BIM3008-Assignment5 Report

Junyang Deng (120090791)

January 13, 2023

1 Introduction

Breast cancer is the most common type of cancer for females. One important task for researchers to improve the survival rate of BRCA patients is to identify the cancer stage and apply different treatment strategies. We can train a model to classify cancer stages using multi-omics data collected from patient samples. Multi-omics data includes gene expression data (RNA-seq), miRNA expression data and DNA methylation data.

In this project, important features of RNA-seq, methylation and miRNA-seq are first selected by differentially expressed analysis. Then, models that are based on different data sources are built, and cross-validation is used to find the best parameters. Next, a scoring function is created to combine all the models. The scoring function combines all the expected results with different weights and decide the final predicted label. The scoring function is also optimized by cross-validation. Finally, the overall model performance is evaluated.

2 Procedure

2.1 Data Retrieval

Data required for this project is retrieved from the UCSC Xena database (cohort: GDC TCGA Breast Cancer (BRCA)). We used

- unnormalized gene expression RNA-seq data (HTSeq - Counts) `TCGA-BRCA.htse_counts.tsv`
- DNA methylation (Illumina Human Methylation 450) `TCGA-BRCA.methylation450.tsv`
- miRNA-seq data (stem loop expression - miRNA Expression Quantification) `TCGA-BRCA.mirna.tsv`
- Phenotype labels (phenotype) `TCGA-BRCA.GDC_phenotype.tsv`

2.2 Data Preprocessing

Phenotype label

The phenotype table is a 1284×140 matrix. Each row represents a sample and each column represents a phenotype label. Only *sample IDs* and *cancer stage diagnoses* are used in this project, so the rest of columns are removed. To make the classification task more realistic, we combined the substages of cancer. Originally, there are 13 diagnoses labels (including `not reported`). After combinding the substages (e.g. merge stage ia and ib into stage i), only 5 labels remain. I further discard conditions whose sample ID is not included in the counts matrix. The final number of samples in each stage is listed in Table 1.

Label	stage i	stage ii	stage iii	Sum
Number	202	693	275	1,170

Table 1: Cancer stage labels

RNA-seq data

Original read counts data is a 60483 × 1218 matrix, where each row is a gene and each column is a sample. Since the unit of data in the database is log2(count+1), I reverse the log2 back to natural numbers. Then, the DESeq2 library in R is used to calculate the normalized counts matrix.

```
library(DESeq2)
dds <- DESeqDataSetFromMatrix(countData = counts,
                              colData = phenotype,
                              design = ~diagnoses)
normalized_counts <- counts(dds, normalized=TRUE)
write.table(normalized_counts, # export for further analysis
            file="normalized_counts.tsv", sep="\t",
            quote=F, col.names=NA)
```

DNA methylation data

The original data is a 485577 × 890 matrix. Each row represents a possible methylation site, and each column represents a sample. After removing all the methylation sites with NA and sites that are zero on all samples, the shape of matrix becomes 363791 × 890.

miRNA-seq data

The original data is a 1881 × 1202 matrix. After removing miRNA-seq data with zero on all samples, the matrix becomes 1604 × 1202. After processing four datasets, I used the merge function consecutively to combine all dataframes. Samples that exist in the last dataframe have information in all four datasets. The final dataframe is 825(samples) × 425879 (features).

2.3 Feature selection

To reduce noise and improve model performance, we can do feature selection to pick out important genes for the classification task. Differential analysis is used in this study.

RNA-seq data

Genes with have constant expression level among different samples not only could not benefit stage classification, but also create noise for the task. So I conducted differentially expressed gene analysis to find out important genes. The code below (which follows the R code above) is the analysis for stage i vs ii. This procedure is repeated for stage ii vs iii, and stage i vs iii. A total of 13 genes are found in the intersection between three groups of DEGs.

```
dds <- DESeq(dds, quiet = FALSE)
res <- results(dds, contrast = c("diagnoses", "stage␣i", "stage␣ii"))
write.csv(res, "DEG_1vs2.csv")
```

However, when I used PCA to do dimensionality reduction on the data, samples with different stages cannot be separated well (Figure 1). This indicates that the accuracy for predict will not be good either.

Another finding is that the homogeneity within stages is not high. When LDA is conducted on the whole dataset, as shown in Figure 3, three groups can be clearly separated. However, when LDA is conducted on train set and applied on the test set, groups in test set cannot be separated (Figure 4 and 5).

DNA methylation data

I planned to do the differential analysis on DNA methylation data. However, the file was too large (5.1G) and I failed to do the analysis on my computer. Therefore, I used chi-square to select 500 methylation data with lowest p-value. The UMAP embeddings figure of these 500 methylation label are shown in Figure 2. Then, I also used LDA to do dimensionality reduction. The result in DNA methylation data is similar to RNA-seq data. Groups can be separated when they're used in the training procedure. When applying the model on the test set, groups cannot be separated (Figure 6 and 7).

