SVM & SVM Ensembles in Breast Cancer Prediction

Wrik Bhadra
M.Tech. CSE

Dhruv Kaushik
M.Tech. CSE

Subhani Shaik
M.Tech. CSE

IIIT Delhi, New Delhi 110020, India

Introduction

Breast cancer is one of the most commonly found cancer in women. There are various risk factors associated with it - age, obesity, lack of physical exercise, drinking alcohol, hormonal factors, menstruation etc. One big problem with breast cancer is that it is very difficult to identify cancer tumour in MRI (Magnetic Resonance Imaging) of breast in early stages. Therefore, there is need for intelligent systems which can predict possibility of developing breast cancer in a female patient, by taking into account above mentioned factors as features.

Objective

Min-Wei Huang et. al. [1] worked in this field by developing single SVMs with linear, RBF and polynomial kernels, along with SVM ensembles using bagging and boosting techniques for breast cancer prediction. In this project, our first aim is to reproduce the work done by them and compare our reproduced results with theirs. The work related to this aim is shown in the Part-1 of this report. The second aim is to identify shortcoming in their work and make improvements to overcome them. The work related to second aim is shown further in Part-2. Sample instructions to run our dockerized code is available in Part-3.

Methodology

A. Datasets

There are two datasets used: Breast Cancer Wisconsin dataset [2] and ACM SIGKDD Cup 2008 Breast Cancer challenge dataset [3]. Both are publicly available datasets for cancer prediction. Cancer Wisconsin dataset has 569 data samples and 32 different features. Though Cancer Wisconsin dataset (also referred as Small Dataset in this report) is widely used, it is very small to be used for complex machine learning techniques like ensemble models or deep learning and there is high chance of underfitting. Thus, results were also computed on the bigger ACM SIGKDD Cup 2008 dataset (also referred as Large Dataset in this report), so that results could be cross-verified. ACM SIGKDD Cup dataset has total 102294 data samples and 118 different features.

We have randomly sampled 5000 data points from large dataset. Among these 623 are of positive class (having breast cancer) and remaining 4377 are of negative class (not having breast cancer). In small data set, there are 357 negative class data points and 212 positive class data points.

B. Data Pre-processing

Min-Wei Huang et. al. performed feature selection as a part of data preprocessing. They used Genetic Algorithm implemented in Weka for feature selection. They reduced features to 10 in small dataset and 36 in large dataset by feature selection.

In our project, we have used Chi-Square test for feature selection using the scikit-learn Python library. After applying feature selection, number of features were optimally reduced to 14 in small dataset and 36 in large dataset. The data is normalized using Min-Max scaler normalizer in case of feature selection and using Scaler in case of no feature selection.

C. Classifier Design

In both the datasets, 80% of the data was taken as training data and remaining as testing data. Min-Wei Huang et. al. used Weka Library for developing SVM model. While training any SVM model, they performed 10-fold cross-validation and used the default values of parameters of SVM provided in the Weka library. They created total 18 SVM models. For each of the 2 datasets, these 9 models were developed:

a. Single SVM each for Linear, RBF and Polynomial kernel

b. Ensemble SVM, using Bagging technique (Bootstrap method), each for Linear, RBF and Polynomial kernel.

c. Ensemble SVM, using Boosting technique (Adaboost method), each for Linear, RBF and Polynomial kernel.

In our project, we have used scikit-learn to develop SVM model. We also trained the 18 SVM models mentioned above. Parameters were tuned for each model by performing 5-fold cross-validation while training. We used GridSearchCV in scikit-learn for parameter tuning.

Part-1

Results comparison

Small dataset

Results on Test data **without** Feature Selection:

	Accuracy	F1-Score	AUROC	MCC
Linear SVM	0.9825	0.9796	0.9988	0.9648
	0.625	0.58	0.50	NA
Polynomial SVM	0.9825	0.9796	1.0	0.9648
	0.57	0.58	0.51	NA
RBF SVM	0.9825	0.9796	1.0	0.9648
	0.66	0.52	0.48	NA
Bagging LinearSVM	0.9649	0.9583	0.9975	0.9305
	0.65	0.52	0.50	NA
Bagging Poly SVM	0.9474	0.9362	0.9975	0.897
	0.65	0.58	0.59	NA
Bagging RBF SVM	0.9474	0.9388	0.9938	0.8932
	0.66	0.53	0.50	NA
Boosting Linear SVM	0.9825	0.9796	0.9988	0.9648
	0.64	0.60	0.58	NA
Boosting Poly SVM	0.9298	0.913	1.0	0.8641
	0.63	0.59	0.58	NA
Boosting RBF SVM	0.9123	0.8889	1.0	0.8318
	0.65	0.57	0.58	NA

