Name	**PSID**
Abdiel Loera	1870258
Anh Khoa Nguyen	2234305
Dong Tran	1615527
Gleici Pereira	2097712
Hyundoo Jeong	2212332

Advancing Breast Cancer Diagnosis:
A Predictive Modeling Approach Using the Breast Cancer Wisconsin (Diagnostic) Dataset

Breast cancer is a life-threatening disease that affects millions of individuals globally. The early detection can lead to improved patient outcomes, reduced treatment burden, and enhanced quality of life for those affected. Timely detection of this disease is pivotal, as it not only facilitates more effective treatment but also contributes to improved patient outcomes, diminished treatment burden, and an overall enhanced quality of life for those affected. Considering that, our primary research question revolves around the development of a robust predictive model, leveraging the Breast Cancer Wisconsin (Diagnostic) Dataset. The overarching objective is to create a model capable of accurately forecasting the diagnosis of breast masses, classifying them as either benign or malignant.

This single, focused research question aims to address a pivotal aspect of breast cancer diagnosis, concentrating on a singular response variable, which is the nature of the diagnosis. The Breast Cancer Wisconsin (Diagnostic) Dataset, sourced from the UCI Machine Learning Repository and contributed by Dr. William H. Wolberg, is particularly well-suited for this investigation and serves as a valuable resource for training and evaluating our predictive model. With 569 observations and 32 variables, it provides a comprehensive set of data points that encompass both real-valued features related to cell nuclei characteristics and "worst" measurements representing the mean of the three largest values observed for each feature.

The dataset in consideration comprises essential features such as radius_mean, texture_mean, perimeter_mean, and other attributes, which are key indicators of the characteristics of cell nuclei in breast masses. Leveraging this rich dataset, our model seeks to recognize patterns and relationships within these features to predict the likelihood of malignancy and contribute to the advancement of early detection strategies, improving patients' chances of survival and well-being.

Methodology

To develop a predictive model for breast cancer diagnosis, we employed two distinct supervised learning algorithms: K-Nearest Neighbors (KNN) and Support Vector Machines (SVM). The choice of these algorithms was driven by their well-established effectiveness in classification tasks and their ability to handle the complexity of medical datasets.

K-Nearest Neighbors (KNN):

The KNN algorithm is a simple yet powerful method for classification. The process involves calculating the Euclidean distance between the features of a given observation and those of its K-nearest neighbors in the training dataset. The prediction for the observation is then determined by a majority vote among its neighbors. The formula for predicting the response variable, Y_i for the *i*-th observation is as follows:

$$Y = \text{MajorityVote}\left(Y'_{neighbor_1}\ Y'_{neighbor_2}\ \cdots ,\ Y_{neighbor_K}\right)$$

In our implementation, we used the built-in function in R *knn()*. Where *knn()* requires 4 parameters: *knn*(train, test, cl, k).

- train: matrix of predictors associated with training data.
- test: matrix containing the predictors associated with the data for which we wish to make predictions.
- cl: vector containing the class labels (Y values) for the training observations.
- k: a value for K, the number of nearest neighbors to be used by the classifier.

Support Vector Machines (SVM):

SVM is a versatile algorithm capable of handling both linear and non-linear classification tasks. For our breast cancer diagnosis model, we employed SVM with selected kernels, including the radial kernel. A kernel is a function that maps the original feature space into a higher-dimensional space, where the data becomes more separable. Kernels measure the similarity between pairs of observations to create boundaries of various shapes. The hyperplane equation corresponding to the SVM model with the radial kernel are as follows:

$$K(\mathbf{x}_i, \mathbf{x}_j) = exp(-\gamma \sum_{k=1}^{p} (x_{ik} - x_{jk})^2), \gamma > 0$$

Similarities between observations are measured considering their Euclidean distance. High similarity is assigned to nearby points and low similarity to distant points. Gamma controls the flexibility of the decision boundary. When gamma is small, the curvature is low, and the decision region is broad resulting in a better generalization. When gamma is large, the curvature is high, and the decision region is narrower resulting in potential overfitting.