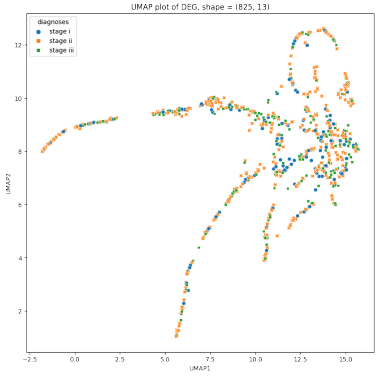

Figure 1: UMAP plot of DEGs

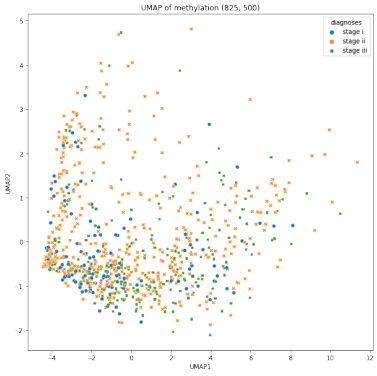

Figure 2: UMAP plot of DNA methylation

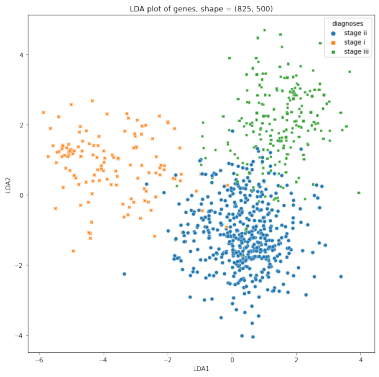

Figure 3: Conduct LDA on the whole dataset

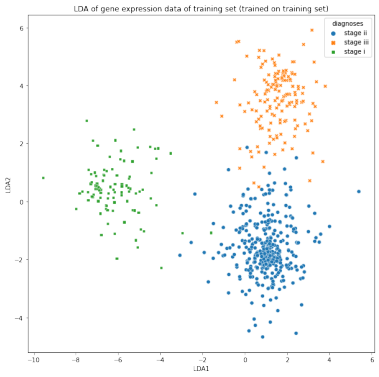

Figure 4: Train LDA on training set and plot the training samples

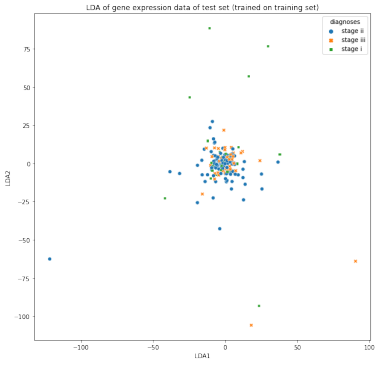

Figure 5: Train LDA on training set and plot the test samples

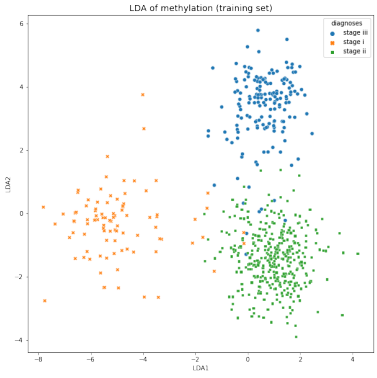

Figure 6: Conduct LDA on methylation training dataset

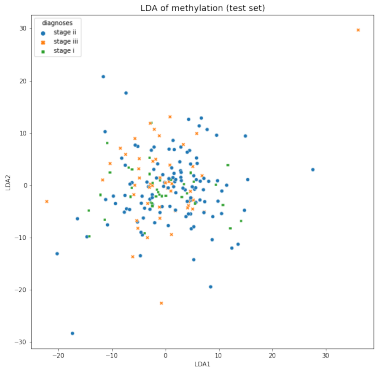

Figure 7: Conduct LDA on methylation training dataset and plot the test data

miRNA-seq data

I used chi-square to select important miRNA for stage classification. 30 miRNA are chosen based on the chi-square value. I also conducted LDA analysis on miRNA data. The result is nearly the same as above (RNA-seq and DNA methylation). And it is omitted here.

2.4 Model selection and cross-validation

The next task is to select the best hyperparameter using cross-validation. The `GridSearchCV` class is used here to perform this task. After 5-fold cross-validation, the best parameters for svm classifiers for all three datasets are listed in Table 2.

Table 2: Best parameters

Algorithm	Best parameters	Score
SVM	`'gamma': 'auto', 'kernel': 'linear', 'C': 0.1`	0.5936

2.5 Scoring function and ensemble model

Finally, a scoring function is used to summarize the result of three models. The scoring function is defined as:

$$y = \lambda_1 \times y_1 + \lambda_2 \times y_2 + \lambda_3 \times y_3$$

where y_1, y_2 and y_3 are 3×1 the results of three models. y_i is a 3 by 1 vector $[y_{i1}, y_{i2}, y_{i3}]$. y_{ij} represents the probability of a sample belonging to stage j (predicted by model i). And there exists a relationship described in Equation (1).

$$y_{i1} + y_{i2} + y_{i3} = 1 \tag{1}$$

Since none of the model can give reasonable predictions, their combinations are not satisfactory either. The result of ensemble model is shown in Table 3.

Table 3: Performance of ensemble models

Weights	Accuracy
(0.1, 0.3, 0.6)	0.5266
(0.1, 0.5, 0.4)	0.5266
(0.2, 0.4, 0.4)	0.5266
(0.2, 0.6, 0.2)	0.5266
(0.3, 0.3, 0.4)	0.5459
(0.3, 0.5, 0.2)	0.5459
(0.4, 0.2, 0.4)	0.5266
(0.5, 0.1, 0.4)	0.5266

2.6 Result

In this study, three data sources are used to predict the stage of cancer. After preprocessing, small models are first trained on dataset from different sources, then a scoring function is used to combine the results. The result is not good, probability because the heoterogeneity within stages is too high and not distinguishable.