Legend

Original Paper	
Our work	

Small dataset

Results on Test data **with** Feature Selection:

	Accuracy	F1-Score	AUROC	MCC
Linear SVM	0.9649	0.959	0.9962	0.9288
	0.97	0.94	0.97	NA
Polynomial SVM	0.9474	0.9412	0.9862	0.8942
	0.94	0.91	0.91	NA
RBF SVM	0.9649	0.96	0.9975	0.9288
	0.96	0.95	0.95	NA
Bagging Linear SVM	0.9474	0.9388	0.9975	0.8932
	0.97	0.95	0.98	NA
Bagging Poly SVM	0.9474	0.9388	0.9962	0.8932
	0.95	0.93	0.98	NA
Bagging RBF SVM	0.9474	0.9388	0.9975	0.8932
	0.96	0.94	0.98	NA
Boosting Linear SVM	0.9649	0.96	0.9962	0.9288
	0.96	0.96	0.98	NA
Boosting Poly SVM	0.9474	0.9362	0.9962	0.897
	0.94	0.94	0.98	NA
Boosting RBF SVM	0.9649	0.96	0.9962	0.9288
	0.97	0.95	0.98	NA

Legend

Original Paper	
Our work	

Large Dataset

Results on Test data **without** Feature Selection:

	Accuracy	F1-Score	AUROC	MCC
Linear SVM	0.9	0.59	**0.8933**	0.5469
	0.994	0.991	0.525	NA
Polynomial SVM	0.9	0.5536	0.8814	0.53
	0.995	0.994	0.62	NA
RBF SVM	0.916	0.625	0.8732	0.614
	0.9948	0.993	0.57	NA
Bagging Linear SVM	0.912	0.6333	0.8143	0.6012
	0.994	0.992	0.6	NA
Bagging Poly SVM	0.902	0.5505	0.7216	0.5369
	0.995	0.994	0.65	NA
Bagging RBF SVM	0.916	0.6182	0.755	0.613
	0.995	0.993	0.625	NA
Boosting Linear SVM	0.92	0.775	0.83	0.75
	0.993	0.92	0.85	NA
Boosting Poly SVM	**0.9298**	0.71	0.75	0.7
	0.992	0.92	0.84	NA
Boosting RBF SVM	0.92	**0.77**	0.78	0.74
	0.995	0.94	0.86	NA

Legend

Original Paper

Our work

Large Dataset

Results on Test data **with** Feature Selection:

	Accuracy	F1-Score	AUROC	MCC
Linear SVM	**0.914**	**0.6387**	0.8784	0.6097
	0.9938	0.991	0.5	NA
Polynomial SVM	0.906	0.5983	0.883	0.5677
	0.995	0.993	0.6	NA
RBF SVM	0.904	0.5932	**0.8825**	0.5593
	0.9948	0.993	0.54	NA
Bagging Linear SVM	0.908	0.54	0.6973	0.5691
	0.993	0.991	0.5	NA
Bagging Poly SVM	0.864	0.1053	0.5417	0.2189
	0.995	0.994	0.6	NA
Bagging RBF SVM	0.896	0.4468	0.6516	0.4953
	0.994	0.993	0.57	NA
Boosting Linear SVM	0.856	0	0.8154	0
	0.994	0.99	0.85	NA
Boosting Poly SVM	0.886	0.3596	0.79	0.426
	0.991	0.92	0.84	NA
Boosting RBF SVM	0.858	0.0274	0.8362	0.101
	0.994	0.94	0.85	NA

Legend

Original Paper	
Our work	

Best performing models for small dataset are tabulated below:

Small Dataset			Feature Selection
Classification Accuracy	Paper	RBF SVM with Boosting (0.98)	FS
	Our	**Linear SVM with Boosting (0.98225)**	NFS
ROC	Paper	linear/poly SVM with Bagging (0.98)	FS
	Our	**RBF SVM with Boosting (1.0)**	NFS
F1 Measure	Paper	RBF SVM (0.988)	FS
	Our	**Linear SVM (0.9796)**	NFS

