Assessing the Effectiveness of the K-Nearest Neighbors Algorithm in Predicting Breast Cancer.

To what extent can the K-Nearest Neighbors algorithm be used to predict breast cancer?

Computer Science

Word count: 3998

Contents:

Introduction

Throughout history, healthcare has evolved drastically and has allowed people to live longer and healthier lives. Via technological advancements, high definition equipment and analysis provides the necessary early detection and preventative intervention so as to improve the survival rate and tackle the relevant health implications. Technology applications surround healthcare and have been one of the biggest factors for the increase in life expectancy. In 1945, the average life expectancy in the United States was 64 years, while in 2020 it increased to 78 years.[1] This drastic change has been a result of many innovations and new approaches.

In healthcare the most common type of cancer is breast cancer, which affects 2.3 million people worldwide. Early detection of diseases such as breast cancer have resulted in an increase in life expectancy. Specifically, breast cancer is most common amongst females, but in some exceptional cases can also be found in males. This cancer affects multiple people's lives and can have fatal impacts especially if not detected in the early stages to be treated.[2]

Artificial Intelligence, is a field of science that has human intelligence demonstrated by machines that can reason, learn and act as a human would. It has revolutionized the world and

[1] "Life expectancy in the United States from 1900 to 2021." Statista, 2021, https://www.statista.com/statistics/1040079/life-expectancy-united-states-all-time/. Accessed 24 Aug 2024
[2] Breast Cancer. https://www.who.int/news-room/fact-sheets/detail/breast-cancer. Accessed 1 Sep. 2024.

especially healthcare. By analyzing vast amounts of data and training machine learning algorithms, a model can be constructed to predict whether a patient has a certain disease. Consequently, machine learning models have been widely used to predict different types of cancers with one of the most common and important applications being in breast cancer. One of the most effective types of algorithms that are used to predict breast cancer is the K-nearest neighbors algorithm.[3]

This paper will explore the process of extracting data from patients, the abilities of the K-nearest neighbors algorithm, and compare the K-nearest neighbors algorithm to other algorithms such as K-means clustering algorithm, logistic regression and random forest. Also, this paper will delve into the limitations of the K-nearest neighbors algorithm (KNN) alongside the different hyperparameters that can be enforced on the algorithm to improve accuracy. There will also be a comparison between different machine learning models and their accuracy in predicting whether a biopsy sample is benign or malignant.

Breast Cancer & Breast Cancer Detection Methods

Breast Cancer is a type of cancer that develops in the breast tissue and can be present in one or two breasts. Once it is present it spreads to the rest of the body parts and can be fatal if not treated. As it is a very common type of cancer, tests to detect it are performed regularly to determine if the cancer is benign, meaning that it is stable and does not spread to other parts of

[3] Sarkar, M., and T. Y. Leong. 'Application of K-Nearest Neighbors Algorithm on Breast Cancer Diagnosis Problem.' Proceedings of the AMIA Symposium, 2000, pp. 759–63. PubMed Central, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2243774/. Accessed 29 Aug. 2024.

the body, or malignant, that it is not stable and spreads to other parts of the body which can become fatal.[4]

More specifically, in this article we will be showing an example of how Artificial Intelligence and machine learning models are used in breast cancer analysis. One of the processes in a breast cancer exam is to screen a breast in the form of a mammogram or x-ray. In a mammogram, a radiologist must detect abnormalities and changes in the breast, such as small white spots called calcifications, masses, or any other suspicious areas that might be cancerous. You may see an example of 4 different mammograms labeled as breast cancer or not in Figure 1[5].

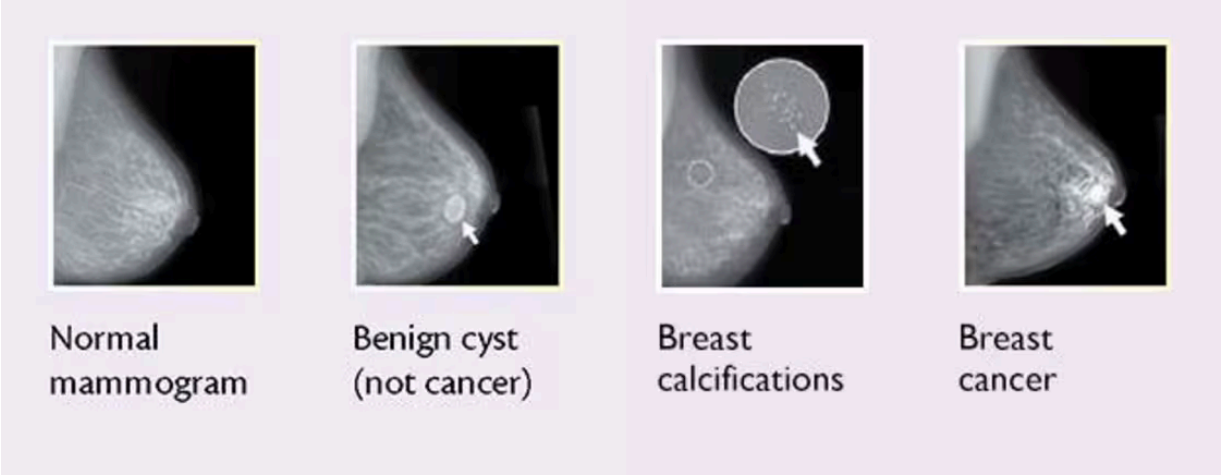


Figure 1: Four different mammograms each labeled as having or not having breast cancer.

Radiologists can sometimes find it extremely difficult to determine whether a patient has breast cancer, as they must visually analyze medical images, a process that is inherently prone to

[4] What Is Breast Cancer? https://www.cancer.org/cancer/types/breast-cancer/about/what-is-breast-cancer.html. Accessed 29 Aug. 2024.
[5]"Breast Cancer: Pictures, Lumps, Signs." Today, NBCUniversal Media, 6 Oct. 2021, https://www.today.com/health/breast-cancer/pictures-of-breast-cancer-lumps-signs-rcna5360. Accessed 30 Aug. 2024.

human error. Factors such as fatigue, cognitive bias, and variability in interpretation can make pattern recognition in images particularly challenging, potentially leading to misdiagnosis. Hence, the creation of an Artificial Intelligence model is needed in order to help the radiologists and increase the accuracy of cancer detection. The first ever AI model to outperform radiologists was created by DeepMind. It was trained on thousands of mammograms and proved an astonishing accuracy compared to radiologists. In fact, when the model was put to test by 25,000 women in the United Kingdom and 3,000 women in the United States, the system reduced the percentages of cases where the radiologists missed breast cancer by 9.4% in the United States and 2.7% in the United Kingdom. In addition to that, it managed to reduce incorrect positive readings of breast cancer by 5.7% in the US and 1.2% in the United Kingdom[6].

However, a mammogram may not always yield accurate results that indicate the existence or non-existence of a cancer. Therefore, when doctors require a more accurate but time-consuming and costly examination, they collect biopsy samples from patients.[7] To effectively analyze the KNN algorithm, we will focus on biopsy samples, as they provide the most accurate real-life data for determining whether a patient has breast cancer.

