Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: 16 December 2019
Project Report

OXFORD

Expression Analysis

Preterm Birth Prediction Using Microarray

Tianmin Zhang [1]*, Yuting Wang [1]*, Zhichao Zhang [1]*, Jongwon Im [2]* and Ruihua Tian [1]*

[1]Electrical Engineering, Columbia University,
[2]Biomedical Engineering, Columbia University.

*To whom correspondence should be addressed.

Abstract

Motivation: In obstetrics, preterm birth (PTB) is giving birth before 37 weeks of gestation, which leads to short-term complications including breathing and heart problems and long-term complications such as behavioral, vision and hearing impairment. Current prediction of spontaneous preterm birth involves the length of cervix and preterm birth history. We tried to determine the risk of preterm birth by developing prediction models given whole blood gene expression data collected from pregnant women to see if whole blood transcriptome changes are related to preterm birth.
Results: We first conducted differential expression analysis in R to pick top genes, implemented binary classification and survival analysis, validated the predictions of Control vs. sPTD and Control vs. PPROM in Python. The submission scores, AUC, indicate that microarray data can be considered as a predictor. To explore improvements, we used sampling to balance the dataset, tried different classifiers, employed clustering methods, chose samples based on gestational age for comparison. Results show that for this problem, a classifier fed with a more balanced dataset or a survival model using the latest sample performs better. SVM outperforms Random Forest and MLP. Co-expression analysis by clustering makes little difference to the improvement of performance.
Availability and implementation: Scripts and data are available at GitHub.
Contact: tz2378@columbia.edu, yw3167@columbia.edu, zz2668@columbia.edu, ji2275@columbia.edu, and rt2751@columbia.edu
Supplementary information: Supplementary data are available at *DREAM Preterm Birth Prediction Challenge, Transcriptomics* online.

1 Introduction

Preterm birth is a birth that takes place at a gestational age earlier than 37 weeks and it is one of the most veiled health problems in the world, affecting 15 million babies every year, as a major contributor to newborn deaths and disabilities. Preterm birth leads to health problems for both the gravida and the infants. Symptoms can be uterine contractions which occur more often than every ten minutes or the leaking of fluid from the vagina. Premature infants are at greater risk for cerebral palsy, delays in development, hearing problems and sight problems. The earlier a baby is born, the greater these risks will be. However, it is unclear what caused over 50 percent of preterm births problems. Although there are a few factors that we know about, for instance, the length of cervix (Romero *et al.*, 2013), the most critical predictor of preterm birth is a previous preterm birth (Romero *et al.*, 2014), whether the woman has already had one herself, or the family has a history of preterm birth. This means that there are significant genomic indicators that have a considerable impact on this problem. For example, twin and family studies suggest that 30 to 40% of the variation in birth timing and in the risk of preterm birth arises from genetic factors that largely but not exclusively reside in the maternal genome (**?**).

From this project, we are trying to predict the preterm birth by looking at the maternal whole blood gene expression data and further understand the pathway that causes the preterm birth for future treatment.

2 Materials and Methods

The training dataset consists of microarray data of total 435 samples from 196 patients profiled using two platforms, Affymetrix HTA 2.0 and Affymetrix HG 2.1 ST Array, while the test set includes data of 304 samples from 87 women. The gene expression matrix data, of which each row are named with ENTREZ gene IDs, were preprocessed from raw .CEL files and are used here as predictors for the problem. For an individual, there are

© The Author 2019. Published by Oxford University Press. All rights reserved. For permissions, please e-mail: journals.permissions@oup.com

several samples. Each sample is annotated with significant information, such as GA (gestational age when sampled), GADel (gestational age at delivery) and Group (whether PTB or not, namely Control, sPTD and PPROM).

The goal is to make use of microarray data, classify Control v.s sPTD and Control v.s PPROM, and obtain an AUC as distinct to 0.5 (random prediction) as possible.