Best performing models for large dataset are tabulated below:

Large Dataset			Feature Selection
Classification Accuracy	Paper	RBF SVM with Boosting (0.9952)	NFS
	Our	**Boosting Poly SVM (0.93)**	NFS
ROC	Paper	Linear SVM with Boosting (0.876)	FS
	Our	**Linear SVM (0.8933)**	NFS
F1 Measure	Paper	RBF SVM with Boosting (0.995)	NFS
	Our	**Boosting RBF SVM**	NFS

Inference

1. We observed that despite following the same pipeline, model performance for small dataset after feature selection is usually much better than that without feature selection. There is a possibility of underfitting in the small dataset leading to such high performance.
2. There is a significant class imbalance in both the datasets. This could be an issue to authenticity of the results obtained.
3. There is no single SVM model that is giving best values on all the performance measures. Different models perform best wrt different performance measures.
4. In small dataset, we found that linear kernel performs best (Accuracy and F-Score) and Boosting gives better results than single SVM.
5. On large dataset, Boosting usually gives better results than single SVM. In terms of accuracy, Poly SVM performs best.

Part-2

Objective

First objective was to deal with class imbalance. Second was to check performance in deep learning model.

Our Contribution

One major problem that we realized was the class imbalance in dataset. To deal with this, we decided to test performance on a model that can implicitly handle class imbalance. One such model we identified after research was NuSVM and checked the performance of NuSVM over the two datasets.

Also, there is huge amount of data in the large dataset. Therefore, we decided to test its performance in deep learning model as the data seems sufficient for it (very less probability of overfitting). We checked performance of three-layer ANN over Large dataset.

Results

NuSVM results for small dataset are tabulated below:

NuSVM - Small Dataset				
	Accuracy	F1-Score	AUROC	MCC
Linear NuSVM	0.9824	0.9756	0.9963	0.9621
Polynomial NuSVM	0.9883	0.9837	0.9998	0.9749
RBF NuSVM	0.9824	0.9756	0.9964	0.9621
Bagging Linear NuSVM	0.9531	0.9314	0.9879	0.901
Bagging Poly NuSVM	0.9668	0.9539	0.993	0.9282
Bagging RBF NuSVM	0.9297	0.8989	0.9817	0.8477

NuSVM results for large dataset are tabulated below:

NuSVM - Large Dataset without FS				
	Accuracy	F1-Score	AUROC	MCC
Linear NuSVM	0.4253	0.18	0.578	0.1058
Polynomial NuSVM	0.9649	0.8326	0.9983	0.8281
RBF NuSVM	0.9958	0.9829	0.999	0.9805

ANN results for large dataset are tabulated below:

ANN - Large Dataset				
	Accuracy	F1-Score	AUROC	MCC
All Points	1.0	0.9982	1.0	0.99
Sampled Points	1.0	1.0	1.0	1.0

Inference

1. We found that the performances of NuSVM for small dataset is almost same as for normal SVM over the same data set
2. The performance of NuSVM over the subset of large dataset (5000 sampled points) is much better than that for normal SVM over the same data set
3. So we conclude that the class imbalance can be handled using NuSVM
4. Ann gives best performance over the large data set without sampling as well as over the sampled subset of large data set.
5. This shows that there is a possibility that by using deep learning model we can achieve better performance than classical machine learning technique for the breast cancer prediction.

Part-3

Docker Image

Sample sequence of instructions to follow in order to execute our code is given below. The docker image is hosted on DockerHub.

- docker pull subhani007/mlba_project:latest
- docker run --name=project -itd subhani007/mlba_project
- docker exec -it project /bin/bash
- python3 source/small_nusvm.py

The remaining python programs can be run using a similar command.

References

[1] Huang MW, Chen CW, Lin WC, Ke SW, Tsai CF (2017). SVM and SVM Ensembles in Breast Cancer Prediction. PLOS ONE 12(1): e0161501. https://doi.org/10.1371/journal.pone.0161501

[2] Breast Cancer Wisconsin Original Dataset (1992). UCI Machine Learning Repo. Internet: https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Original)

[3] ACM SIGKDD Cup 2008 Breast Cancer challenge (2008). Internet: https://kdd.org/kdd-cup/view/kdd-cup-2008/Data