Detecting Malignant Tumors with Machine Learning Techniques

(Fine Needle Aspiration (FNA) of Breast Tissue Masses Analysis)

Company: F.P.T.

Founders:
Alejandro Pesantez

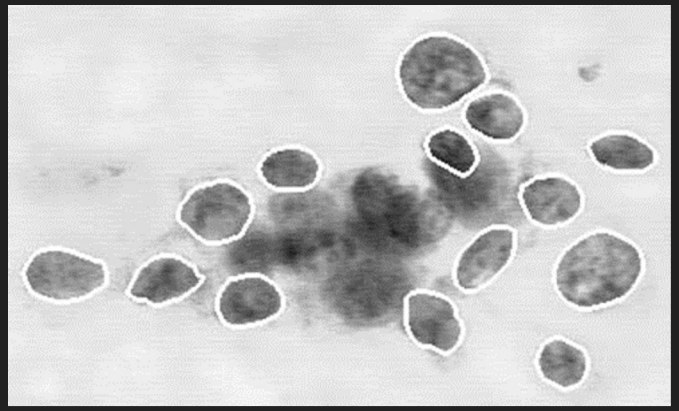
Overview

1. Study Background
2. Exploratory Data Analysis (EDA)
3. Supervised Machine Learning (ML) Algorithm
4. Results
    a. Decision Tree Output
    b. Bagging/ Random Forest Output
    c. K Nearest Neighbor (KNN) Output
    d. Receiver Operating Characteristic (ROC) Curves
5. Discussion
6. Conclusion
7. References

Study Background

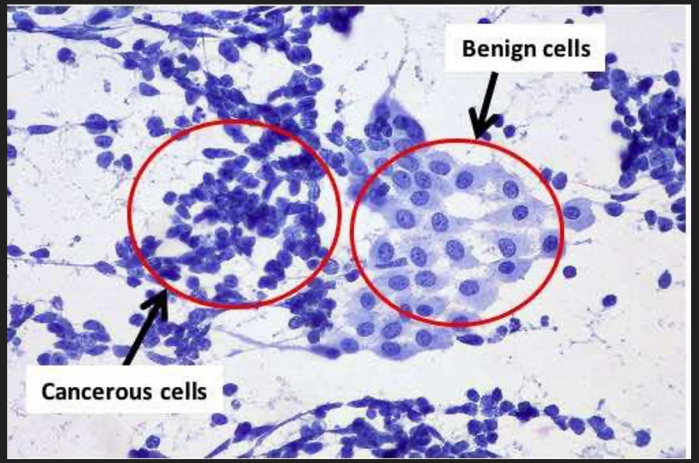
Original FNA Dataset: 569 samples (357 Benign, 212 Malignant) of 10 characteristics for the nuclei. The data set consists of the mean (average), standard error of the mean (precision of the sample mean), and the worst feature of the 10 characteristics, for a total of 30 observations for each.

1. radius (mean/average of distances from center to points on the perimeter)
2. texture (standard deviation/dispersion of gray-scale values)
3. perimeter (boundary total distance)
4. area (number of pixels and adding half to perimeter)
5. smoothness (local variation in radius lengths)
6. compactness (perimeter^2/ area – 1.0)
7. concavity (severity of concave portions of the contour)
8. concave points (number of concave portions)
9. symmetry (segment measurements)
10. fractal dimension ("coastline approximation“ -1)



Picture Source: Mangasarian, Olvi & Street, Nick & Wolberg, William. (1970). Breast Cancer Diagnosis and Prognosis Via Linear Programming. Operations Research. 43. 10.1287/opre.43.4.570.

Exploratory Data Analysis (EDA)

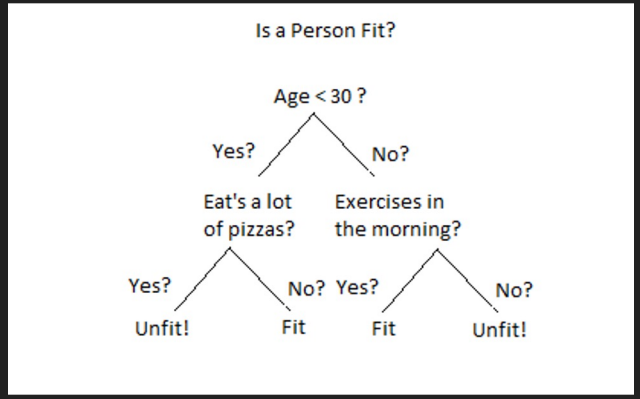


Picture Source: Hussain, M. (2020, February 11). *Bio-informatics, the data science of Biotechnology.* Medium. Retrieved April 7, 2022, from https://medium.com/thecyphy/bio-informatics-the-data-science-of-biotechnology-f5b86306c33d