2.1 Exploratory Data Analysis

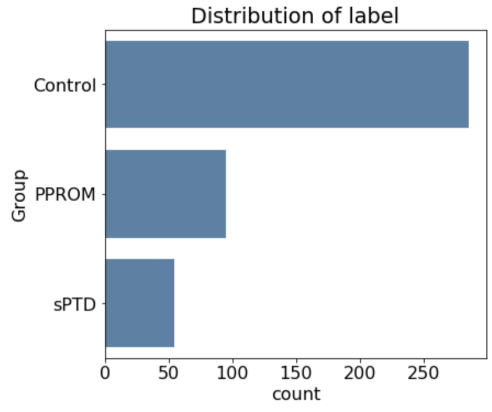


Fig. 1: Distribution of label

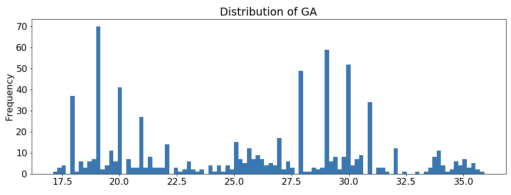


Fig. 2: Distribution of GA

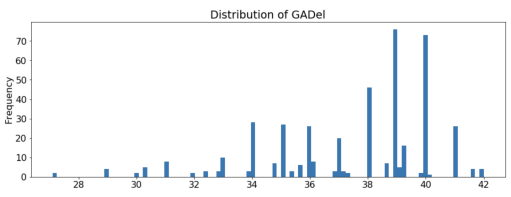


Fig. 3: Distribution of GADel

In this part, we visualized the distribution of the data set and tried to conclude some important attributes and give direction to our research. Firstly, we calculated the distribution of the labels. It could be found in Fig 1 that the distribution of labels is imbalanced, and the number of Control is greater than the sum of other two labels, PPROM and sPTD. Thus the sampling methods are required before training our models.

Another important visualization is the distribution of gestational age and gestational age at delivery in Fig 2 and Fig 3. For gestational age, there is an uneven distribution and most records exist around 17.5 to 21 and 27.5 to 31, but the gestational age at delivery, most records exist around 38 to 41, and there are some overlapping in 25 to 38.

Another analysis is cleaning the dataset and deleting the redundant parts. The dataset provides 739 samples from 283 individuals, so it would be intuitive to select one of the samples for each individual to reduce the dependency between samples and individuals. Thus, we also extracted the samples with largest GA in each individual, which could represent a closer time point to predict the gestational condition for comparison.

2.2 Dealing With Imbalanced Data

There are 55 sPTD and 95 PPROM samples out of 435 available training samples in total. The ratio of Control to Non-control is approximately 5:1 and 3:1 respectively. Such a level of imbalance cannot be ignored directly. Under the circumstances, since a model is trained to minimize the error or maximize the accuracy, it will tend to classify test data to the majority class with imbalanced dataset fed. There are several ways to deal with imbalanced data, for instance, downsampling, oversampling, or adding class weights.

- By downsampling we split the training dataset into positive and negative samples, downsampled the majority class so that both classes have the same number of samples, and then concatenated them. Some useful information may be discarded during downsampling.
- By oversampling we similarly split the training dataset, oversampled minority class and merged samples of two classes. Oversampling methods include naïve random oversampling, SMOTE, ADASYN, etc. SMOTE, Synthetic Minority Over-sampling Technique (Chawla *et al.*, 2002), was employed here. It first finds K nearest neighbors of a data point, and generates a new one by randomly adding the difference between the point and some neighbors. Overfitting is possible to occur in the oversampling.
- By adding class weights the classifier will pay more attention to under-represented class. Usually in Python's sklearn the weights can be adjusted easily through the parameter 'class_weight' by either defining explicitly or setting 'balanced' mode.

2.3 Shuffling Data

If we directly feed the classifier with the sampled data and use stratified K fold cross validation, some folds may only contain samples of one class. In that case, the model cannot tell the difference between positive and negative samples. Therefore, shuffling is necessary.