Machine Learning Techniques

In AI, an algorithm is a set of mathematical rules or procedures used to process data and learn patterns. A model, on the other hand, is the trained version of an algorithm whose

[6] "Google AI Beats Doctors at Breast Cancer Detection—Sometimes." The Wall Street Journal, 30 Jan. 2020, https://www.wsj.com/articles/google-ai-beats-doctors-at-breast-cancer-detectionsometimes-11577901600. Accessed 30 Aug. 2024.
[7] Radiology (ACR), Radiological Society of North America (RSNA) and American College of. ‘Stereotactic Breast Biopsy’. Radiologyinfo.Org, https://www.radiologyinfo.org/en/info/breastbixr. Accessed 1 Sep. 2024.

parameters and weights are adjusted based on the data it has learned from. When training a model there are three different learning methods: supervised, unsupervised and reinforced. Supervised learning requires data that is labeled and consists of a set of features. Unsupervised learning does not require labeled data and instead attempts to derive patterns from data[8]. In reinforcement learning, a model interacts with an environment, learning through a trial-and-error process by receiving feedback directly on its results. Reinforcement learning will not be used as it is designed for multi-step decision-making, like playing a game or driving a car, whereas breast cancer prediction is a one-step classification task. The KNN algorithm is a supervised machine learning model. Among the other models used, K-means clustering is an unsupervised learning algorithm, while random forest and logistic regression are both supervised learning models

Background Research:

I. Choice for Algorithms to Compare

The algorithms selected for comparison with KNN were chosen for a specific reason. All are among the most commonly used in healthcare for predictive tasks. K-means clustering was included as an unsupervised learning model, providing a contrast to KNN and allowing an evaluation of whether a different learning approach is more effective for breast cancer detection. Random forest, a decision tree-based algorithm, was selected for its ability to handle noisy data, making it a valuable comparison to KNN. Since both are supervised learning models, this helps

[8] 'K Means Clustering - Introduction'. GeeksforGeeks, 2 May 2017, https://www.geeksforgeeks.org/k-means-clustering-introduction/. Accessed 3 Sept. 2024.

assess KNN's effectiveness to handle noise in breast cancer data. Lastly, logistic regression was chosen as a strong comparison to KNN allowing an evaluation of whether a binary classification model can outperform KNN's multi-class classification in detecting breast cancer.

II. K-Nearest Neighbors Algorithm

The KNN algorithm is a supervised machine learning algorithm with the goal of classing data points based on the bulk class of their "k-nearest neighbors". The k-nearest neighbors are a number of data points denoted by k, that are the closest distance to an unlabeled data point. To calculate the distance between points, multiple metrics are used in the KNN such as the Euclidean Distance, Hamming Distance, Manhattan Distance and many more. To ensure a fair comparison between the models being analyzed, Euclidean distance will be used in the KNN algorithm, as it is also employed in the K-means clustering algorithm.

$$(X_i, X_j) = \sqrt{\sum_{k=1}^{d} (x_{i,k} - x_{j,k})^2}$$

Figure 2: Mathematical equation of the KNN algorithm.

The euclidean distance is calculated using the equation in Figure 2. In the equation, d represents the number of features in the data set. $\sum_{k=1}^{d}$ sums the distances between the data points from $k = 1$ until d. While $x_{i,k}$ is a data point i of a feature k and $x_{i,k}$ is a data point i of a feature k.[9]

III. K-Means Clustering Algorithm

The first algorithm that will be used for comparison with KNN is the well-known and widely used K-means clustering algorithm. Unlike the KNN algorithm, it is an unsupervised machine learning algorithm.

K-means is an algorithm that works by partitioning n number of observations, into k different clusters, a part of the data that share similar characteristics. The aim is to partition the data so that each observation belongs to the cluster with the nearest mean. Hence, the model finds the cluster centroids, the mean of the data points belonging to the specific cluster, that represent the center of the data. The mathematical equation for the K-means clustering algorithm can be seen in Figure 3.

[9] 'K-Nearest Neighbors(KNN)'. AlmaBetter, https://www.almabetter.com/bytes/tutorials/data-science/k-nearest-neighbors. Accessed 2 Sept. 2024.

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i - \mu_j \right\|^2$$

Figure 3: Mathematical equation of the K-means clustering algorithm.

In order to better evaluate the K-means clustering algorithm it is important to get an understanding of the mathematical formula. The equation variables are as follows: i represents a data point i, and j represents a cluster j. $\sum_{j=1}^{k}$ is the sum of each cluster j, while $\sum_{i=1}^{n}$ is the sum of all data points i to n that are part of cluster j. The x_i variable is the data set point which is a dimensional vector and is from cluster j. Lastly, μ_j is the centroid of cluster j.[10]

IV. Random Forest Algorithm

Another algorithm that will be used for comparison with KNN is the random forest algorithm, a supervised machine learning algorithm that utilizes decision tree models through an ensemble method. The random forest predicts an outcome by partitioning the data into predictors.

[10] Weisstein, Eric W. K-Means Clustering Algorithm. https://mathworld.wolfram.com/. Accessed 3 Sept. 2024.

There is not one single equation for the random forest model but there is a set of steps that have different equations in them. In general, decision trees form predictions by calculating which class is the most common amongst the training sets of observations within the partitions they have created. There are essentially nodes, decisions to be made, that are calculated by various metrics. Two very common metrics that are used to choose partitions are the Gini index and Entropy.[11] The Gini index calculates the probability of a data point chosen at random to be misplaced in a class. Entropy is used to measure the randomness and unpredictability of the data. The Gini index metric will be used in this random forest model and the equation can be seen in Figure 4.

$$j = \sum_{k=1}^{n} (1 - \hat{y}_{j,k})$$

Figure 4: Mathematical equation of the Gini Index, part of the random forest algorithm.

In the Gini index, j is a node in the decision tree and $\sum_{k=1}^{n}$ sums over n, the number of classes of the decision tree, from $k = 1$ to n. While $\hat{y}_{j,k}$ is the proportion of data points of node n that are in the class k.[12] For simplicity, we will assume that the Gini index is the sole criterion used by the random forest model to determine where to split nodes and make decisions.

[11] What Is Random Forest? | IBM. 20 Oct. 2021, https://www.ibm.com/topics/random-forest. Accessed 4 Sept. 2024.
[12] Benco, Stefano. 'Analyzing the Decision Tree Algorithm'. Math for Business, the Ultimate Applied Math Blog, 4 July 2019, https://mathforbusiness.com/data-science/analyzing-the-decision-tree-algorithm/. Accessed 4 Sept. 2024.