Models' Robustness and Advantages/Disadvantages:

- K-Nearest Neighbors (KNN):
  Advantages: Simple, easy to implement, and effective in capturing local patterns.
  Disadvantages: Sensitive to outliers and requires careful selection of K.

- Support Vector Machines (SVM):
  Advantages: Effective in high-dimensional spaces, robust to overfitting, and versatile with various kernel functions.
  Disadvantages: Computationally expensive for large datasets.

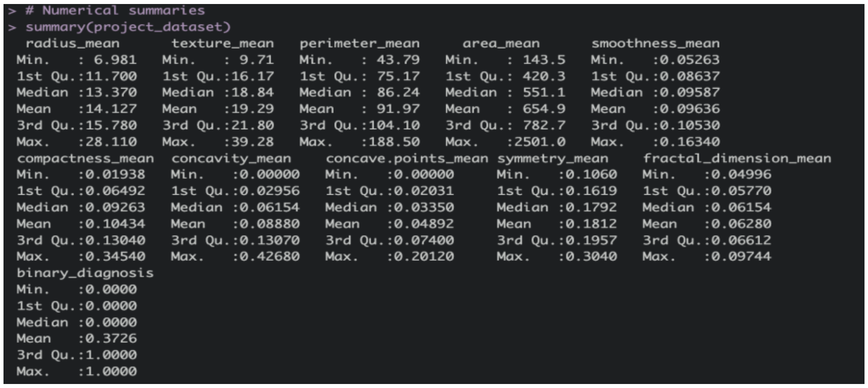
Assessment of Robustness:

To assess the robustness of both models, we employed cross-validation techniques. For KNN, we utilized simple validation set approach and Leave-One-Out Cross-Validation (LOOCV), ensuring that each observation served as both a training and test set. For SVM, we also employed validation set approach and K-fold cross-validation, where the dataset was divided into K subsets, and the model was trained and tested K times, rotating the subsets.

These cross-validation techniques were crucial in evaluating the models' generalizability and determining their effectiveness in predicting breast cancer diagnoses. The subsequent analysis aimed to compare the models' performance and identify the one with superior accuracy and predictive capabilities.

Data Analysis

Numerical and Graphical summaries of the dataset



```
> # Numerical summaries
> summary(project_dataset)
  radius_mean      texture_mean   perimeter_mean     area_mean      smoothness_mean
 Min.   : 6.981   Min.   : 9.71   Min.   : 43.79   Min.   : 143.5   Min.   :0.05263
 1st Qu.:11.700   1st Qu.:16.17   1st Qu.: 75.17   1st Qu.: 420.3   1st Qu.:0.08637
 Median :13.370   Median :18.84   Median : 86.24   Median : 551.1   Median :0.09587
 Mean   :14.127   Mean   :19.29   Mean   : 91.97   Mean   : 654.9   Mean   :0.09636
 3rd Qu.:15.780   3rd Qu.:21.80   3rd Qu.:104.10   3rd Qu.: 782.7   3rd Qu.:0.10530
 Max.   :28.110   Max.   :39.28   Max.   :188.50   Max.   :2501.0   Max.   :0.16340
 compactness_mean  concavity_mean    concave.points_mean symmetry_mean    fractal_dimension_mean
 Min.   :0.01938   Min.   :0.00000   Min.   :0.00000     Min.   :0.1060   Min.   :0.04996
 1st Qu.:0.06492   1st Qu.:0.02956   1st Qu.:0.02031     1st Qu.:0.1619   1st Qu.:0.05770
 Median :0.09263   Median :0.06154   Median :0.03350     Median :0.1792   Median :0.06154
 Mean   :0.10434   Mean   :0.08880   Mean   :0.04892     Mean   :0.1812   Mean   :0.06280
 3rd Qu.:0.13040   3rd Qu.:0.13070   3rd Qu.:0.07400     3rd Qu.:0.1957   3rd Qu.:0.06612
 Max.   :0.34540   Max.   :0.42680   Max.   :0.20120     Max.   :0.3040   Max.   :0.09744
 binary_diagnosis
 Min.   :0.0000
 1st Qu.:0.0000
 Median :0.0000
 Mean   :0.3726
 3rd Qu.:1.0000
 Max.   :1.0000
```