2.4 Selecting Features

There are 29459 genes in the microarray while there are only 435 training samples. Overfitting occurs easily if all the genes are selected as features to fit the model. Therefore, we first did differential expression analysis in R to pick top 100 genes sorted by P value in descending order. Then co-expression analysis was conducted to reduce collinearity between genes.

- Differential expression analysis

  Because only microarray data, rather than count data in our homework, are available, we used the package limma (Ritchie *et al.*, 2015) instead. limma uses linear models to analyze designed microarray experiments. By specifying the design matrix to indicate to which group the samples belong, it can unveil genes which express extremely high or low compared to another group. An empirical Bayes method is used to provide stable inference, showing improved power for small-sized arrays in particular.
- Co-expression analysis

  Co-expression analysis is undertaken based on top 100 differentially expressed genes. Two clustering approaches, K-means

and hierarchical clustering, are used. After assigning each gene a cluster label and ignoring the cluster with few genes, an average 'metagene' was calculated for every cluster. These metagenes were passed as the final features to fit the classification or survival analysis model.

2.5 Classification

The classifiers we used are support vector machine, multi-layer perceptron and random forest. All three models are utilized from sklearn, which provides useful and straightforward API. Support vector machine is a classic supervised learning classifier, and could perform well in dealing with optimization problems with high dimensional features. Multi-layer perceptron is a neural network consisting of an input layer, an output layer, and one or more hidden layers. By backpropagation to adjust the weights, the model could be trained to classify the non-linearly separable labels. Random forest is a model based on decision tree, and it could reduce the degree of overfitting by fitting on randomly selected subsets and combining them together.

2.6 Survival Analysis

Since TTD (interval in weeks from sample to delivery) is provided in the annotation file, survival analysis can be applied to estimate the duration of an event, specifically, the delivery in the project. The concordance index is often used to measure how well a biomarker predicts time-to-event, therefore serving as a feature importance estimator in the project. The parameter 'Event' was set to be 'True' for all samples because all the patients have delivered their baby. Similar to the feature selection in classification, we calculated the concordance of index of each gene and selected top 100 genes sorted by absolute difference between it and 0.5 in descending order, generated metagenes by co-expression analysis, and used Xgboost as the model. Xgboost predicts risk scores while what we need is the probability of preterm birth. So risk scores were normalized to range from 0 to 1, interpreted as probabilities.

There are more than one sample for each patient. To find out if how early the sample is taken will make a difference, the training was undertaken in two directions, fed with the whole training set and the latest samples of patient, for comparison.

2.7 Evaluation

2.7.1 Cross validation

We used cross validation to test the model on the training set to avoid overfitting (model performs well to the training data but poor to the test data), and underfitting (model performs poor both on training and test data). It helps to choose the best model we can use on a new set of data.

With K-fold, We repeatedly held out and averaged scores after K different holdouts. Each data point would be in a validation set once, and be in a training dataset k-1 times. In this way, it would reduce underfitting because we use most of the data for fitting, and it would reduce overfitting because most of the data is used in the validation set.

2.7.2 Receiver Operating Characteristic with cross validation

We created K-fold cross-validation and showed the ROC response of each one of those. Receiver Operating Characteristic (ROC) is used to measure how the classifier performs with cross-validation. Y axis of ROC shows true positive rate ranging from 0.0 to 1.0, X axis shows false positive rate ranging from 0.0 to 1.0.

The area under the curve (AUC) can help to decide which model has a better prediction. The top left corner - a smaller false positive rate and a higher true positive rate can be an ideal curve. But not a larger AUC always guarantee a better result. When the ROC curve has a higher slope, which means a high true positive rate and a low false positive rate, is also a situation we are looking for. The Mean ROC curve helps to visualize how it varies for different held out dataset.

