A Robust Model For Diagnostic Breast Cancer

Frohan Foroutan, 100871200

Mohamed Hozayen, 100997960

DATA5000 – Introduction to Data Science

School of Computer Science, Carleton University

Dr. Elio Velazquez

April 15, 2020

Contents

1 Motivation

There are over 100 types of cancer that ruthlessly affects people around the world. Excluding non-melanoma skin cancers, breast cancer is the most common cancer and the second leading cause of death from cancer among Canadian women [1]. Canadian Cancer Society estimates that every day, average 74 Canadian women are diagnosed with breast cancer where, on average, 14 will die from breast cancer. Early detection of breast cancer can greatly improve prognosis and survival chances by promoting early clinical treatment to patients; thus, making use of Machine Learning more important and it has become a necessity in cancer research. It is fascinating to be able to possibly save a life by using data.

Additionally, as future and current academics, we need to take our obligation to society more seriously. We need to be self-conscious of the privileges granted to us by society, in tenure and our intellectual freedoms and academic lifestyle, come in exchange for the value we are expectedly to produce. Thus, we initially hoped to further motivate this diverse group of peers to make social impact, more specifically the health and wellbeing of people, the starting point of their work rather than something we say will happen later, be done by someone else, or magically happen on its own. Such intentions to reduce the burden on the healthcare professionals become more important than ever in face of the corona-virus pandemic.

2 Research Question

Cells are the smallest unit in body that makes up all the tissues and organs. New cells replace old and damaged cells when they die. Cancer starts when cells begin to grow rapidly and out of control which results in forming a tumor [2]. Tumor is often categorized as benign, meaning not dangerous, or malignant which has the potential to be dangerous. Benign tumors are not considered cancerous meaning their cells are close to normal in appearance, they grow slowly, and they do not spread to other parts of the body. Malignant tumors are cancerous. This type of cells if left alone, can spread to other parts of the body. As result, the main question in detecting a cancer, and thus our research question, is identifying whether a cell is benign or malignant.

3 Literature Review

In the literature, several Artificial Intelligence (AI) techniques are applied for breast cancer diagnosis to improve classification accuracy using the Wisconsin Breast Cancer Diagnosis (WBCD) Dataset [3]. A genetically optimized neural network (GONN) is proposed in [4] introducing new crossover and mutation operators. Performance matrices used are accuracy, sensitivity, specificity, Receiver Operator Characteristics (ROC) curves, and Area-Under-the-Curve (AUC) under the ROC curve with classical back propagation models. In [5], a computer-based method applies multilayer perceptron (MLP) neural network based on enhanced non-dominated sorting genetic algorithm (NSGA-II) to optimize accuracy and network structure. However, in [6], the model uses gain directed simulated annealing genetic algorithm wrapper (IGSAGAW) to remove most irrelevant features, and cost sensitive support vector machine (CSSVM). Furthermore, in [7], the model uses SVM-based ensemble learning structures to improve performance. Finally, in [8], a comparison is presented among several machine learning algorithms including SVM and k-Nearest Neighbors (k-NN). Table 1 shows a concise comprehensive summary.

Notably, the literature lacks a coherent comprehensive machine learning architecture design including feature engineering, meta-learning, testing, and appropriate performance matrices for class imbalance in WBCD dataset.

Table 1: Literature review summary of AI techniques on WBCD dataset.

Criteria	**Algorithm Used**	**Performance measurements**			**Limits**
Reference		**Accuracy**	**Specificity**	**Sensitivity**	
[4]	GONN	✓	✓	✓	Small dataset
[5]	MLP + NSGA II	✓	✓	✓	MLP can get stuck in a local minima
[6]	IGSAGAW + CSSVM	✓	✓	✓	Computationally expensive
[7]	Ensemble SVM	✓	✓	✓	Computationally expensive
[8]	SVM	✓	✓	✓	High learning time
	KNN	x	x	x	Computationally expensive

4 Classification Model

The WBCD dataset presents a classification problem to diagnose breast cancer from fine-needle aspirates (FNA) images of potential cancerous cells; malignant or benign [3].