The summary statistics for the breast cancer dataset provide a comprehensive snapshot of key features related to cell nuclei characteristics. The range, mean, and median values for attributes such as radius, texture, and perimeter offer insights into the size and shape variations of cell nuclei. Additionally, descriptors like smoothness, compactness, and concavity shed light on the textural and irregular shape characteristics. The distribution of the binary diagnosis variable indicates that, on average, approximately 37% of the cases are classified as malignant. This numerical summary serves as an initial guide for understanding the dataset, laying the groundwork for further exploration and analysis of potential patterns or correlations within the data

1. **K-Nearest Neighbors (Gleici Pereira, Hyundoo Jeong, Abdiel Loera)**

For our KNN model, we will manipulate our dataset so we can work better with the data. We will exclude the column id, since it is a variable only for identification purposes and not important for research, exclude all columns with standard deviation and worst measurements for each variable, and we will transform the response variable *diagnosis* into a binary variable that will contain a 1 if the response is M (malignant) and 0 if the response is B (benign), and since our variables are not in the same scale we will also scale the data so it can produce more meaningful and accurate results.

The dataset we will be working on will be a concise dataset *project_dataset* with only mean measurements for all variables and the binary variable *binary_diagnosis* which helps us to better use and assess the results of the response variable *diagnosis*.

Validation Set Approach

Our initial step involved the meticulous process of choosing the most effective tuning parameters for the K-Nearest Neighbors (KNN) model. For that, we used the validation set approach initially.

Model Parameter Tuning for KNN:

To ascertain the optimal parameter for KNN, a range of values from K = 1 to K = 20 was systematically employed. Subsequently, each model was fitted, and the associated test errors were meticulously assessed. The paramount goal was to identify the K value that minimizes the test error, signifying an optimal balance between bias and variance.

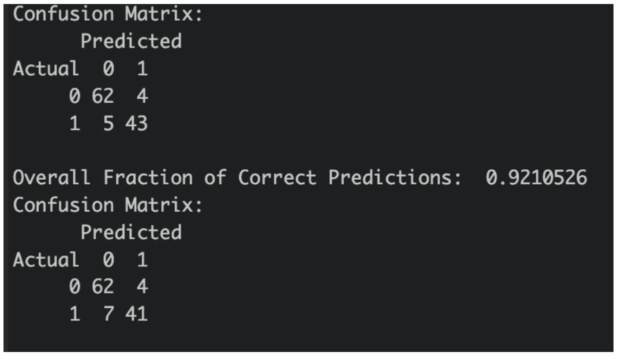
We proceeded by training the model and selecting the optimal K that yielded the smallest test error:

```
train.response <- project_dataset[train.dataset, "binary_diagnosis"]
test.response <- project_dataset[-train.dataset, "binary_diagnosis"]
k.set <- seq(1, 20, by = 1)
test.error <- numeric(length(k.set))
set.seed(1)
for (i in 1:length(k.set)) {
  knn.results <- knn(train = train.predictors, test = test.predictors,
                     cl = train.response, k = k.set[i])
  test.error[i] <- mean(knn.results != test.response)
}
```

```
## Returns K value with best test error rate
> min(test.error)
[1] 0.07894737
> which.min(test.error)
[1] 2
```

From our results, K = 2 gives us the best test error of 7.89% using the validation set approach.

Confusion Matrix:



```
Confusion Matrix:
      Predicted
Actual  0  1
     0 62  4
     1  5 43

Overall Fraction of Correct Predictions:  0.9210526
Confusion Matrix:
      Predicted
Actual  0  1
     0 62  4
     1  7 41
```

The confusion matrix illustrates the performance of the k-Nearest Neighbors (KNN) algorithm for different k values in classifying breast cancer cases as benign or malignant. With k ranging from 1 to 20, the models consistently demonstrate high accuracy, achieving an overall fraction of correct predictions between 90.35% and 92.11%. Notably, the confusion matrix for k = 3 shows 62 true negatives, 43 true positives, 4 false positives, and 5 false negatives, indicating effective discrimination between benign and malignant cases. These results suggest that the KNN algorithm performs well in accurately categorizing breast cancer instances, with minimal misclassifications.

