based system for automatic screening of diabetic retinopathy," *Knowledge-based systems*, vol. 60, pp. 20–27, 2014.

[4] Google, "Classification: Accuracy," in *https://developers.google.com/machine-learning/crash-course/classification/accuracy*. Google.

[5] Yury Kashnitsky, "Topic 3. decision trees and knn," in *https://www.kaggle.com/kashnitsky/topic-3-decision-trees-and-knn*. May 2021, Kaggle.