4.1 Experiment Design

The primary goal for the designed experiment is to develop a robust generalized model, i.e. the model performs as expected on future unseen data. Furthermore, achieving such model requires applying elegant state of the art techniques including regularization, meta-learning, and feature engineering. Note the focus is not on manipulating a specific machine learning model, however, decision trees were applied as a prototype algorithm within the model architecture [9].

Figure 1 presents the experiment design to be followed for breast cancer classification. The dataset is split into training, validation, and test sets. The training and validation are used to perform feature engineering along with training a classification model (decision trees) with the presence of meta-learning approach.

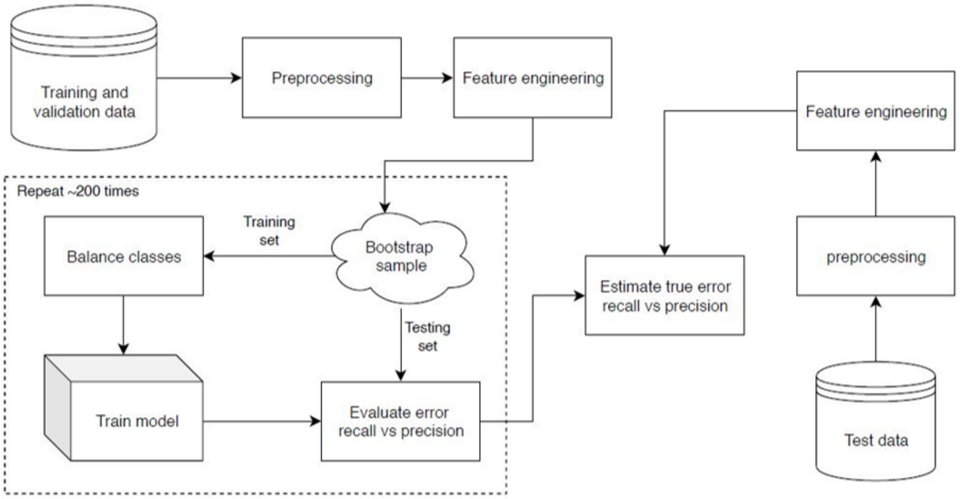


Figure 1: Experiment design for breast cancer diagnosis.

Due to class imbalance as described in Section 4.2, recall vs precision performance matrix is used to emphasis on correctly classifying malignant cells (true positive) with high precision, thus increasing precision at high recall percentage. Finally, bootstrapping is performed to estimate true error with confidence intervals ($\mu \pm \sigma$).

4.2 Dataset description

The dataset contains 30 features computed from a digitized image of an FNA of a breast mass. Specifically, features describe characteristics of the cell nuclei shown in images including radius, texture, perimeter, area, smoothness, and compactness. The class distribution of the obtained dataset is 357 benign samples and 212 malignant samples, hence, moderate class imbalance.

4.3 Preprocessing

Outliers are detected using 1.5 inter-quartile (IQ) rule. The 1.5 IQ rule is applied around the median since the median is more robust to extreme outliers. Following outlier detection, detected outliers are replaced with the median, and finally, features are standardized.

4.4 Feature Engineering

Optimal features give the best precision at recall 50% with the optimal tree depth with respect to overfitting precautions. Figure 2 illustrates an exhaustive methodology for optimal feature selection.