V. Logistic Regression Algorithm

Finally, the logistic regression algorithm will be compared to KNN. Logistic regression is also a supervised machine learning algorithm used for binary classification, a classification where a given data point is predicted to belong to one of two defined classes, or data factors. In our case the two classes are benign or malignant. The model is based on probability as it uses the logistic function, most commonly known as the sigmoid function, to map out the predictions and their probabilities ranging from 0, benign, to 1, malignant.[13] The equation of the logistic regression algorithm can be seen in Figure 5.

$$P = \frac{e^{a+bX}}{1 + e^{a+bX}}$$

Figure 5: Mathematical equation of the logistic regression algorithm.

The probability of a data point being benign or malignant is defined by P. In the fraction euler's number is raised to the power of $a + bX$, where a represents the intercept term, which is a value that is used in many machine learning algorithms and allows the model to make a prediction even if all the predictors are an invalid number such as 0. X represents a predictor and

[13] 'Logistic Regression in Machine Learning'. GeeksforGeeks, 9 May 2017, https://www.geeksforgeeks.org/understanding-logistic-regression/. Accessed 6 Sept. 2024.

b represents the coefficient or weight of that predictor which is calculated when fitting the model with data from the train set.[14]

Methodology:

I. Data Set

To analyze the effectiveness of the KNN algorithm the BRCA dataset which comes from the *dslabs* package will be used. The programming language used to analyze this data set is one that I am very familiar with, R. R is mostly known for its easy and fast use in statistical computing[15]. The BRCA dataset was selected because, when seeking more definitive and certain results, biopsies are preferred over mammograms. Additionally, the BRCA dataset is well-known and widely used for machine learning case studies in healthcare. Its public availability and compliance with ethical guidelines also make it a suitable choice, as it contains sensitive medical data.

The dataset contains breast cancer biopsy samples of benign and malignant tumors. It has 30 different predictors, which are features of the biopsies selected and 569 biopsy samples. The BRCA dataset is a list consisting of a vector of samples y, and predictors x. Vector x, contains features describing properties of the size and shape of cell nuclei extracted from a biopsy microscope image. While the vector y contains sample classifications, with B (Benign) and M

[14] Logistic Regression. http://faculty.cas.usf.edu/mbrannick/regression/Logistic.html. Accessed 8 Sept. 2024.
[15] R: BRCA Dataset. https://search.r-project.org/CRAN/refmans/Przewodnik/html/brca.html. Accessed 1 Sept. 2024.

(Malignant). A part of the data set with vector x and vector y can be seen in Figure 6 and Figure 7 respectively.

```
$x
      radius_mean texture_mean perimeter_mean area_mean smoothness_mean
 [1,]       13.54        14.36           87.5       566          0.0978
 [2,]       13.08        15.71           85.6       520          0.1075
 [3,]        9.50        12.44           60.3       274          0.1024
 [4,]       13.03        18.42           82.6       524          0.0898
 [5,]        8.20        16.84           51.7       202          0.0860
```

Figure 6: Part of vector x of BRCA dataset.

```
$y
  [1] B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B
 [40] B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B
 [79] B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B
[118] B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B
[157] B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B
[196] B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B
[235] B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B
[274] B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B
[313] B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B
[352] B B B B B M M M M M M M M M M M M M M M M M M M M M M M M M M M M M
[391] M M M M M M M M M M M M M M M M M M M M M M M M M M M M M M M M M M
```

Figure 7: Part of vector y of BRCA dataset.

The data set is split into two sets, the train set and the test set. The train is used to train, tune and choose the parameters of the algorithms. The test set is used to test the trained models, compare them with other models and determine the best performing one with the highest accuracy.

II. K-Nearest Neighbors Algorithm

To build the KNN model the number of nearest neighbors, k, needs to be selected and many times the optimal value for k is unknown. As a result, to find the optimal value for k a sequence of different values of k needs to be conducted. The number of neighbors is important as a large number can result in overfitting the model, while a small number can result in underfitting the model. The code in Figure 8 generates a sequence of odd numbers from 3 to 21 to train the KNN model, which is then tested on the test set.

```
# Set the seed for reproducibility
set.seed(7, sample.kind = "Rounding")

# Define tuning grid
tuning <- data.frame(k = seq(3, 21, 2))

# Train the KNN model with cross-validation
train_knn <- train(train_x, train_y,
                   method = "knn",
                   tuneGrid = tuning,
                   trControl = trainControl(method = "cv", number = 10))  # 10-fold CV

# Make predictions on the test set
knn_preds <- predict(train_knn, test_x)

# Get accuracy results for all values of k
accuracy_results <- train_knn$results
knn_accuracy <-max(accuracy_results$Accuracy)

# Find highest accuracy k
best_k <- accuracy_results$k[which.max(accuracy_results$Accuracy)]
best_accuracy <- max(accuracy_results$Accuracy)
```

Figure 8: Code to train and test the KNN model using different k-values.

It is important to get a better understanding of the accuracies of different neighbours by plotting a graph of the Accuracy Vs. Number of Neighbors.

```r
ylim_range <- c(min(accuracy_results$Accuracy) - 0.05, max(accuracy_results$Accuracy) + 0.05)

# Plot the results with custom styling
plot(accuracy_results$k, accuracy_results$Accuracy, type = "o", pch = 16, lwd = 2, col = "blue",
     xlab = "Number of Neighbors (k)", ylab = "Accuracy",
     ylim = ylim_range,
     xaxt = 'n',
     bty = "l")   #
axis(1, at = accuracy_results$k, labels = accuracy_results$k, cex.axis = 1.1)
grid(nx = NA, ny = NULL, col = "gray", lty = "dotted")
points(accuracy_results$k, accuracy_results$Accuracy, pch = 19, col = "darkgreen", cex = 1.5)
lines(accuracy_results$k, accuracy_results$Accuracy, col = "darkorange", lty = 1, lwd = 2)
# Red point & dashed line for best k
points(best_k, best_accuracy, col = "red", pch = 19, cex = 2)
abline(v = best_k, col = "red", lty = "dashed", lwd = 2)
text(best_k, best_accuracy, labels = paste("Best k =", best_k, "\nAccuracy =", round(best_accuracy * 100, 2), "%"),
     pos = 3, col = "red", cex = 1.2)
title(main = "KNN Accuracy vs Number of Neighbors (k)", col.main = "darkblue", font.main = 2)
```

Figure 9: Code to create graph of KNN Accuracy Vs. Number of Neighbors (k).

It can be seen in Figure 10 that the optimal number of neighbors is 5.

