A Comparison of KNN and Decision Tree Model Performance in Health Data Classification

Parsa Yadollahi - 260869949
parsa.yadollahi@mail.mcgill.ca
Brian Hu - 260915192
brain.hu@mail.mcgill.ca
Liyun Huang - 260913905
liyun.huang@mail.mcgill.ca

Abstract

The rapid technological advances made in recent years have led to an abundance of applications in classifying health data and improving predictions using various learning algorithms. Two fundamental classification algorithms are K-Nearest Neighbors (KNN) and Decision Trees (DT), which are both supervised algorithms. In this project, we explore the performance of these two algorithms on two medical datasets: the Diabetic Retinopathy Debrecen dataset (DR) and the Hepatitis dataset (HP). We pre-process the data, train the two models and observe the accuracy, precision, and recall on subsequent test sets in order to compare and contrast the two models and discuss our findings.

The results from our project indicate that the KNN model with the Manhattan cost function produced higher accuracy than the Decision Tree on the HP dataset, due to the greater number of highly correlated features. Conversely, the Decision Tree produced higher accuracy on the DR dataset, with the Gini index as the best cost function.

1. Introduction

In this project, we were tasked with implementing two classification algorithms - K-Nearest Neighbors and Decision Trees - and comparing them on two distinct health datasets.

Both models were implemented as Python classes where the main methods of the class were used to train the model and create a prediction based on a set of inputs. We later used a ratio of 8:1:1 to split the datasets into three separate sets: train, validation, and test sets, and used the validation set to tune the hyperparameters of the models.

Data classification, especially that pertaining to health data, has been extensively studied and researched in the context of using various algorithms to best predict different diseases. One such study is *Heart Diseases Prediction System Using CHC-TSS Evolutionary, KNN, and Decision Tree Algorithm* (Saxena et al., 2018), which explores the accuracy of different data mining algorithms on the classifier.

Furthermore, considerable research is often conducted in the healthcare domain with papers often employing a single algorithm to gather and analyze the results. Two prominent examples are *Medical Health Big Data Classification Based on KNN Classification Algorithm* (Xing and Bei, 2019) and *Using Decision Tree for Diagnosing Heart Disease Patients* (Shouman et al., 2011). In each paper, emphasis is placed on the advantages and tradeoffs in their respective classification methods; the former concludes that despite the simplicity of the KNN algorithm, its efficiency is greatly reduced with large sample sizes and feature attributes, and the latter concludes that decision trees are most accurate when a range of techniques are applied to them, such as equal frequency discretization and majority voting.

2. Datasets

2.1 Hepatitis Dataset

The Hepatitis dataset (HP) consists of 155 instances with 20 attributes (including the class attribute) classifying if a patient with hepatitis lives or dies. These 155 instances are distributed as 32 classes with label *1*, indicating those patients have died, and 123 classes labelled as *2*, indicating the patient had survived.

2.2 Diabetic Retinopathy Dataset

The Diabetic Retinopathy (DR) dataset consists of 1151 instances with 20 attributes (including the class attribute) classifying if an image contains signs of diabetic retinopathy or not. These 1151 instances are distributed as 540 classes with label *0* indicating those patients have diabetic retinopathy and 611 classes labelled as *1* indicating they do not.

2.3 Preprocessing

The two datasets were first preprocessed by adding column names to the corresponding columns. Then, we removed the invalid records which were either null or had a considerable number of missing values. We dropped two attributes: *Protime* and *AlkPhosphate* from the Hepatitis dataset since they had a considerable number of empty rows consisting of 67 and 29 missing values. It is safe to drop these because they do not provide important information to our analysis and have no significant effect on the accuracy of our model. This decision allowed us to remove 18.7%, instead of 48.4% of the data from the dataset, leaving more samples for training and testing. After removing these columns we were left with a class distribution of 105 living and 24 deaths.