Leave-one-out Cross-Validation (LOOCV) Approach

We proceed by testing our model with a better cross-validation technique. In LOOCV, each observation is used as a validation set while KNN model that we created is trained on all observations(diagnosis). This process is repeated for each observation in the dataset, and the average error rate is calculated. We performed LOOCV by iteratively leaving out one data point at a one time, training model on the remaining dataset, and evaluating its performance on the omitted point as KNN model is trained for varying values of K (1...20) and obtained optimal K = 2, and the test error is calculated by for K.

Source code:

```
#perform LOOCV
for(i in 1:nrow(scaled_data)){
  train_predictors <- scaled_data[-i, ]
  train_response <- project_data$binary_diagnosis[-i]

  test_predictor <- as.matrix(scaled_data[i, , drop = FALSE])
  test_response <- project_data$binary_diagnosis[i]

  # Perform K = 2 (Based on K-value with best test error rate in KNN)
  knn_result <- knn(train = train_predictors, test = test_predictor, cl = train_response, k = 2)
  loocv_errors[i] <- as.numeric(knn_result != test_response)
}
loocv_error_rate <- mean(loocv_errors)
```

```
print(loocv_error_rate)
[1] 0.08098592
```

From our results, LOOCV test error estimate is 8.1% based on previous KNN Model.

2. Support Vector Machines (Anh Khoa Nguyen, Hyundoo Jeong)

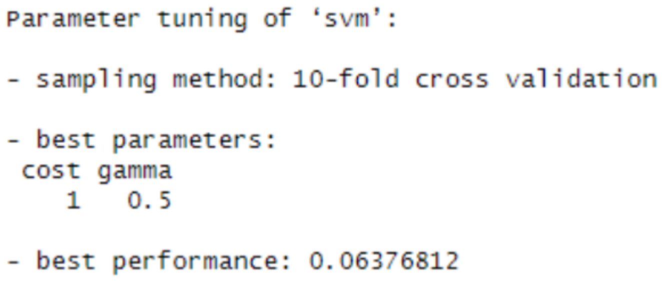
For our SVM model, we also need only the mean value for all the predictors like radius, texture, area, smoothness, etc. We also don’t want the “id” or standard deviation and the worst values of those columns, because knowing those values doesn’t really help us predicting whether a patient has breast cancer or not, plus, the standard deviation and the worst values are not highly correlated with the mean values, so we just go ahead and create a new data frame with only the diagnosis and all the mean columns of the predictors. And I believe we do need to scale the data, since the values for the measurements are scaled differently, so we want to scale our dataset for the best performance.

For our specific dataset, the “diagnosis” column is the response variable, and its classifying M = malignant, B = benign, but we want to convert that into numeric so that we can work with SVM, so our new column will be “diagnosis_numeric” for diagnosis is 0 for benign, 1 for malignant

We proceed to use the tune function to find our optimal value of cost and γ. We then proceed to train our model after getting the optimal values. After training our model, we then create a table to show our predictions and record our prediction errors on testing data.

Source code:

```
> tune.out=tune(METHOD =svm,diagnosis_numeric~., data=
scale_df[train,],kernel="radial",ranges=
list(cost=c(0.1,1,10,100,1000),gamma=c(0.5,1,2,3,4)),type='C-classification')
> summary(tune.out)
```



```
Parameter tuning of ‘svm’:

- sampling method: 10-fold cross validation

- best parameters:
 cost gamma
    1   0.5

- best performance: 0.06376812
```

```
> svm_model <- svm(diagnosis_numeric~., data= scale_df[train, ], kernel="radial", cost=1,
gamma=0.5)
> svm.pred <- predict(svm_model, scale_df[-train,])
> table(true=scale_df[-train,"diagnosis_numeric"],pred=svm.pred)
```

```
true  0  1
   0 69  2
   1  5 38
```

From the table, we can compute our testing error is 7/114 = 0.0614 = 6.14%, which is a small error and a satisfactory performance.