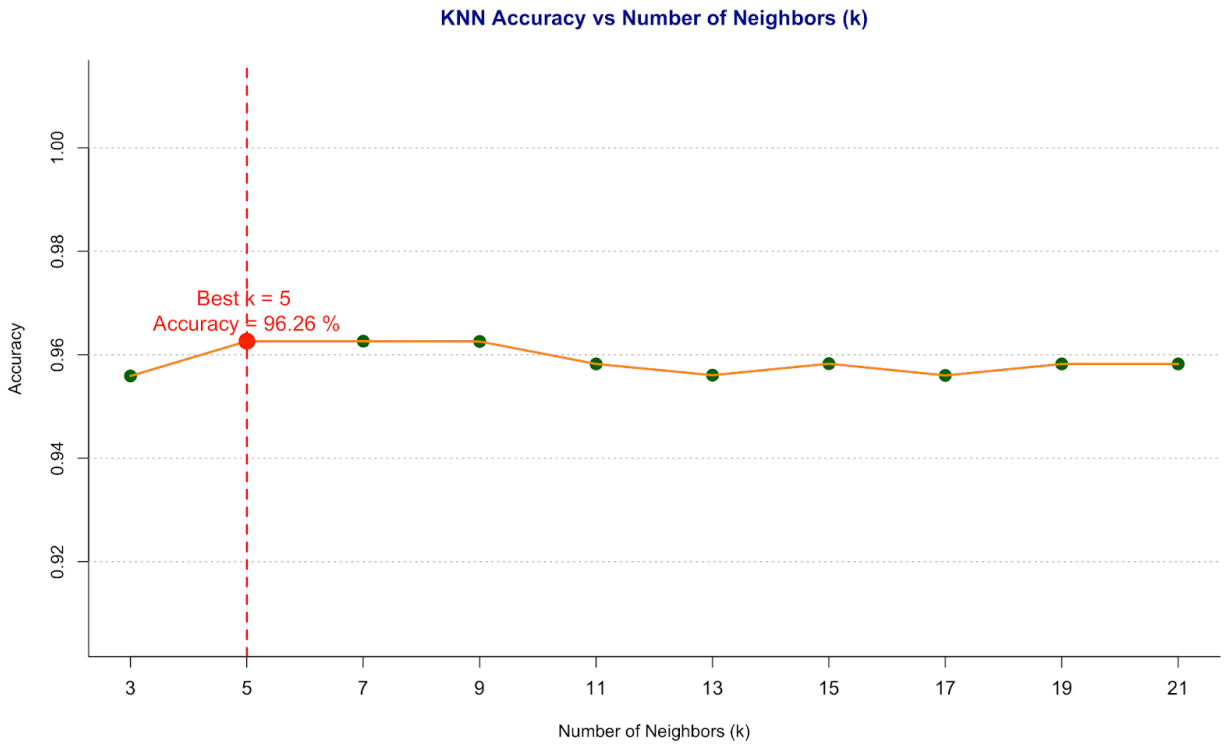


Figure 10: Graph of KNN Accuracy vs Number of Neighbors (k).

The KNN model is a supervised machine learning algorithm and needs clear data without a lot of noise, as its predictions are based on previous data that it has seen. Also, the KNN model does not need data to be directly defined in clusters but needs some sort of clusters in the data so that it can perform with high accuracy. For simplicity purposes the weights that the KNN algorithm assigns to different data points in the neighborhood will not be explored in this paper.

III. K-Means Clustering Algorithm

When building the K-means clustering algorithm, it is crucial to optimize for the best centroid and that is why I conducted a sequence with different centroids to see which one yields the greatest accuracy. Underfitting will deem the algorithm to be inaccurate because it oversimplifies the data by grouping it into less clusters than it actually belongs to. On the other hand, overfitting results in a decrease in accuracy because the algorithm uses too many clusters and picks up on what is called data noise, which are natural anomalies found in all data. This will result in the algorithm becoming biased to the train set and not perform well on the test set. The code in Figure 11 generates a sequence of 1 to 5 centroids to train the algorithm, and tests it on the test set.

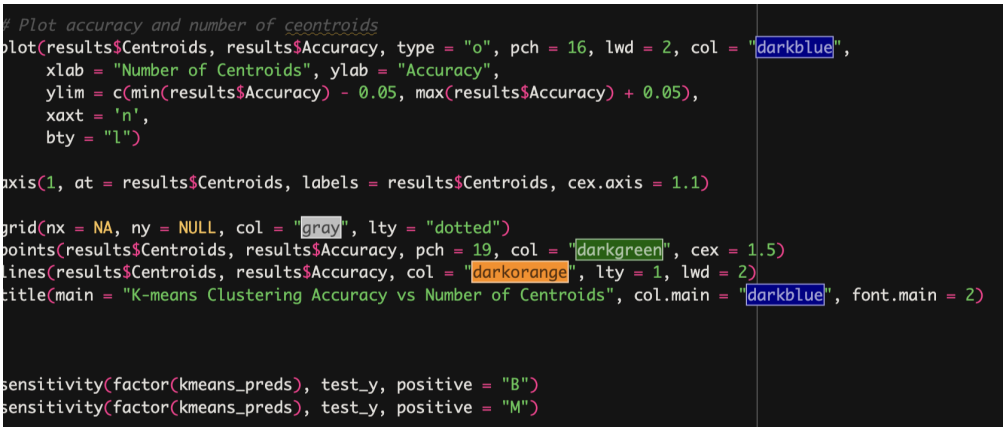
```r
# Create predict_kmeans function
predict_kmeans <- function(x, k) {
  centers <- k$centers

  # calculate distance to cluster centers with manual Euclidean distance
  distances <- sapply(1:nrow(x), function(i) {
    apply(centers, 1, function(y) sqrt(sum((x[i, ] - y)^2)))
  })

  max.col(-t(distances))  # select cluster with min distance to center
}
# Scale the data
train_x_scaled <- scale(train_x)
test_x_scaled <- scale(test_x)
results <- data.frame(Centroids = integer(), Accuracy = numeric())
# Loop 1 to 5 centroids
for (centroids in 1:5) {
  set.seed(3, sample.kind = "Rounding")    # Ensure reproducibility
  k <- kmeans(train_x_scaled, centers = centroids, iter.max = 100)
  kmeans_preds <- predict_kmeans(test_x_scaled, k)
  if (centroids == 2) {
    kmeans_preds <- rep("B", length(kmeans_preds))
  } else if (centroids == 2) {
    kmeans_preds <- ifelse(kmeans_preds == 2, "B", "M")
  } else {
    kmeans_preds <- ifelse(kmeans_preds == 2, "B", "M")
  }
  # Calculate the accuracy
  accuracy <- mean(kmeans_preds == test_y)
  results <- rbind(results, data.frame(Centroids = centroids, Accuracy = accuracy))
}
print(results)
```

Figure 11: Code to train and test K-means clustering algorithm.

Graphing the number of centroids and their corresponding accuracy provides a better understanding of the clusters.



```
# Plot accuracy and number of ceontroids
plot(results$Centroids, results$Accuracy, type = "o", pch = 16, lwd = 2, col = "darkblue",
     xlab = "Number of Centroids", ylab = "Accuracy",
     ylim = c(min(results$Accuracy) - 0.05, max(results$Accuracy) + 0.05),
     xaxt = 'n',
     bty = "l")

axis(1, at = results$Centroids, labels = results$Centroids, cex.axis = 1.1)

grid(nx = NA, ny = NULL, col = "gray", lty = "dotted")
points(results$Centroids, results$Accuracy, pch = 19, col = "darkgreen", cex = 1.5)
lines(results$Centroids, results$Accuracy, col = "darkorange", lty = 1, lwd = 2)
title(main = "K-means Clustering Accuracy vs Number of Centroids", col.main = "darkblue", font.main = 2)



sensitivity(factor(kmeans_preds), test_y, positive = "B")
sensitivity(factor(kmeans_preds), test_y, positive = "M")
```

Figure 12: Code to graph K-means clustering Accuracy Vs. Number of Centroids.