Unsupervised feature selection methods are applied to processed data

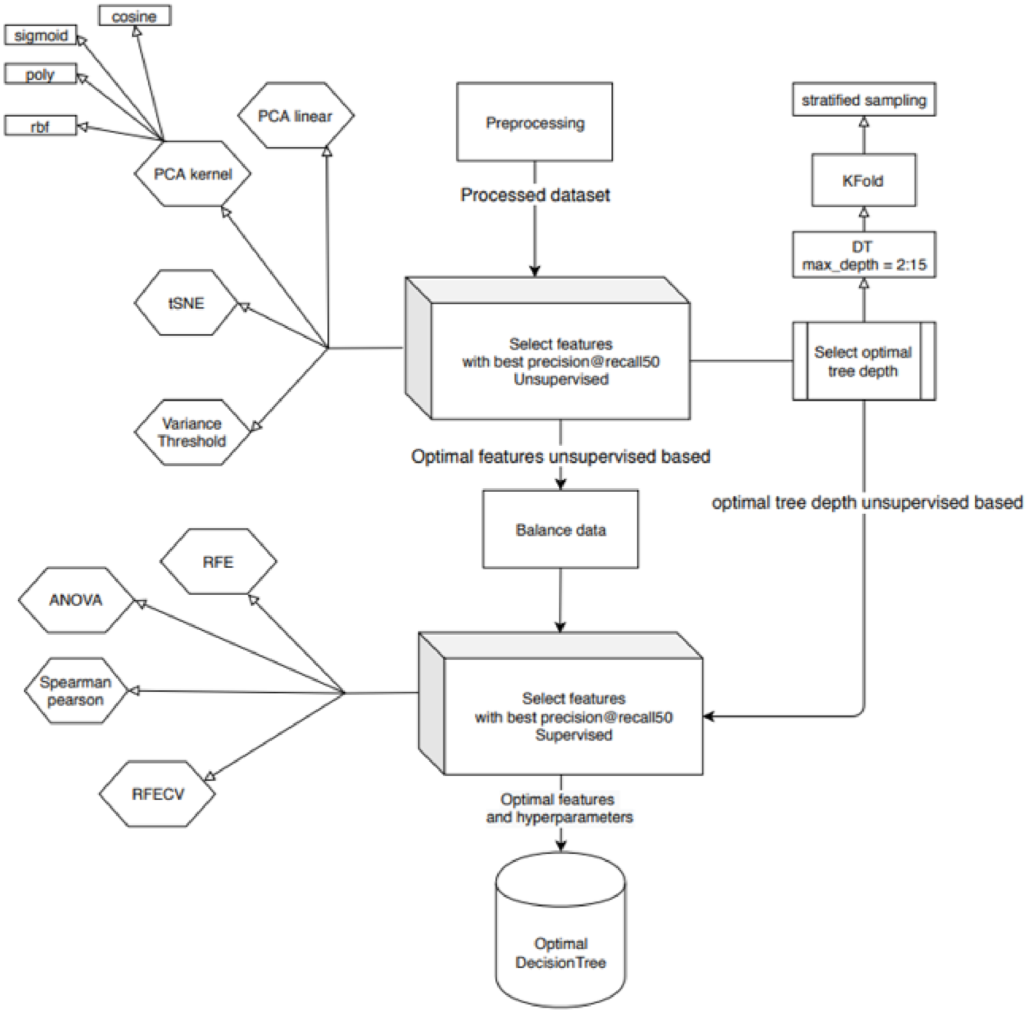

cosine
sigmoid
poly
rbf
PCA kernel
PCA linear
tSNE
Variance Threshold
Preprocessing
Processed dataset
Select features with best precision@recall50 Unsupervised
stratified sampling
KFold
DT max_depth = 2:15
Select optimal tree depth
Optimal features unsupervised based
optimal tree depth unsupervised based
Balance data
RFE
ANOVA
Spearman pearson
RFECV
Select features with best precision@recall50 Supervised
Optimal features and hyperparameters
Optimal DecisionTree

Figure 2: Optimal feature selection design.