K-fold Cross-Validation for SVM

We proceeded to test another powerful cross-validation technique. K-fold cross-validation is a powerful technique employed in machine learning to evaluate the performance of a Support Vector Machine (SVM) model. It optimally utilizes the available data by dividing the dataset into k subsets or 'folds.' The SVM model is trained and evaluated k times, with each iteration employing a different fold as the test set and the remaining data as the training set. We perform K-fold cross-validation based on SVM model, which is trained in advance, so I choose number of folds is 10 according to above summarizing result in SVM.

```
# Calculate and print test error rate
test_error_svm <- mean(svm.pred != scale_df$diagnosis_numeric)
print(test_error_svm)
[1] 0.1672535
```

In our results, k-fold CV test error rate is 16.72% which is significantly higher than other methods, so it is not significantly a good method for this specific dataset.

Comparing Models and Test Error Rates

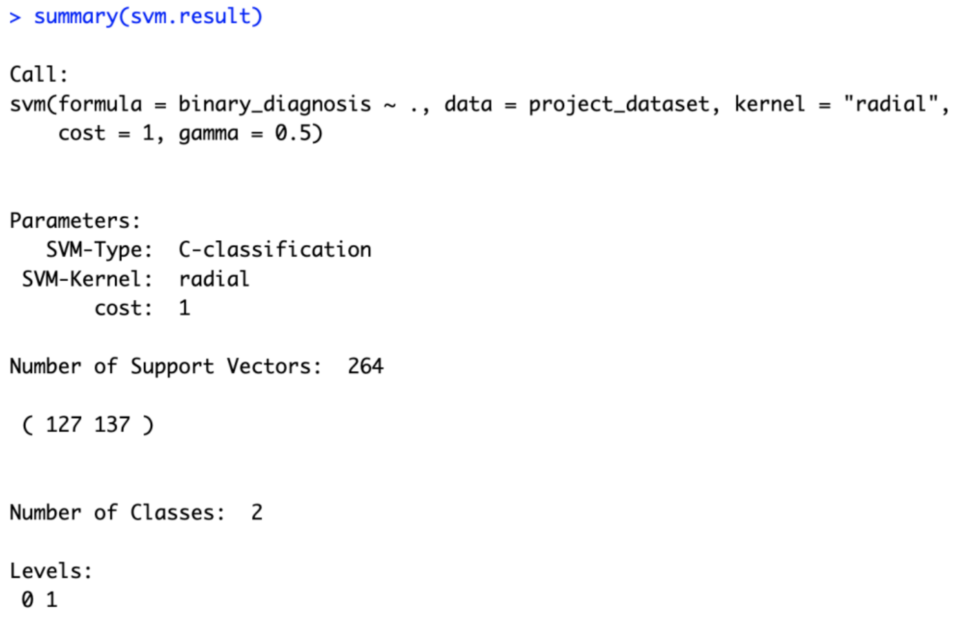
The SVM model demonstrated a better performance, achieving the lowest test error rate of 6.14% through cross-validation. In contrast, the best rate given by the KNN model, assessed using the cross-validation set approach, exhibited a slightly higher test error rate of 7.89%.

Consequently, based on the specific characteristics of this dataset, SVM outperforms KNN, establishing itself as the more effective model. The optimal parameters identified for SVM, namely cost = 1 and gamma = 0.5, have been employed in the training of the ultimate model for enhanced predictive accuracy.