Figure 13 shows that the optimal number of clusters is 2 based on the resulting accuracies.

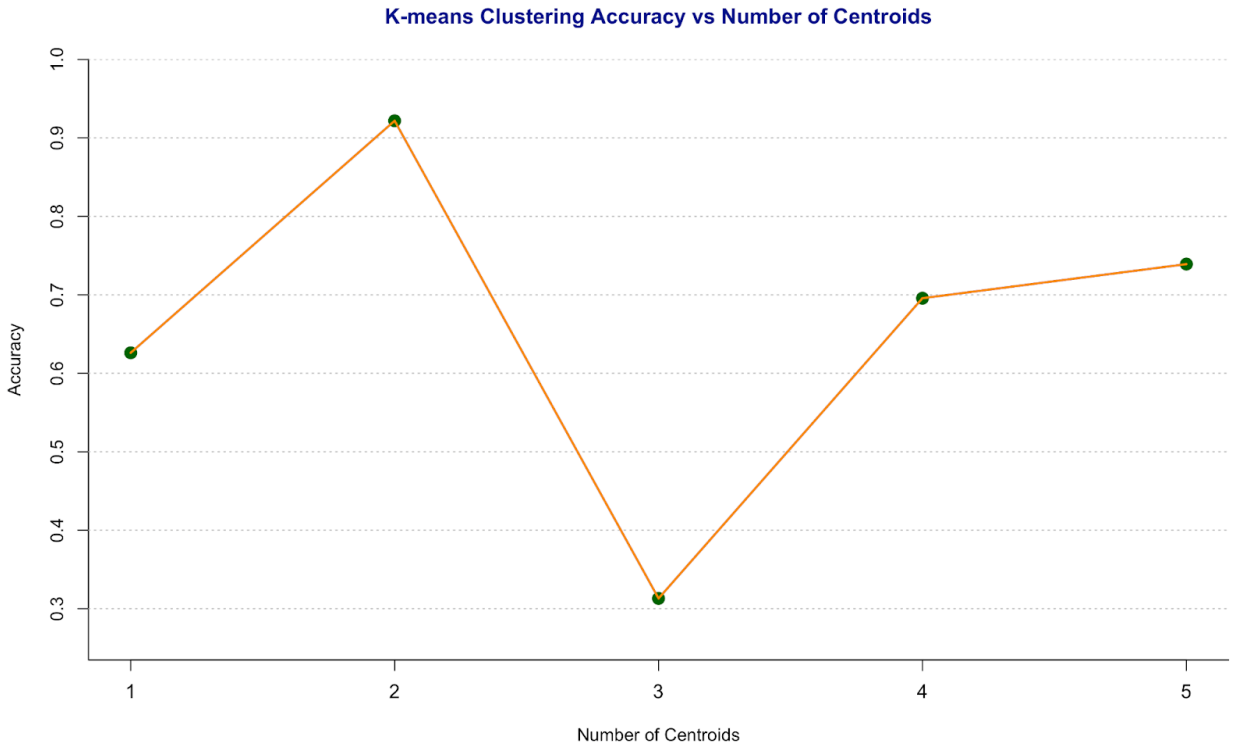


Figure 13: Graph of K-means clustering Accuracy vs. Number of Centroids.

IV. Random Forest Algorithm

When building the random forest algorithm, an important hyperparameter to optimize is the amount of input features, denoted by the name mtry in R programing. This hyperparameter determines which features result in the splitting of a node. For example, a mtry of 3 says that in each split the algorithm considers 3 features.[16] The features are then calculated by the Gini index, and the one with the lowest value in the Gini index is chosen since the lower the Gini index value, the more important that feature is.

[16] Ellis, Christina. 'Mtry in Random Forests'. Crunching the Data, 28 Aug. 2022, https://crunchingthedata.com/mtry-in-random-forests/. Accessed 4 Sept. 2024.

```r
set.seed(9, sample.kind = "Rounding") # simulate R 3.5
tuning <- data.frame(mtry = c(1, 2, 3, 4, 5, 6, 7, 8, 9))
train_rf <- train(train_x, train_y,
                  method = "rf",
                  tuneGrid = tuning,
                  importance = TRUE)
rf_preds <- predict(train_rf, test_x)
# Best mtry
train_rf$bestTune
# Accuracy of best
mean(rf_preds == test_y)
# Important variables
varImp(train_rf)
fit <- rpart(train_y ~ ., data = data.frame(train_x, train_y))
custom_palette <- c("#66C2A5", "#FC8D62")
rpart.plot(fit, type = 3, extra = 101, under = TRUE, faclen = 0, tweak = 1.2,
           box.palette = custom_palette,  # Apply custom colors
           shadow.col = "gray", fallen.leaves = TRUE)
```

Figure 14: Code to train, test and visualize the random forest algorithm.

Figure 15 illustrates the decision tree structure and its resulting predictions, with percentages indicating the proportion of data classified into each category.

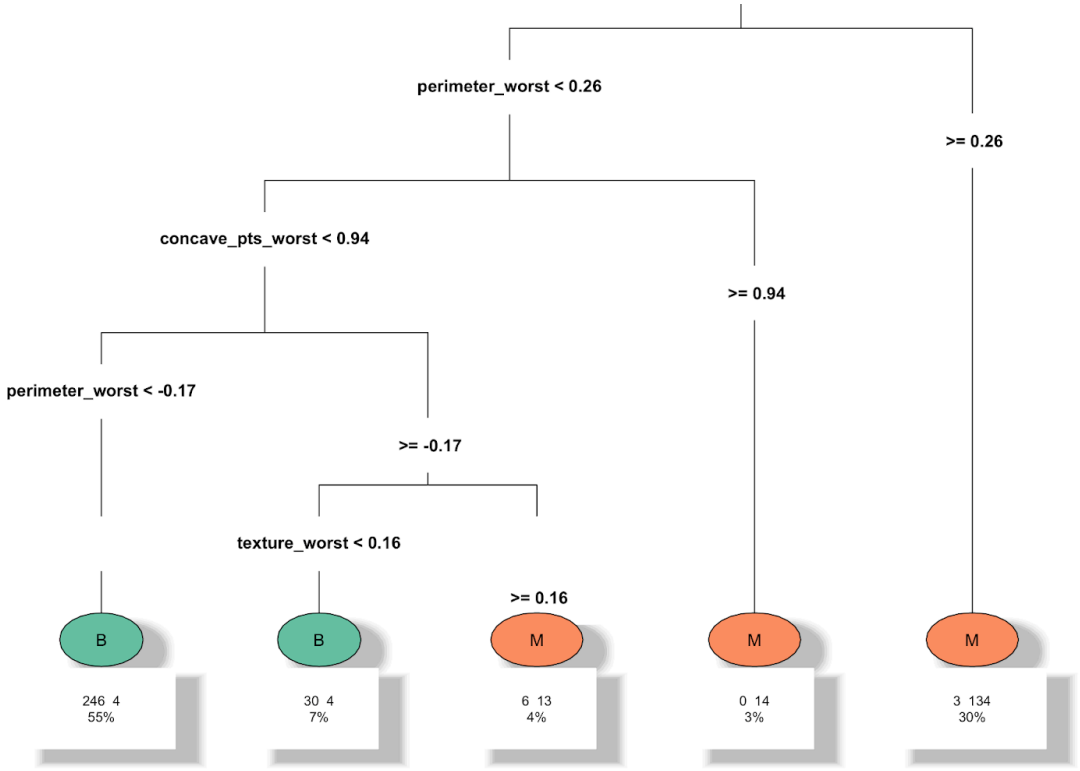


Figure 15: Illustration of some of the tree decisions of the random forest algorithm.