2.7.3 Area under Precision-Recall curves

Precision measures a model's performance when predicting a positive class. Recall does the same as sensitivity. Precision-Recall is helpful to measure the performance of model when the classes are very imbalanced. A high precision shows a small false positive rate and a high recall shows a small false negative rate. A high precision and high recall leads to a high area under the curve and a very accurate classifier. It is mostly used in binary classification to measure how well the output is. When using in multi-class classification, the output needs to be binarized.

3 Results

We implemented differential expression analysis using the package limma in R and saved top significant genes as .csv files for SPTD and PPROM respectively. We implemented the rest, such as classification and survival analysis, in Python using Google Colaboratory. To make full use of the training set, tune parameters and validate the model, we used stratified K Fold cross validation (K=6). Due to the variance in model training, there is a slight fluctuation in cross validation scores.

3.1 Cross Validation Results

3.1.1 Imbalanced vs. balanced dataset

We compared the performance of classification with different sampling methods, including without sampling, downsampling, oversampling, and adding class weights, to see if balancing data is helpful, all using SVM (linear) to classify and K-means (maximum clusters of 10) to cluster. As Fig 4a shows, the AUC increases slightly in Control vs. SPTD classification while it increases 0.12, 0.08, and 0.07 respectively in Control vs. PPROM classification on average, indicating with a more balanced dataset, the classifier performs better.

3.1.2 SVM (Linear) vs. Random Forest vs. MLP

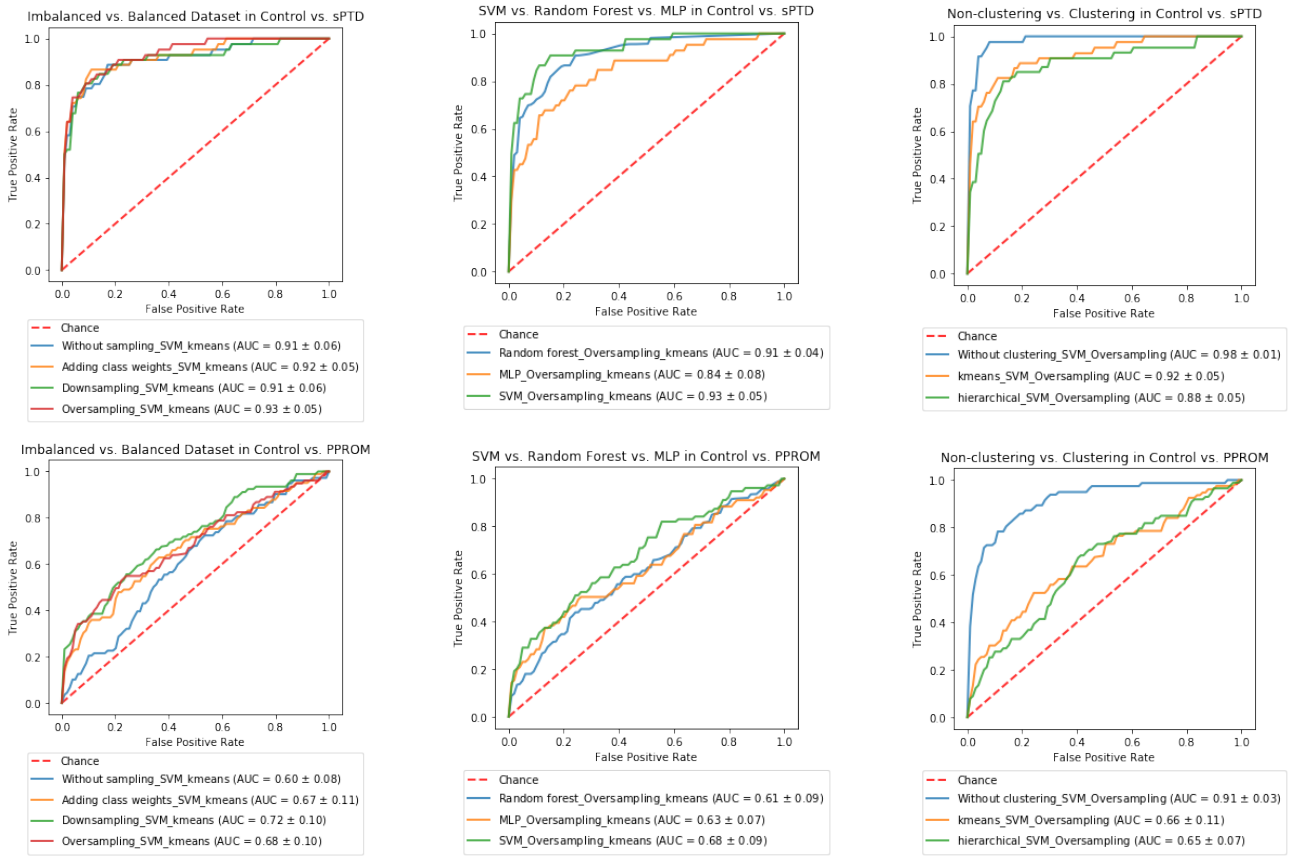
We also compared the effect of different classifiers. Using K-means (maximum clusters of 10) to cluster and oversampling, shown in Fig 4b, the average AUC in Control vs. SPTD is 0.93, 0.91 and 0.84 while in Control vs. PPROM it is 0.68, 0.61 and 0.63 for SVM, Random Forest and MLP respectively. Therefore, SVM outperforms the others for this problem.