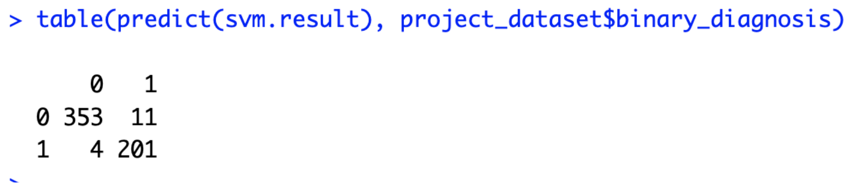
Best Model Fitting

Using the optimal parameters cost = 1 and gamma = 0.5, we proceeded to train the data. The following are a few steps taken, the results of our fitted model, and some important summaries of the SVM object.

```
#Fit best model
set.seed(1)
svm.result <- svm(binary_diagnosis~., data = project_dataset, kernel = "radial",
                  cost = 1, gamma = 0.5, scale = TRUE)
```



```
> summary(svm.result)

Call:
svm(formula = binary_diagnosis ~ ., data = project_dataset, kernel = "radial",
    cost = 1, gamma = 0.5)


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  radial
       cost:  1

Number of Support Vectors:  264

 ( 127 137 )


Number of Classes:  2

Levels:
 0 1
```



```
> table(predict(svm.result), project_dataset$binary_diagnosis)

      0   1
  0 353  11
  1   4 201
`
```

```
# Test error rate
> mean(predict(svm.result) != project_dataset$binary_diagnosis)
[1] 0.02636204

#Prediction Accurary
mean(predict(svm.result) == project_dataset$binary_diagnosis)
[1] 0.973638

#Plot the model
plot(svm.result, project_dataset)
plot(svm.result, project_dataset, formula = ~binary_diagnosis)
```

Results

In evaluating the results, our SVM model exhibited a commendable predictive performance, achieving a test error rate of 2.64% on the large Breast Cancer Wisconsin dataset. This signifies a meaningful ability to accurately classify instances of malignancy or benignity based on the chosen features. However, it is noteworthy that our k-fold cross-validation resulted in a relatively higher test error rate of 16.72%.

To enhance the predictive accuracy further, we recognize the potential to refine the model by either incorporating additional relevant variables or delving deeper into the interrelationships between existing ones. This iterative process aims to identify and eliminate unnecessary features, thereby improving the model's ability to generalize unseen data.

In summary, while our SVM model demonstrates respectable performance, the pursuit of refinement through feature engineering and a deeper understanding of variable relationships remains pivotal for achieving even higher predictive accuracy in the context of this complex dataset.

Conclusion

In addressing the primary research question framed in the introduction, our project focused on developing a robust predictive model for breast cancer diagnosis using the Breast Cancer Wisconsin (Diagnostic) Dataset. The goal was to create a model capable of accurately classifying breast masses as either benign or malignant, contributing to the advancement of early detection strategies.

Upon completing the data analysis, our findings indicate that the Support Vector Machine (SVM) model surpassed the K-Nearest Neighbors (KNN) model in terms of predictive performance. The SVM model achieved a lower test error rate of 2.64%, compared to the 7.89% error rate of the KNN model. This outcome underscores the efficacy of leveraging SVM in discerning patterns within the dataset's features related to cell nuclei characteristics. The SVM model's optimal parameters, identified as cost = 1 and gamma = 0.5, were employed in the final model, emphasizing the importance of parameter tuning in enhancing predictive accuracy.

While the project yielded valuable insights, some challenges were encountered during the data analysis. The relatively higher test error rates, particularly with the KNN model and SVM performing K-fold cross validation, indicate the complexity of the dataset and underscore the need for further refinement. To address these challenges and enhance the data analysis we could have done a more in-depth investigation into the interrelationships between variables which could unveil subtle dependencies, contributing to a more nuanced understanding of breast cancer characteristics, or perhaps combining predictions from multiple models, which could harness the strengths of different algorithms and potentially yield a more robust and accurate predictive model.

Ultimately, while our SVM model demonstrates promising predictive capabilities, continuous refinement and exploration of the dataset's complexities are essential for advancing the field of breast cancer diagnosis. The challenges encountered during the analysis provide valuable insights for future research aimed at enhancing the accuracy and reliability of predictive models in this critical domain.

Source

Dataset sourced from the UCI Machine Learning Repository (http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)). Originally obtained from the University of Wisconsin Hospitals, Madison, and the dataset's primary contributor is Dr. William H. Wolberg.