It is very important to optimize for the mtyr value to find which value yields the best accuracy and uses that in the random forest model.

```
# Plot
ggplot(rf_results, aes(x = mtry, y = Accuracy)) +
  geom_point(size = 3, color = "darkorange") +
  geom_line(color = "darkgreen", size = 1.5) +
  geom_point(aes(x = best_mtry, y = best_accuracy), color = "red", size = 4) +
  geom_vline(xintercept = best_mtry, linetype = "dashed", color = "red", size = 1) +
  annotate("text", x = best_mtry, y = best_accuracy + 0.002,
           label = paste("Best mtry =", best_mtry, "\nAccuracy =", round(best_accuracy * 100, 2), "%"),
           color = "red", size = 5, vjust = -0.5) +
  labs(title = "Random Forest Accuracy vs. mtry",
       x = "mtry (Number of Randomly Selected Predictors)",
       y = "Accuracy") +
  ylim(y_min, y_max) +
  theme_minimal(base_size = 15) +
  theme(plot.title = element_text(hjust = 0.5, color = "darkblue", size = 18, face = "bold"))
```

Figure 16: Code to create graph of Random Forest algorithm accuracy Vs. mtyr values.

Figure 17 shows that an mtry value of 4 achieved the highest accuracy. When mtry values are below 4, the model under-trains by using too few features, limiting its ability to account for all relevant factors before splitting a node. Conversely, higher mtry values result in overtraining, where too many features are considered, leading to reduced accuracy.

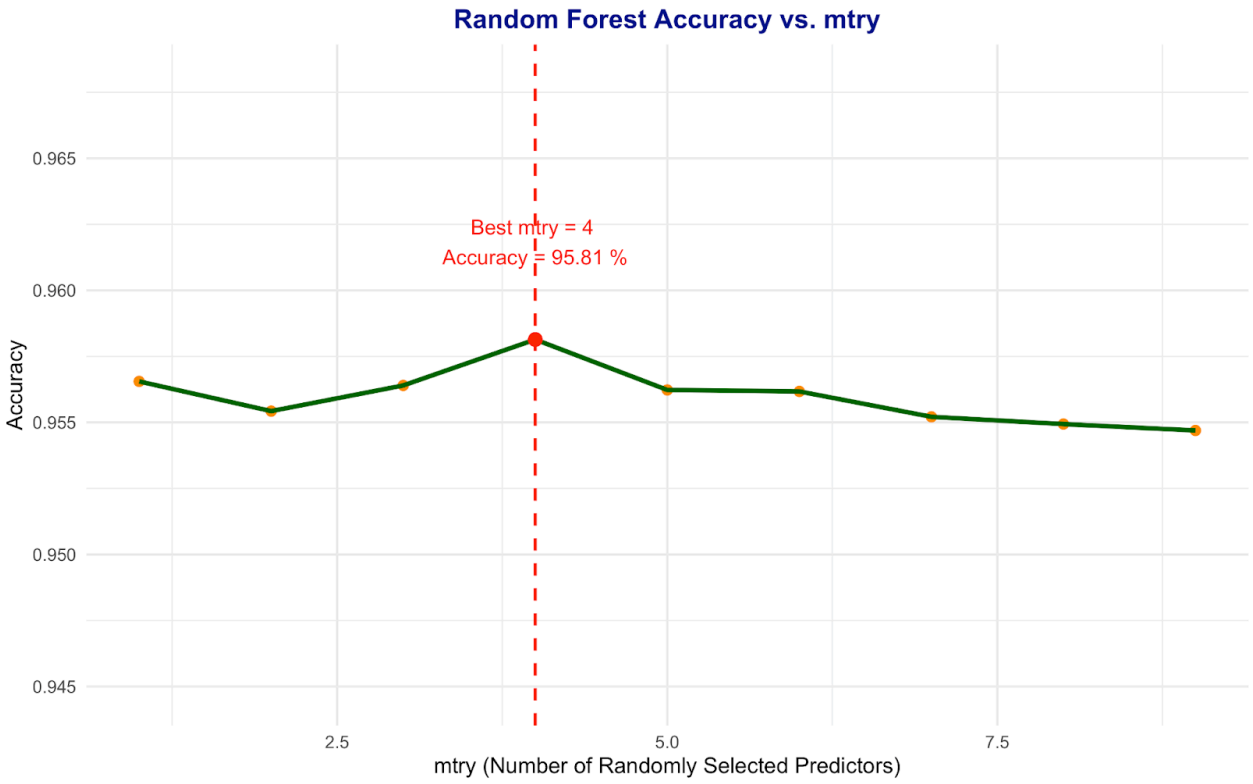


Figure 17: Graph of Random Forest Accuracy vs. mtry.

V. Logistic Regression Algorithm

The process for building the logistic regression model is slightly different. Unlike the rest of the machine learning models used, it does not have multiple hyperparameters to tune or

change in order to increase the accuracy. However, a method that can help increase the accuracy of the model is the use of regularization which is very commonly used in the logistic regression and linear regression machine learning models.

Regularization is a method applied to machine learning models in order to avoid overfitting, by penalizing large weights in the model. Overfitting is very common amongst supervised machine learning algorithms and especially in logistic regression which is designed for binary classification. For simplicity purposes the mathematical equations behind regularization will not be talked about in depth. There are two types of regularization L1, most commonly known as lasso regression and L2, most commonly known as ridge regression. L1 regularization adds the absolute value of the weight in the loss function. L2 regularization adds the square value of the weight to the loss function.[17] A loss function quantifies the difference between an algorithm's predictions and the actual outcomes, providing a measurable way to evaluate performance.

```
set.seed(1, sample.kind = "Rounding")
train_glm <- train(train_x, train_y, method = "glm")
glm_preds <- predict(train_glm, test_x)


# Accuracy
mean(glm_preds == test_y)

# Use Lasso and Ridge regularization
train_glmnet <- train(train_x_scaled, train_y,
                      method = "glmnet",
                      tuneGrid = expand.grid(alpha = c(0, 1),  # Lasso (alpha = 1), Ridge (alpha = 0)
                                             lambda = seq(0.001, 0.1, length = 10)))  # Lambda values

# Extract  results
results <- train_glmnet$results

# Sequence of lambda values
lambda_values <- seq(0.0001, 1, length = 100)
```

Figure 18: Code to train two logistic regression models with L1 and L2 regularization.