For the Diabetic retinopathy dataset, there were no rows dropped since there were no rows that contain missing values. To keep the class labels consistent across the two datasets, with *0* as healthy/survived and *1* as unhealthy/died, we changed the label *2*, which indicated survival to *0*. We also converted all columns in both datasets to numerical type for easier analysis later on. Lastly, we normalized the dataset for training/testing purposes.

2.4 Statistics

We performed statistical analyses on both datasets to examine the distributions of their labels and attributes.

As shown in Figure 1, the Hepatitis dataset is unbalanced, with only 18.6% of instances corresponding to the *Died* label. Among the 17 attributes, *Ascites* had the strongest correlation with *Class*. With further investigation with the heatmap, we found that *Ascites* are closely related to patients who survived hepatitis. Since the Hepatitis dataset originally contained both categorical and numerical attributes, we calculated summaries on continuous variables to analyze relationships between the features.

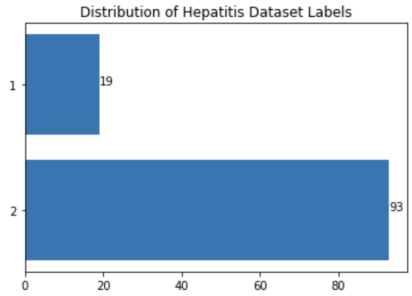

Figure 1: Distribution of *Died* and *Lived* labels

From Figure 2, we can see that there are 53% of patients with no signs of diabetes and 47% of patients with signs of diabetes in the Diabetic retinopathy dataset. The MA detection and exudates attributes, regardless of confidence levels, showed similar correlations with *Class*, whereas *Prescreen* showed the least association with *Class*. Overall, features in this dataset did not show a strong correlation with *Class*. Since all attributes are numerical, we performed numerical analyses on all columns.

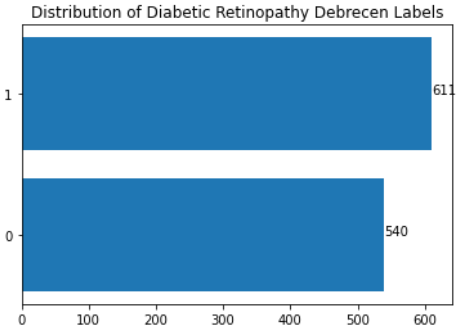

Figure 2: Distribution of *No Sign* and *Sign* labels

2.5 Ethical Concerns

We should be aware of the data distributions and acknowledge that they could lead to biased prediction results. If the instances collected are not representative enough for the patient population, the model could make inaccurate predictions on one's diagnosis. We should also take the platforms that we obtained the data from into consideration and make sure they are legal and respect the patient's privacy. Since the

UCI Machine Learning Repository is publicly accessible, it's safe to assume that this platform is compliant and legal.

3. Models

The implementations for both were heavily based on the examples given during the course's tutorial sessions (Rabbany, 2022).

3.1 K-Nearest Neighbor Model

The KNN algorithm is used for classification; the goal is to predict the correct class for the test data by calculating the distance between the test data and all the training points. Similar to the Nearest Neighbor algorithm, KNN expands by selecting the *k* number of points closest to the test data. This extension allows the KNN algorithm to calculate and select the highest probability of the test data belonging to the class of *k* training data. We used three distance functions to calculate the distance between two points: Euclidean, Manhattan, and Minkowski. The results will be discussed later in the paper.

3.2 Decision Tree Model

The Decision Tree model can be used for both classification and regression tasks; in this project, it was used for classification. The goal is to create a model that can predict the class of a target variable by learning decision rules derived from the training samples. The algorithm begins with the original training set as the root node, then iterates through each unused attribute to calculate its entropy and information gain (IG) and select the attribute that contributes the least entropy or the most IG. Subsets of data are produced by splitting the original dataset with the selected attribute. In this project, we specified the depth of the decision tree to avoid overfitting. We used three cost functions: misclassification, entropy, and Gini Index, to split the node in the Decision Tree. The accuracy of each function will be discussed later.