- Fine Needle Aspiration of Breast Tissue Data
  - No NA's in the original data
  - Rescaled data to range from 0 to 1
- Final Data Set
  - 569 observations by 11 variables
  - Predictors: The means of radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, and fractal dimension
  - Response Variable: Diagnosis (Malignant/Benign)

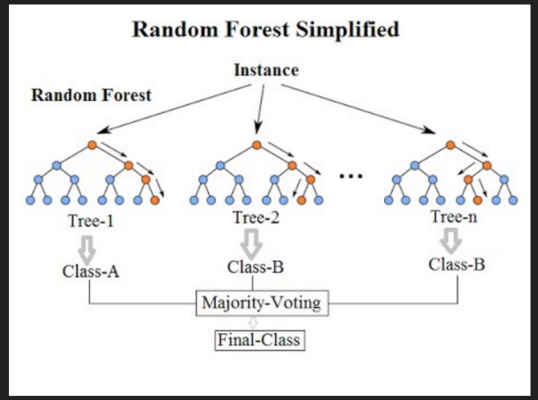
Supervised Machine Learning (ML) Algorithm

Decision Tree



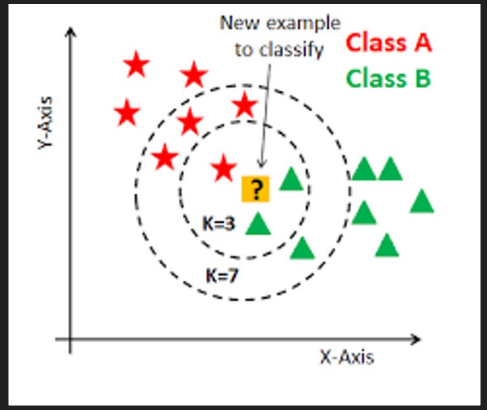
- Recursive partitioning model
- Splits the data to make more pure groups
- Can tune model by cross validating and pruning to avoid overfitting

Bagging/Random Forest



- Builds decision trees on different samples and takes their majority vote for classification and average in case of regression.
- Bagging uses the number of variables randomly sampled at each split from all variables in the data
- To compare models you can use the Out of Bag error rate
- Hyperparameters are ntree and mtry

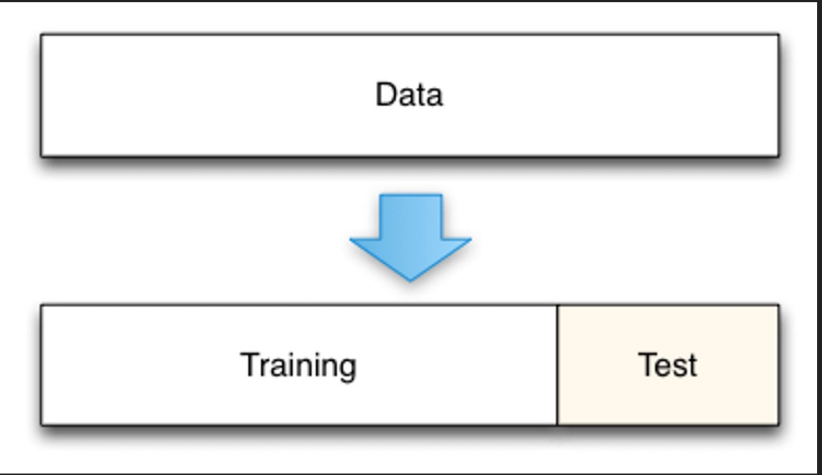
K Nearest Neighborhood (KNN)



- Works by finding the distances between a query and all the examples in the data.
- Selects the specified number examples (K) closest to the query
- Then votes for the most frequent label (in the case of classification) or averages the labels (in the case of regression)
- Only has one hyperparameter K