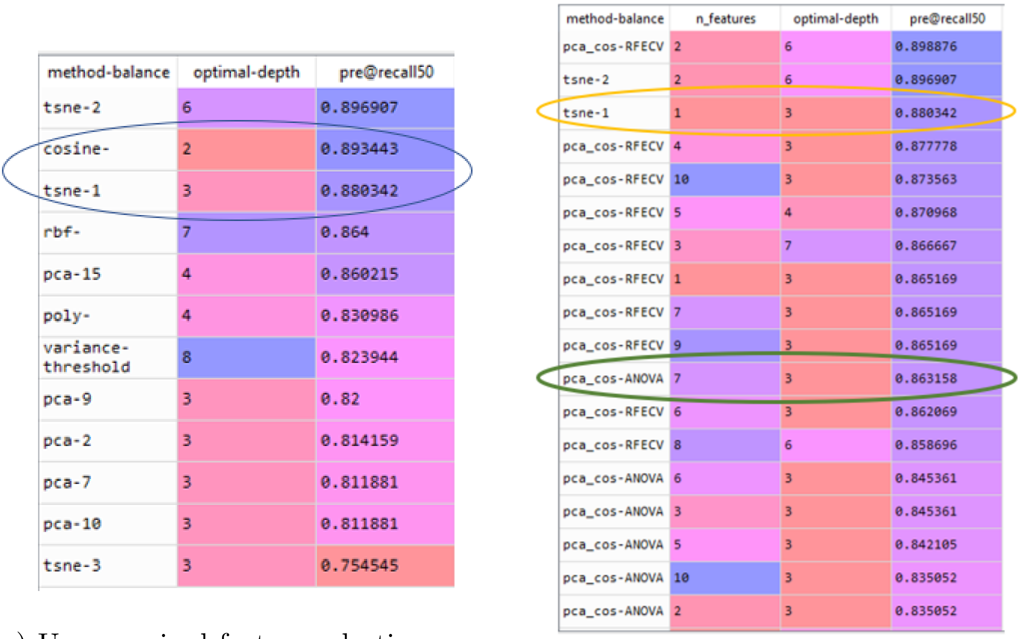
with the evaluation of precision at recall 50% at different tree depth for different methods. Afterwards, the data are balanced using weight-balancing and supervised methods are applied similarly to unsupervised methods.

According to Figure 3, the optimal features seem to be obtained using PCA with a cosine kernel (PCA-cos) at tree depth of six with precision of

89.7% at recall 50%, and after using Recursive Feature Elimination with Cross Validation (RFECV) selecting best 5 features of PCA-cos.

However, the tree depth of six and two features are concerning and there is a significant chance of overfitting. A robust model shall learn the data instead of memorizing patterns, specially given the limited dataset we have access to during the time of the project. Thus, we apply regularization and good engineering decisions, where we choose to move forward with low tree depths and avoid supervised feature selection algorithms with cross-validation techniques. Subsequently, from Figure 3a and 3b, the optimal features are obtained using PCA-cos and Analysis of Variance (ANOVA) method. That is, best 7 features of PCA-cos; features 28, 8, 27, 7, 6, 3, and 23, with a tree depth of 3 and a precision of 86.3% at recall 50%.

Finally, here comes the beauty of engineering, data preprocessing and optimal feature engineering are fully automated using custom-made libraries with only 9 lines of codes! (See Figure 4)



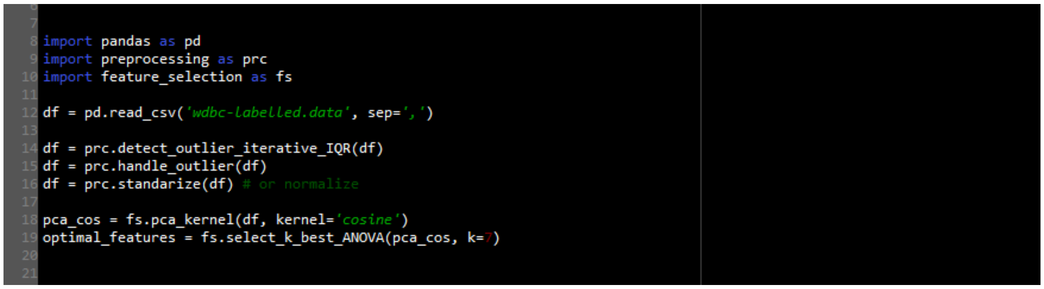
method-balance	optimal-depth	pre@recall50
tsne-2	6	0.896907
cosine-	2	0.893443
tsne-1	3	0.880342
rbf-	7	0.864
pca-15	4	0.860215
poly-	4	0.830986
variance-threshold	8	0.823944
pca-9	3	0.82
pca-2	3	0.814159
pca-7	3	0.811881
pca-10	3	0.811881
tsne-3	3	0.754545