3.3 Evaluation metrics

For all the models, we used accuracy, precision, and recall as the evaluation metrics for measuring the performance of the model on all three splits of data (precision and recall will be discussed in Section 4.4). We calculated these metrics using a multi-classification confusion matrix; the exact computations are detailed in the corresponding codebase. These metrics provide a standardized benchmark on how the models perform on unseen data.

4. Results

A summary of the results recorded during this project can be found at the bottom of the paper in Table 1.

4.1 KNN Model

We tuned the hyperparameter for the KNN model using the validation set and realized that increasing the value of K rapidly increased the accuracy we were receiving, but past a certain threshold, the accuracy slowly declined and converged to a point. We can see this in Figure 1 when looking at the performance of the KNN model on the DR dataset using the Euclidean cost function. The best accuracy we received was stagnating at roughly 66% with a K value of 9 (Figure 3).

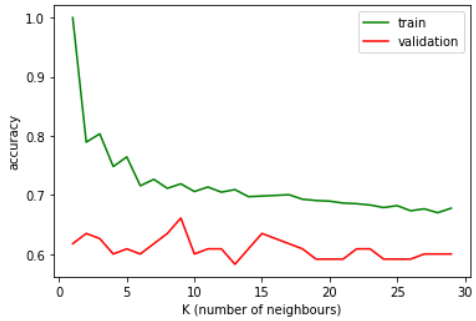

Figure 3: Accuracy vs. K value on DR dataset

We used three different distance functions to determine which would yield the best results: Euclidean, Manhattan, and Minkowski. On the training dataset, the Minkowski performed almost 10% higher. Other than that, there were no major differences between the three cost functions for either of the data.

4.2 Decision Tree Model

By setting a range of depths we wanted to explore, we found the optimal depth for the Decision Tree model using the validation set. For both datasets, the training accuracy increased as the depth increased. However, for the Hepatitis dataset, the testing accuracy is the highest when the Decision Tree only has two

layers. The testing accuracy for the Diabetic dataset shows an unstable trend as the depth increases, and eventually converges to a certain value. The observations are shown by the figures below after testing the models on the two datasets with the misclassification cost function. Figure 4 shows that the performance on the Hepatitis dataset peaks at 81.8% when the model's depth is 2. Figure 5 shows that the model has the highest accuracy at 62.6% with the DR dataset.

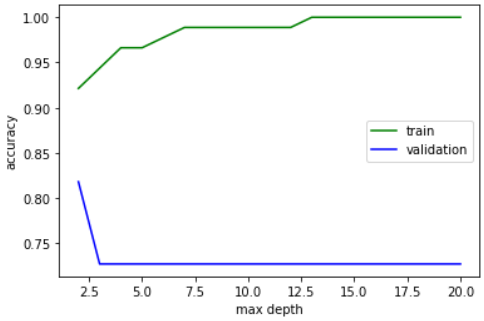

Figure 4: Accuracy vs. Depth on HP dataset

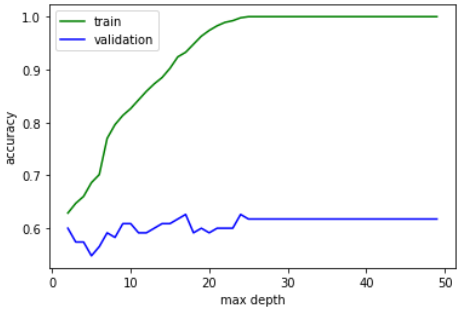

Figure 5: Accuracy vs. Depth on DR dataset

We used three different cost functions to see which one can produce the highest accuracy with the Decision Tree. We found that the Entropy function gave the best testing performance, with 83.3% for the Hepatitis dataset and 76.6% for the DR dataset.