[17] 'L1 and L2 Regularization Methods, Explained'. Built In, https://builtin.com/data-science/l2-regularization. Accessed 8 Sept. 2024.

It is important to plot and compare the two regularization methods.

```r
y_min <- min(results$Accuracy) - 0.02
y_max <- max(results$Accuracy) + 0.05

# Graph
ggplot(results, aes(x = lambda, y = Accuracy, color = factor(alpha))) +
  geom_point(size = 3) +
  geom_line(size = 1.5) +

  labs(title = "Accuracy vs Lambda for Lasso and Ridge Regularization",
       x = "Lambda (Regularization Strength)",
       y = "Accuracy",
       color = "Regularization Type") +
  scale_color_manual(values = c("darkblue", "darkorange"), labels = c("Ridge", "Lasso")) +
  ylim(y_min, y_max) +
  theme_minimal(base_size = 15) +
  theme(plot.title = element_text(hjust = 0.5, color = "darkblue", size = 18, face = "bold"))
```

Figure 19: Code for Graph comparing Logistic Regression Accuracies using L1 and L2 Regularization

Figure 20 shows that Ridge regularization outperforms Lasso regularization across all lambda values.

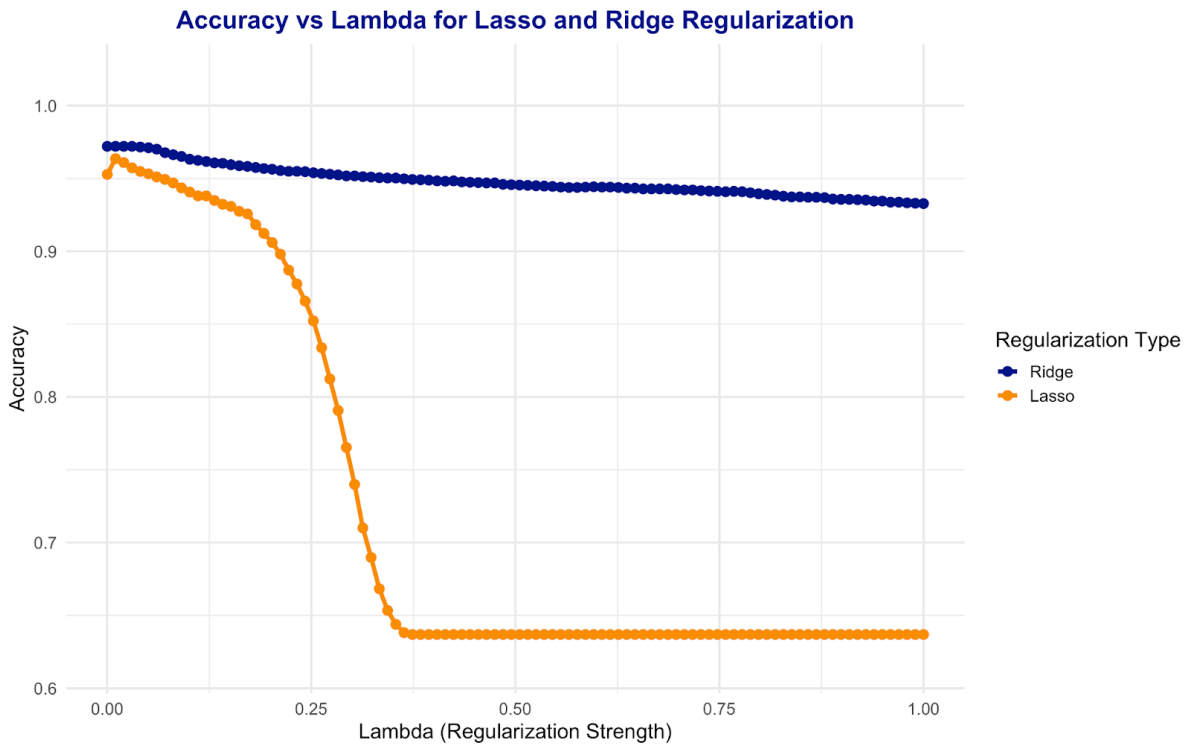


Figure 20: Graph plotting Accuracy vs lambda values for Lasso and Ridge Regularization

Ridge Regularization achieved the highest accuracy of 96.4% with a lambda of 0.0506, while Lasso Regularization reached a maximum accuracy of 95.9% with a lambda of 0.0102. Regularization positively impacts the model by reducing overfitting and improving performance, particularly through the addition of L2 regularization to the loss function of logistic regression.

Analysis

The KNN algorithm demonstrated statistical significance in breast cancer prediction, outperforming all but one of the models studied. The results of the algorithms are shown in Figure 21.

Model	Accuracy
K-means Clustering	0.922
KNN	0.963
Random Forest	0.958
Logistic Regression	0.964

Figure 21: Accuracy results of all machine learning algorithms on the test set

When tested on the test dataset with the optimal neighbor value of k=5, the KNN algorithm achieved a high accuracy of 96.3%. This demonstrates that the KNN model effectively utilized neighboring data points to make accurate predictions most of the time. In comparison, the K-means clustering algorithm achieved an accuracy of 92.2%, the random forest model reached 95.8%, and logistic regression outperformed KNN by a margin of 0.1%, achieving the highest accuracy at 96.4%.

Understanding why the KNN algorithm performed as it did—and outperformed most other algorithms—is crucial in assessing its ability to predict breast cancer. To gain deeper insight, its results will be compared with other models to analyze the factors influencing accuracy.

KNN attempts to group labeled data, which are of good quality in this data set and in this specific application, giving it an advantage over the K-means clustering algorithm. The KNN needs clustered data whilst the K-means clustering algorithm creates its own clusters resulting in a lower accuracy as more mathematical assumptions about the clusters need to be made. This results in the K-nearest neighbors having an accuracy of 4.1% more than K-means clustering algorithm.

Furthermore, the KNN algorithm performed slightly better than the random forest algorithm in this study, as it requires fewer hyperparameters to tune, making it more efficient for smaller datasets. While random forest is generally robust against noise due to its ensemble nature, in small datasets, the creation of multiple decision trees can sometimes lead to overfitting if not properly tuned. As a result, the KNN algorithm achieved an accuracy 0.5% higher than the random forest model.

However, the KNN algorithm was outperformed by the logistic regression algorithm by 0.1%. This is a very small percentage difference but is important to understand. To begin with, KNN has a small advantage over logistic regression as it can handle binary and multi-class classifications, whilst the logistic regression model is specified to handle only binary classification. However in this data set the classification is a binary one, benign or malignant. Therefore, the logistic regression model makes less mathematical assumptions and the KNN needs more mathematical assumptions.