(a) Unsupervised feature selection performance.

method-balance	n_features	optimal-depth	pre@recall50
pca_cos-RFECV	2	6	0.898876
tsne-2	2	6	0.896907
tsne-1	1	3	0.880342
pca_cos-RFECV	4	3	0.877778
pca_cos-RFECV	10	3	0.873563
pca_cos-RFECV	5	4	0.870968
pca_cos-RFECV	3	7	0.866667
pca_cos-RFECV	1	3	0.865169
pca_cos-RFECV	7	3	0.865169
pca_cos-RFECV	9	3	0.865169
pca_cos-ANOVA	7	3	0.863158
pca_cos-RFECV	6	3	0.862069
pca_cos-RFECV	8	6	0.858696
pca_cos-ANOVA	6	3	0.845361
pca_cos-ANOVA	3	3	0.845361
pca_cos-ANOVA	5	3	0.842105
pca_cos-ANOVA	10	3	0.835052
pca_cos-ANOVA	2	3	0.835052

(b) Supervised feature selection performance.

Figure 3: Feature selection performance summary.



```
import pandas as pd
import preprocessing as prc
import feature_selection as fs

df = pd.read_csv('wdbc-labelled.data', sep=',')

df = prc.detect_outlier_iterative_IQR(df)
df = prc.handle_outlier(df)
df = prc.standarize(df) # or normalize

pca_cos = fs.pca_kernel(df, kernel='cosine')
optimal_features = fs.select_k_best_ANOVA(pca_cos, k=7)
```

Figure 4: Summary of preprocessing and optimal feature engineering.

4.5 Training

In this project, we used a decision tree as a regular classifier and AdaBoost as the meta learning algorithm. This section discusses the approach to training the model.

4.5.1 Decision Trees

Decision trees are known to be brittle classifiers; therefore, it is important to perform regularization for them to generalize. SciKit-Learn library provides several ways of regularizing the decision trees, such as limiting the depth of the tree, the number of leaves, the samples to split, the samples per leaf, or the minimum impurity gain per split. These regularization methods are tested and limiting the depth of the tree is the simplest method in addition to performing equally well, and given the duration of the project, limiting the depth of the tree is the regularization method used.

4.5.2 Adaptive Boosted Decision Trees

Since AdaBoost fits extremely well with a decision tree classifier, the main hyperparameter to be determined was the number of decision trees to boost, as too many decision trees would lead to overfitting of the data. Given a depth of three is the optimal decision tree depth for feature engineering in Section 4.4, it appears to follow that the AdaBoost meta-learning algorithm should be applied to the same style of tree.

However, further investigation shows that a depth of one produced extremely good results. This was reinforced by further research into the underlying theory behind AdaBoost. Since AdaBoost works best on weak learners (i.e., a decision tree with only one depth, called a decision stump), it is extremely advantageous (and simple) to limit the decision tree such that it would be a weak learner.

Once the decision trees were limited to a depth of one, the optimal number of decision trees is determined through iterations until the complete training data fits without any error or until reached to a specified maximum number of estimators.

4.5.3 Testing

A holdout test is used to estimate the true error where a 20% subset of the dataset is reserved for holdout testing. The trained models are then applied to the holdout test data.

A bootstrapping test of 100 iterations was applied to the 80% of the data not withheld for the holdout test. The models are each retrained 100 times on a 50% stratified data split with replacement. The precision at recall greater than 50% is calculated at each iteration. At completion of the bootstrap test, the mean and standard deviation of the precision value are calculated.

Figure 5 shows a performance summary of the Adaboost decision tree classifier. The bootstrapping estimate is 0.9251 with a standard deviation of 0.0479 (0.9251 ± 0.0479). 0.632 bootstrapping estimate gives the upper bound performance of a model, but as shown, the holdout test surpasses it at a value of 0.9412. That is because of the skew in model performance distribution due to data limitation. Having a model trained on a sufficient amount of data gives a normal distribution performance. In other words, the model performance is deterministic around the mean.