Supervised Machine Learning (ML) Algorithm

- Decision Tree
  - Split Data into Train (80%) and Test (20%) data sets
  - Created bigger model predicting diagnosis, where all variables from data used are included
  - Tuned model with CP plot and picked the value of CP that had the lowest relative error for new decision tree
- Bagging/ Random Forest
  - Split Data into Train (80%) and Test (20%) data sets
  - Started with bagging model to predict diagnosis so we could include all variables in model
  - Looked at Variable importance and noticed five were most important
  - Made the number of variables randomly sampled at each split five for final model
- K Nearest Neighborhood (KNN)
  - Split Data into Train (80%) and Test (20%) data sets
  - Looked at rule of thumb and set K as square root of N
  - Tuned models by changing K to see which model had the highest accuracy, and chose final model to be the value of K that produced the model with highest accuracy

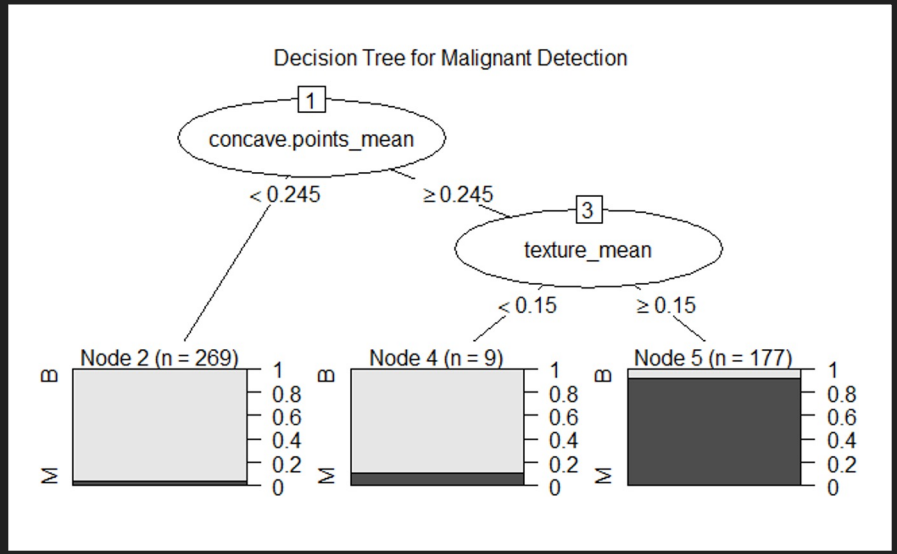


Picture Source: Sen, P. (2020, June 16). *Only train and test set is not enough for generalizing ML model, significance of validation set.* Medium. Retrieved April 7, 2022, from https://medium.com/analytics-vidhya/only-train-and-test-set-is-not-enough-for-generalizing-ml-model-significance-of-validation-set-cf68bb26881a