In the context of predicting breast cancer the KNN model seems to surpass most machine learning models. There may be a few explanations for the reasons why the KNN did not perform as well such as its scalability as it does not scale well with large data sets and can end up using a lot of computing power and time. Additionally, KNN is slightly sensitive to incorrect data and

noise of data as it can be seen by the equation itself. This can have an effect on the accuracy specifically if there are large outliers in the data. However, the dataset used was a small one and most of the data was correct and contained little noise comparably. The KNN does have the advantage of multi-class classification yet in this dataset the classification was binary. Hence a specialized logistic regression model on binary classification was able to beat it by 0.1% which can be considered almost statistically insignificant. Moreover, this dataset used biopsies but with a dataset that would have more classes such as early-stage conditions like fibroadenoma, cysts, and adenosis, the KNN model would perform better.

As a result, the KNN is indeed an effective algorithm to be used in everyday life examples and with great accuracy can be used to predict breast cancer. Nonetheless, it is important to note that the KNN model does surpass algorithms in structured data (such as biopsies) while in unstructured data (such as images) would most likely underperform. Additionally, KNN may perform slightly worse in binary classification since it is not specifically designed for such tasks. However, it is likely to excel in multi-class classification, where its ability to group data points based on similarity becomes more advantageous. Therefore, in structured data KNN has the ability to be used in everyday applications to predict breast cancer.

Conclusion

Cancer is a disease that has negatively impacted peoples' lives and Breast Cancer has by far affected the most people. It is a very dangerous and fatal disease, and sometimes even if detected can not be treated. Nonetheless, in most of the cases early detection and treatment of

breast cancer can have a positive effect of completely removing it or inhibiting it for a long time period.

The KNN algorithm is one of the most widely used machine learning algorithms in healthcare. The reason for its widespread use is that it is very effective with small datasets, which are very common in healthcare because diseases are specific to people hence less data can be collected. Breast cancer is a more common type of disease and has the capability of having larger data sets, yet the collection of the data is not only a rigorous process but also an ethical one. KNN is a supervised machine learning algorithm that needs clear and ordered data.

Out of all the machine learning algorithms that were tested the KNN algorithm performed the second best. It is obvious that in the prediction of Breast Cancer, KNN is a very effective algorithm and has the potential to achieve high accuracy. By a very small percentage it is not the highest achieving, but that is a result of the data being very specific to the logistic regression model with the binary classification.

Nonetheless, there is still room for improvement, and to truly test the KNN algorithm to its maximum capability, we must utilize all available hyperparameters and fine-tune them thoroughly. In addition, employing an even larger dataset could potentially enhance its performance.

Works Cited

"Life expectancy in the United States from 1900 to 2021." Statista, 2021, https://www.statista.com/statistics/1040079/life-expectancy-united-states-all-time/. Accessed 24 Aug 2024

Breast Cancer. https://www.who.int/news-room/fact-sheets/detail/breast-cancer. Accessed 1 Sep. 2024.

Sarkar, M., and T. Y. Leong. ‘Application of K-Nearest Neighbors Algorithm on Breast Cancer Diagnosis Problem.’ Proceedings of the AMIA Symposium, 2000, pp. 759–63. PubMed Central, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2243774/. Accessed 29 Aug. 2024.

What Is Breast Cancer? https://www.cancer.org/cancer/types/breast-cancer/about/what-is-breast-cancer.html. Accessed 29 Aug. 2024.

Benco, Stefano. ‘Analyzing the Decision Tree Algorithm’. Math for Business, the Ultimate Applied Math Blog, 4 July 2019, https://mathforbusiness.com/data-science/analyzing-the-decision-tree-algorithm/. Accessed 4 Sept. 2024.

"Breast Cancer: Pictures, Lumps, Signs." Today, NBCUniversal Media, 6 Oct. 2021, https://www.today.com/health/breast-cancer/pictures-of-breast-cancer-lumps-signs-rcna5360. Accessed 30 Aug. 2024.

Breast Cancer. https://www.who.int/news-room/fact-sheets/detail/breast-cancer. Accessed 1 Sep. 2024.

Ellis, Christina. ‘Mtry in Random Forests’. Crunching the Data, 28 Aug. 2022, https://crunchingthedata.com/mtry-in-random-forests/. Accessed 4 Sept. 2024.

"Google AI Beats Doctors at Breast Cancer Detection—Sometimes." The Wall Street Journal, 30 Jan. 2020, https://www.wsj.com/articles/google-ai-beats-doctors-at-breast-cancer-detectionsometimes-1157 7901600. Accessed 30 Aug. 2024.

"K-Nearest Neighbors(KNN)". AlmaBetter, https://www.almabetter.com/bytes/tutorials/data-science/k-nearest-neighbors. Accessed 2 Sept. 2024.

‘K Means Clustering - Introduction’. GeeksforGeeks, 2 May 2017, https://www.geeksforgeeks.org/k-means-clustering-introduction/. Accessed 3 Sept. 2024.

"Life expectancy in the United States from 1900 to 2021." Statista, 2021, https://www.statista.com/statistics/1040079/life-expectancy-united-states-all-time/. Accessed 24 Aug. 2024.

‘Logistic Regression in Machine Learning’. GeeksforGeeks, 9 May 2017, https://www.geeksforgeeks.org/understanding-logistic-regression/. Accessed 6 Sept. 2024.

Logistic Regression. http://faculty.cas.usf.edu/mbrannick/regression/Logistic.html. Accessed 8 Sept. 2024.

‘L1 and L2 Regularization Methods, Explained’. Built In, https://builtin.com/data-science/l2-regularization. Accessed 8 Sept. 2024.

Radiology (ACR), Radiological Society of North America (RSNA) and American College of. ‘Stereotactic Breast Biopsy’. Radiologyinfo.Org, https://www.radiologyinfo.org/en/info/breastbixr. Accessed 1 Sep. 2024.

R: BRCA Dataset. https://search.r-project.org/CRAN/refmans/Przewodnik/html/brca.html. Accessed 1 Sept. 2024.

Sarkar, M., and T. Y. Leong. ‘Application of K-Nearest Neighbors Algorithm on Breast Cancer Diagnosis Problem.’ Proceedings of the AMIA Symposium, 2000, pp. 759–63. PubMed Central, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2243774/. Accessed 29 Aug. 2024.

Weisstein, Eric W. K-Means Clustering Algorithm. https://mathworld.wolfram.com/. Accessed 3 Sept. 2024.

What Is Breast Cancer? https://www.cancer.org/cancer/types/breast-cancer/about/what-is-breast-cancer.html. Accessed 29 Aug. 2024.

What Is Random Forest? | IBM. 20 Oct. 2021, https://www.ibm.com/topics/random-forest. Accessed 4 Sept. 2024.

What Is the K-Nearest Neighbors Algorithm? | IBM. 4 Oct. 2021, https://www.ibm.com/topics/knn.Accessed 2 Sept. 2024.