4.3 Decision Boundary Plots

To determine which features to visualize using the decision boundary plot, we determined the features which had the highest correlation with the *Class* attribute of each dataset. These two most highly correlated features are *Spiders* and *Ascites* in the HP dataset, and *MA1* and *MA2* in the DR dataset. However, since *MA1* and *MA2* are the same results of MA detection at different confidence levels, the next highest correlated feature was selected, which is *EX7*. Figures 6 and 7 show the decision plots of *MA1* and *EX7* of both the KNN and the Decision Tree models.

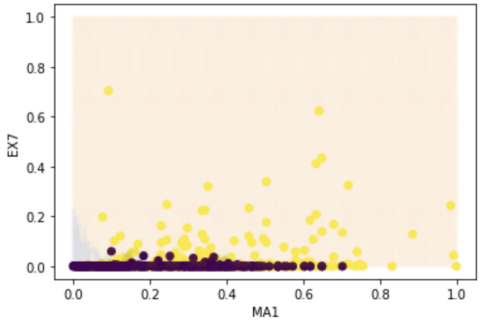

Figure 6: *MA1* vs. *EX7* plot using KNN model with Euclidean distance function

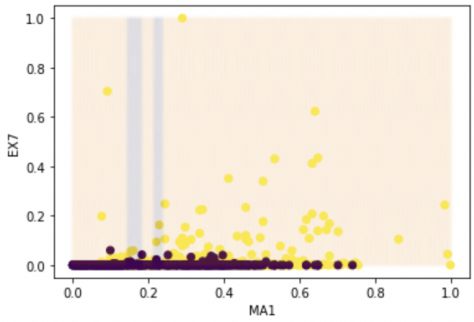

Figure 7: *MA1* vs. *EX7* plot using Decision Tree model with misclassification cost function

From the two figures, we can see that the decision boundaries plotted using the KNN model are more conservative, with the classification area of the *MA1* feature limited to near the origin. Conversely, the Decision Tree model divided the classification area through the Y dimension based on where the *MA1* data points are concentrated.

4.4 Creative Evaluations

Since KNN is sensitive to noisy features, we decided to remove some features from both datasets based on how these attributes are correlated with *Class*, hoping to increase the test accuracy for KNN. For the Hepatitis dataset, we selected the following features: *Ascites*, *Albumin*, *Varices*, and *Spiders*, based on the heatmap analysis we performed previously. We found that the testing accuracies are both 100% for the three different distance functions. For the DR dataset, since there is no particular feature that is closely correlated to *the Class* label based

on our heatmap analysis, we selected features based on their importance in diagnosing Diabetic Retinopathy; however, the accuracy decreased for all three distance functions. Our reasoning for this is that, since the DR dataset features are not closely correlated to *Class*, removing some features might provide the model with less information to classify labels.

Given that our project is in the domain of health data classification, false negatives have much more significant consequences and must be avoided when possible. We decided to measure the precision and recall in addition to the accuracy and observe if there are any significant changes in the optimal parameters and subsequent recording of these evaluation metrics.

From Table 2, we can see that the KNN model measured both a higher precision and recall value than the Decision Tree model. We can also observe that the Decision Tree model performed worst for the precision and recall on the DR dataset. Given that the recall value is higher for the KNN model, it is thus a better model when taking into account avoiding any false negatives.

5. Discussion and Conclusion

The KNN model performs better than the Decision Tree model on the Hepatitis dataset. We reason that this is due to the number of highly correlated attributes contained in the dataset, allowing the dataset to be separated into more distinct sections by the model.

On the other hand, the accuracies with the Decision Tree are higher than those with KNN for the DR dataset. Many values in the normalized columns of the DR dataset appeared to be 0, though we did not remove any record during preprocessing, we speculate that these might be invalid data. The Decision Tree is also less prone to the curse of dimensionality, given the number of features in the dataset. The KNN model is incapable of dealing with missing values, and might therefore perform poorly on the DR dataset. Since there are 19 attributes in the DR dataset and all of them do not have a strong correlation with *Class,* KNN might have difficulty finding the right weights and deciding which features are important for classification.