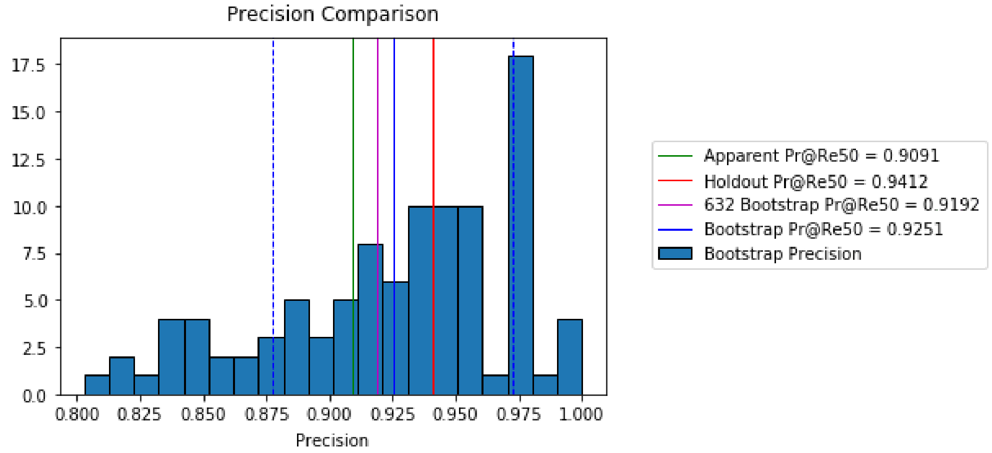


Figure 5: Adaboost decision tree performance.

5 Conclusion

A robust generalized classification model architecture, that contributes to the classification problem of breast cancer diagnose and is independent of a machine learning algorithm, is developed by applying state of the art techniques including prepossessing, optimal feature engineering, meta learning, thorough performance analysis, and good engineering decisions. In general, data are a concern in the machine learning field. As shown in this project, data limitation is a substantial obstacle specially with decision trees. For such an issue, merging support vector machine with the classification model architecture is worth looking into.

References

[1] C. C. S. A. Committee, “Canadian cancer statistics 2019,” 2019. [Online]. Available: https://www.cancer.ca/en/cancer-information/cancer-type/breast/statistics/?region=on

[2] V. Chaurasia and S. Pal, “A novel approach for breast cancer detection using data mining techniques,” *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 3297, pp. 2320–9801, 01 2014.

[3] D. Dua and C. Graff, “UCI machine learning repository,” 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[4] A. Bhardwaj and A. Tiwari, “Breast cancer diagnosis using genetically optimized neural network model,” *Expert Systems with Applications*, vol. 42, 02 2015.

[5] A. Osman and S. M. Shamsuddin, “Intelligent breast cancer diagnosis based on enhanced pareto optimal and multilayer perceptron neural network,” *International Journal of Computer Aided Engineering and Technology*, vol. 10, p. 543, 01 2018.

[6] N. Liu, E.-S. Qi, M. Xu, B. Gao, and G.-Q. Liu, “A novel intelligent classification model for breast cancer diagnosis,” *Information Processing Management*, vol. 56, no. 3, pp. 609 – 623, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0306457317307525

[7] H. Wang, B. Zheng, S. W. Yoon, and H. S. Ko, "A support vector machine-based ensemble algorithm for breast cancer diagnosis," *European Journal of Operational Research*, vol. 267, no. 2, pp. 687 – 699, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0377221717310810

[8] H. Asri, H. Mousannif, H. A. Moatassime, and T. Noel, "Using machine learning algorithms for breast cancer risk prediction and diagnosis," *Procedia Computer Science*, vol. 83, pp. 1064 – 1069, 2016, the 7th International Conference on Ambient Systems, Networks and Technologies (ANT 2016) / The 6th International Conference on Sustainable Energy Information Technology (SEIT-2016) / Affiliated Workshops. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877050916302575

[9] M. Hozaeyn and F. Foroutan, "A robust model for diagnostic breast cancer," Apr 2020. [Online]. Available: https://github.com/mohamedhozayen/DATA5000-Project