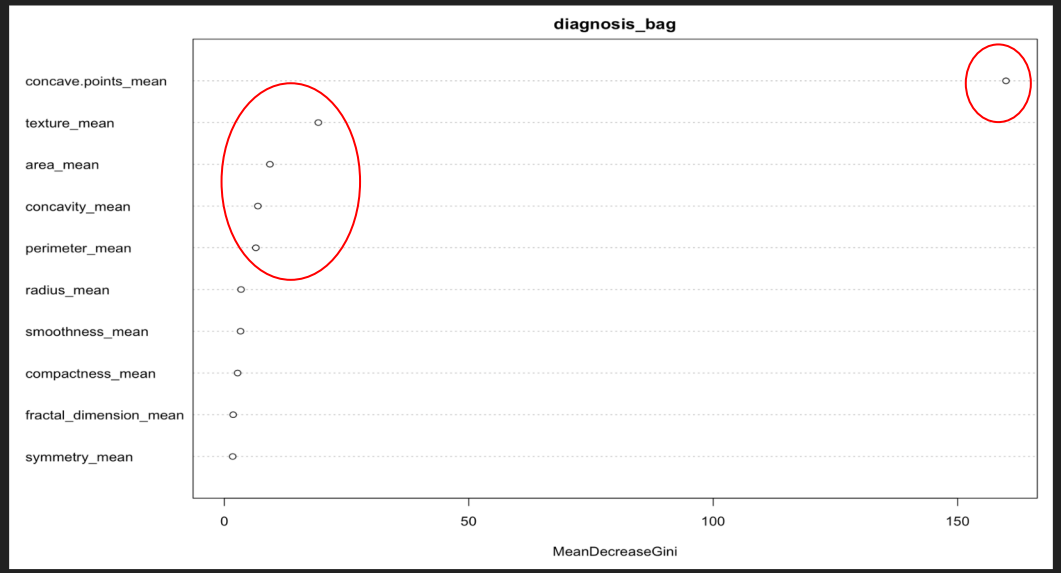
Results

Decision Tree Output

- Classification Model - Flow Chart Structure/Tree-like Structure
- Tuned model (Pruned Tree) with Complexity Parameter for lowest relative error
- Final Tree uses the mean values for Concave Points and Texture
  - Accuracy = 0.886
    - (11.4% misclassified)
  - Sensitivity = 0.7895
    - (21.05% False Negative)
  - Specificity = 0.9342
    - (6.58% False Positive)



Bagging/ Random Forest Output



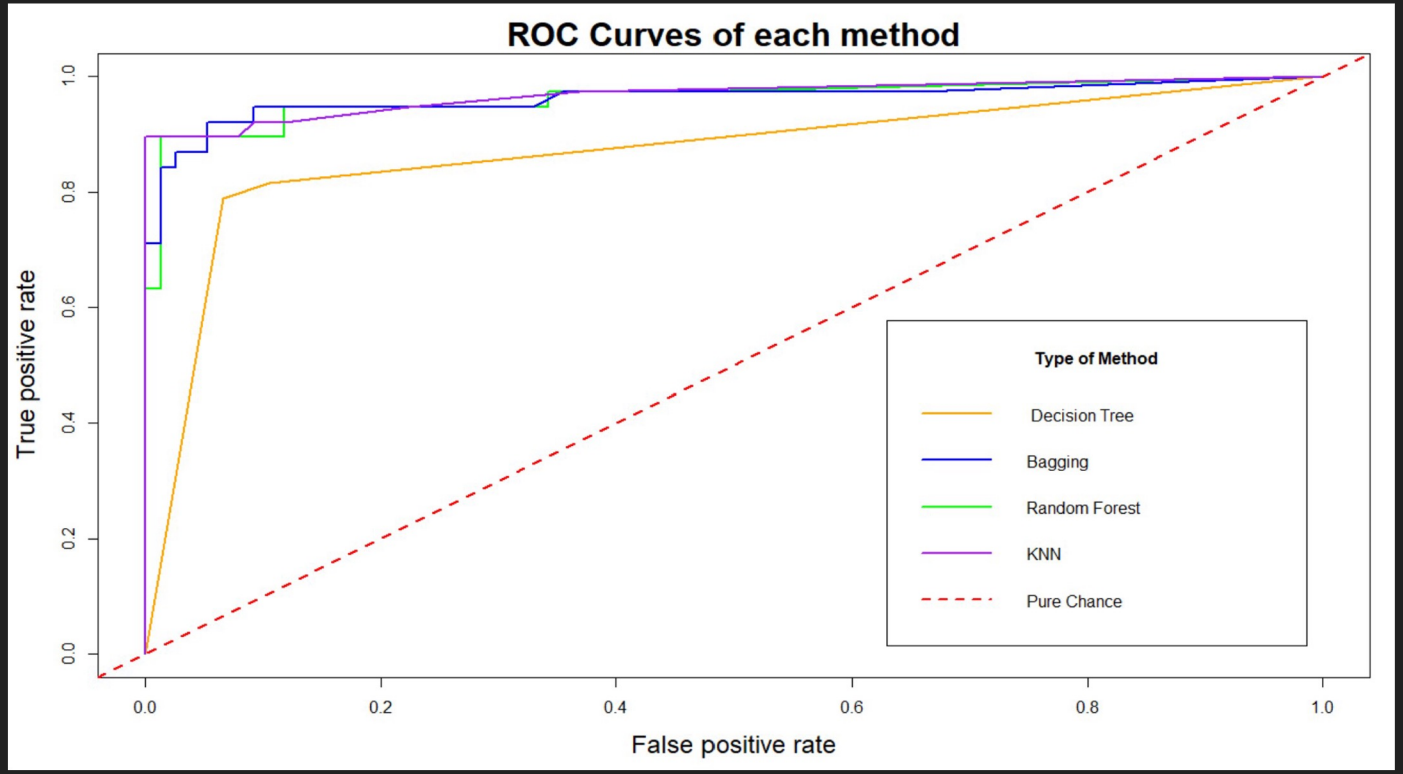
- Used ntree = 201
- Used mtry =
  - 10 for bagging
  - 5 for random forest
- Bagging
  - Accuracy = 0.9386
  - Specificity = 0.9605
  - Sensitivity = 0.8947
- Random Forest
  - Accuracy = 0.9298
  - Specificity = 0.9474
  - Sensitivity = 0.8947