Possible extensions of this project could be to implement k-fold cross validation since the dataset, particularly the Hepatitis dataset, are small and unbalanced. We have tested with many different splits of the dataset. For example, the best result we received for KNN was when splitting the dataset with a ratio of 8:1:1 for train, validation, and test sets respectively, 6:2:2 for train, validation, test, which yielded roughly the same result and 9:0.5:0.5 for validation, test, and train sets, which yielded a better accuracy for validation on the DR dataset but realized it might overfit on the test set. We also noticed that no matter the ratio and where we split the dataset, this returned the same accuracy when testing on the validation set for the Hepatitis data with a result of 100%.

Since the Hepatitis dataset contains categorical and numerical attributes, another possible future investigation could be using separate distance functions on categorical and numerical variables to produce a more reliable accuracy result,

6. Statement of Contribution

We all contributed equally to the completion of the project and report.

7. References

Rabbany, Reihaneh. (2022). *COMP 551: Applied Machine Learning - Winter 2022*. CS551. (n.d.). Retrieved February 8, 2022, from http://www.reirab.com/Teaching/AML22/index.html

Saxena, R., Johri, A., Deep, V., & Sharma, P. (2018). “Heart Diseases Prediction System Using CHC-TSS Evolutionary, KNN, and Decision Tree Classification Algorithm” in *Advances in Intelligent Systems and Computing*.

Shouman, Mai & Turner, Timothy & Stocker, Rob. (2011). “Using decision trees for diagnosing heart disease patients.” in *Conferences in Research and Practice in Information Technology Series*. 121. 23-30.

W. Xing and Y. Bei, "Medical Health Big Data Classification Based on KNN Classification Algorithm," in *IEEE Access*, vol. 8, pp. 28808-28819, 2020, doi: 10.1109/ACCESS.2019.2955754.

Glen, S., & Glen, S. (2019, June 19). *Comparing classifiers: Decision trees, K-NN & Naive Bayes*. Data Science Central. Retrieved February 9, 2022, from https://www.datasciencecentral.com/comparing-classifiers-decision-trees-knn-naive-bayes/

Appendix

Model	Dataset	Best K (KNN) Max depth (DT)	Cost Function	Accuracy (%)		
				Train	Validation	Test
KNN	DR	9	Euclidean	71.9	66.1	67.8
	DR	15	Manhattan	69.9	67.0	71.3
	DR	2	Minkowski	78.8	62.6	58.3
	HP	14	Euclidean	93.2	100	91.7
	HP	14	Manhattan	92.2	100	91.7
	HP	14	Minkowski	93.2	100	91.7
Decision Tree	DR	17	Misclassification	92.1	62.2	75.0
	DR	2	Entropy	88.8	69.6	83.3
	DR	8	Gini Index	92.1	76.5	75.0
	HP	2	Misclassification	62.9	81.8	62.6
	HP	2	Entropy	64.8	81.8	76.5
	HP	2	Gini Index	64.8	100	69.6

Table 1: Accuracies of KNN and Decision Tree models with various cost functions.

Model	Dataset	Best K (KNN) Max depth (DT)	Cost Function	Precision (%)		
				Train	Validation	Test
KNN	DR	11	Euclidean	70.7	67.4	68.2
	HP	2		73.3	100	70.0
Decision Tree	DR	24	Misclassification	99.4	64.3	65.0
	HP	2		86.6	75.0	45.0
				Recall (%)		
KNN	DR	2	Euclidean	84.6	68.6	71.6
	HP	2		95.1	100	70.0
Decision Tree	DR	2	Misclassification	64.2	60.8	67.3
	HP	2		83.2	95.0	40.9

Table 2: Precisions and recalls of KNN and Decision Tree models with associated cost function.