3.1.3 Non-clustering vs. clustering

In order to evaluate whether co-expression analysis is beneficial, we ran the cross validation with different features - all the top 100 genes (either selected by P value of differential expression analysis for classification or by the concordance index for survival model), and clustering (with maximum clusters of 10, either by K-means or by hierarchical clustering, `hierarchical.euclidean.complete`). It is found from Fig 4c and Fig 5 that a model with full 100 features predict better that one with clustered genes in classification while there is no significant difference in survival model. One possible reason why using 100 genes as features works better in classification is that the model can work out more complex decision boundaries with a larger dimension of features.

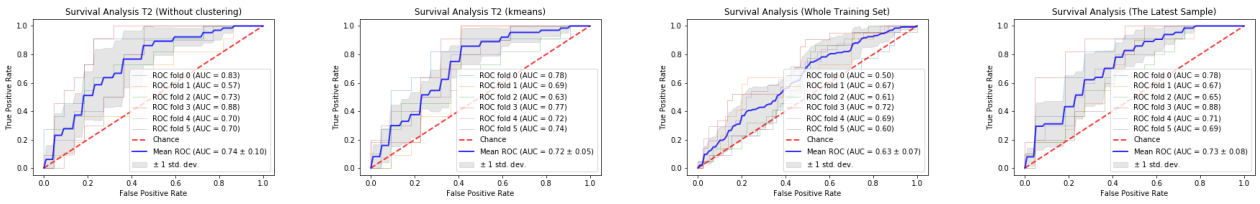
3.1.4 The whole training set vs. the latest sample

We also explored the relation between GA and survival model performance. A patient takes several samples before the delivery. As Fig 5 implies, the model performs better if only the latest samples of each patient



(a) Non-sampling vs. sampling (b) Comparing different classifiers (c) Non-clustering vs. clustering

Fig. 4: Cross validation results in classification



(a) Without clustering (b) Using K-means (c) Using the whole set (d) Using the latest sample

Fig. 5: Cross validation results in survival analysis

are taken into account than all the samples, indicating that the latest sample can be more representative of the expression status before the delivery.

3.2 Submission Scores in DREAM Challenge Leaderboard

We submitted three predictions to the leaderboard (Table 1). It is obvious to see that there is a gap between the cross validation scores and the submission. During the training and the cross validation, the AUC is usually around 0.9 for Control vs. SPTD, 0.6 for Control vs. PPROM in SVM with hierarchical clustering, and 0.7 in survival analysis with hierarchical clustering. There is a distinct drop in the submission scores. One reason is that when we calculated P values in differential expression analysis or concordance indices in survival analysis, the whole training dataset was used. Even though the data was later split into training and validation, the features already cover the information of the whole dataset.