KNN Output

Rule-of-Thumb: Optimal K value = $\sqrt{n} = \sqrt{569} \cong 23.8537$

	K = 3	K = 5, 7, 9	K = 11	K= 13,15	K = 17, 19	K = 21	K = 23	K = 25
Accuracy	.9298	.9474	.9561	.9561	.9474	.9474	.9386	.9474
Sensitivity	.8947	.8947	.8947	.8947	.8684	.8684	.8158	.8421
Specificity	.9474	.9737	.9868	.9868	.9868	.9868	1.0000	1.0000

Receiver Operating Characteristic (ROC) Curves



Ideal curve: AUC = 1
Pure Chance: AUC = 0.5

Type of Method	AUC
Decision Tree	0.87
Bagging	0.9621
Random Forest	0.9616
KNN	0.9675

Discussion

	Decision Tree	Bagging	Random Forest	KNN
Misclassification rate	0.114	0.0614	0.0702	0.0439
Accuracy	0.886	0.9386	0.9298	0.9561
Sensitivity	0.7895	0.8947	0.8947	0.8947
Specificity	0.9342	0.9605	0.9474	0.9868

Conclusion

- Decision tree
  - Performed the least best overall
- KNN
  - Performed the best overall
- Best ML Method in detecting malignant tumors in descending order:
  - KNN, Bagging, Random Forest, and Decision Tree
- Our recommendation: KNN method

References

- Woodard, R. 2022, February 10-24. [Lecture recording]. Notre Dame University.
  - *Supervised Learning (Part I)*
    - https://canvas.nd.edu/courses/33648/modules/items/157848
    - https://canvas.nd.edu/courses/33648/modules/items/157850
    - https://canvas.nd.edu/courses/33648/modules/items/157852
    - https://canvas.nd.edu/courses/33648/modules/items/157862
  - *Supervised Learning (Part II)*
    - https://canvas.nd.edu/courses/33648/modules/items/157870
  - *Supervised Learning (Part III)*
    - https://canvas.nd.edu/courses/33648/modules/items/157884
    - https://canvas.nd.edu/courses/33648/modules/items/157886
    - https://canvas.nd.edu/courses/33648/modules/items/157914
    - https://canvas.nd.edu/courses/33648/modules/items/157916
- https://stackoverflow.com/questions/11741599/how-to-plot-a-roc-curve-for-a-knn-model
- https://stackoverflow.com/questions/13956435/setting-values-for-ntree-and-mtry-for-random-forest-regression-model
- Decision Tree Image - Slide 6
  - Picture Source: Kulkarni, M. (2017, September 7). *Decision Trees for Classification: A Machine Learning Algorithm*. Xoriant. Retrieved April 7, 2022, from https://www.xoriant.com/blog/product-engineering/decision-trees-machine-learning-algorithm.html
- Random Forest/Bagging Image - Slide 6
  - Picture Source: Koehrsen, W. (2017, December 27). *Random Forest Simple Explanation*. Medium. Retrieved April 7, 2022, from https://williamkoehrsen.medium.com/random-forest-simple-explanation-377895a60d2d
- KNN Image - Slide 6
  - Picture Source: Navlani, A. (2018, August 2). *KNN classification tutorial using Sklearn Python*. DataCamp Community. Retrieved April 7, 2022, from https://www.datacamp.com/community/tutorials/k-nearest-neighbor-classification-scikit-learn

Thank you for listening!

FPT@nd.edu