Table 1. Submission Scores

Model	sPTD_AUC	sPTD_AUPR	RRPOM_AUC	PPROM_AUPR
Survival	0.544	0.4264	0.4465	0.4104
SVM	0.4907	0.4081	0.5309	0.4793
Survival+SVM	0.5255	0.5057	0.4835	0.494

Both models use clustered features (hierarchical.euclidean.complete).

Besides, according to the data description in the challenge, all the test samples are collected from the HTA 2.0 platform, if we choose samples from that platform to validate, the validation scores may be closer to the submission scores.

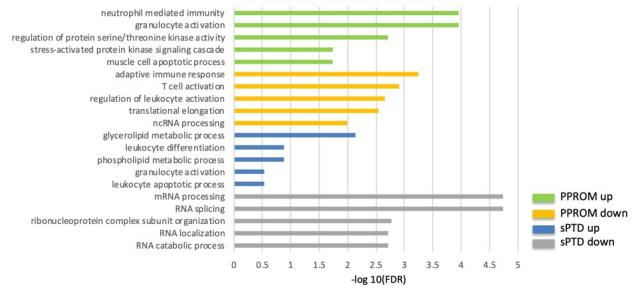


Fig. 6: Identification of biological processes associated with differentially expressed genes in whole blood samples from both sPTD and PPROM patients

3.3 Identification of biological processes that are relevant to preterm birth prediction

Gene ontology analysis (Liao *et al.*, 2019) with 1,000 differentially expressed genes in each sPTD and PPROM compared to control, revealed biological pathways that are relevant to the preterm birth (Fig 6). In PPROM patients, cell apoptotic pathways were up-regulated which is consistent with the previously reported analysis (Saglam *et al.*, 2013). It was noteworthy that innate immune systems including neutrophil and granulocyte activities were up-regulated while adaptive immune systems including T cell and leukocyte activities were down-regulated. In sPTD patients, the only biological process that was significantly (FDR < 0.05) up-regulated was glycerolipid metabolic pathway. However, a lot of mRNA processing and localization biological processes were significantly down-regulated.

Gene ontology analysis can categorize differentially expressed genes that are involved in significantly regulated pathways in both sPTD and PPROM patients. This categorization may improve preterm birth prediction by biologically meaningful feature selection. For example, with gene ontology analysis results that contains gene IDs for each significantly regulated biological processes (see Supplementary Table S1 at GitHub), it is possible to select genes that are involved in multiple pathways which could be high-weighted features for the prediction. In our sPTD prediction model, ATM (Entrez gene ID: 472) gene was excluded because it was ranked at 165th when sorted by p-value, even though it is involved in 3 among 5 most significantly down-regulated biological processes.

Acknowledgements

We would like to thank Professor Wei-Yi Cheng for his excellent lecture and helpful comments.

References

Chawla, N. V. *et al.* (2002). Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, **16**, 321–357.

Liao, Y. *et al.* (2019). Webgestalt 2019: gene set analysis toolkit with revamped uis and apis. *Nucleic acids research*.

Ritchie, M. E. *et al.* (2015). limma powers differential expression analyses for rna-sequencing and microarray studies. *Nucleic acids research*, **43**(7), e47–e47.

Romero, R. *et al.* (2013). A blueprint for the prevention of preterm birth: vaginal progesterone in women with a short cervix. *Journal of perinatal medicine*, **41**(1), 27–44.

Romero, R. *et al.* (2014). Preterm labor: one syndrome, many causes. *Science*, **345**(6198), 760–765.

Saglam, A. *et al.* (2013). The role of apoptosis in preterm premature rupture of the human fetal membranes. *Archives of gynecology and obstetrics*, **288**(3), 501–505.