Independent Component Analysis to Remove Batch Effects from Merged Microarray Datasets

Emilie Renard[1], Samuel Branders[2], and P.-A. Absil[1]

[1] Université catholique de Louvain, ICTEAM Institute
Avenue Georges Lemaître, 4 B-1348 Louvain-la-Neuve (Belgium)
{emilie.renard, pa.absil}@uclouvain.be
[2] Tools4Patient, rue Auguste Piccard, 48 B-6041 Gosselies (Belgium)

Abstract. Merging gene expression datasets is a simple way to increase the number of samples in an analysis. However experimental and data processing conditions, which are proper to each dataset, generally influence the expression values and can hide the biological effect of interest. It is then important to normalize the bigger merged dataset regarding those batch effects, as failing to adjust for them may adversely impact statistical inference. In this context, we propose to use a "spatiotemporal" independent component analysis to model the influence of those unwanted effects and remove them from the data. We show on a real dataset that our method allows to improve this modeling and helps to improve sample classification tasks.

Keywords: Batch effect removal, expression data, spatio-temporal independent component analysis

1 Introduction

Genes hold the information to build proteins, which are the structural components of cells and tissues. The translation of gene information into proteins is known as "gene expression". Nowadays, the development of sequencing technologies allows to measure those expression levels at a reasonable cost. The analysis of the resulting data helps to better understand how genes are working, with the goal of developing better cures for genetic diseases such as cancer.

Due to the limited number of samples that can be processed at the same time in an experiment, the size of such datasets is often limited in samples. However, statistical inferences need a high number of samples to be robust enough and generalizable to other data. As more and more of those datasets are available on public repositories such as GEO http://www.ncbi.nlm.nih.gov/geo/, merging and combining different datasets appears as a simple solution to increase the number of samples analyzed and potentially improve the relevance of the biological information extracted.

Expression levels of genes are the result of interactions between different biological processes, which can increase or decrease the expression level measured. However, noise may also be added at each step of data acquisition, due

to imprecisions or differences in experiment conditions. Confounding factors, or batch effects, that complicate the analysis of genomic data can be for example differences in dates of experiment, differences in laboratory conditions, or even the fact that two samples subsets were treated by two different technicians. The precise effects of the technical artefacts on gene expression levels is often unknown; however some partial information is usually available, such as the batch number, the date of experiment,... When merging different datasets, some of the main confounding factors are typically due to the fact that the samples were not processed in exactly the same conditions from one experiment/dataset to another. Those batch effects can be quite large and hide the effects related to the biological process of interest. Not including those effects in the analysis process may adversely affect the validity of biological conclusions drawn from the datasets [8,7,17]. It is then important to be able to combine datasets from different sources while removing the unwanted variations such as batch effects. From here, we will call *aggregated dataset* the bigger dataset resulting of the concatenation of the smaller datasets, and *sub-datasets* or *batches* these smaller datasets.

An additional difficulty in the process of removing batch effects is that the biological process (or phenotype) of interest could partially correlate with the batches. For example, if we want to combine two sub-datasets with respectively 75/25% and 25/75% of cases/controls, we should check that what is removed during the normalization step is really only the batch effect and does not contain potential useful information about cases/controls.

Different methods exist to tackle the problem of batch effect removal when merging different sub-datasets, each having its advantages and weaknesses [6] [2]. They can be classified in two main approaches: location-scale methods and matrix factorization methods. The location-scale methods assume a model for the distribution of the data within batches, and adjust the data within each batch to fit this model. The goal is to obtain genes with similar mean and/or variance for each batch. A main hypothesis is that by adjusting the gene distributions no biological information is removed. The matrix factorization methods assume that the variations across the sub-datasets (biological or due to confounding factors) can be represented by a small set of rank-one components which can be estimated by means of matrix factorization. The components associated with the batch effects are then removed to obtain the normalized dataset. With this approach, the main hypothesis is that the factorization method is able to pick up the batch effects in some of its resulting components.

In this paper, building on [1] and [12], we propose to use spatio-temporal Independent Component Analysis (ICA) to remove batch effects when combining microarray datasets. We compare our method to three other normalization methods. We show on a real dataset that spatio-temporal ICA allows to better model the factors influencing gene expression levels, and may improve results in a sample classification task.

The paper is organized as follows. Section 2 presents the method, which is validated in Section 3, and conclusions are drawn in Section 4.

2 Proposed method to reduce batch effect

Building on [1] and [12], we propose to use spatio-temporal Independent Component Analysis (ICA) to remove batch effects when combining microarray datasets. After factorization of the aggregated dataset, components showing some correlation with the sub-datasets are removed in order to obtain a final dataset, hopefully cleaned from the main batch effects. The advantage of a matrix factorization approach is that the removed components are interpretable: it is easy to check that they do not correlate with some biological information of interest. In [1], the authors use singular value decomposition (SVD) to model batch effects. However ICA was shown to better model the different sources of variation [17], so we propose here an ICA based approach. We first describe the spatio-temporal ICA of [12], then we explain how we use it to normalize the dataset.

2.1 Spatio-temporal Independent Component Analysis

We consider the aggregated dataset as a gene-by-sample matrix X, where $X_{i,j}$ indicates the value of gene i in sample j. Applying an ICA method to matrix X yields a decomposition

$$X \approx AB^T = \sum_{k=1}^{K} A_{:,k} B_{:,k}^T \tag{1}$$

where component $A_{:,k}$ can be interpreted as the gene activation pattern of component k and component $B_{:,k}$ as the weights of this pattern in the samples.

When computing this decomposition, the question arises whether one should maximize the independence between the columns of A or those of B. Independence across genes means that the activation patterns should be as independent as possible. Independence across samples means that the weights attributed to the activation patterns should be as independent as possible. In earlier times, because of the very vertical shape of matrix X in genetic datasets, independence across genes has been favored in the literature. However aggregating sub-datasets allows to have a more reasonable number of samples. Imposing independence among genes, or samples, or on both was shown to give good results [14]. As both options are justifiable a priori, we use a spatio-temporal ICA; this method introduces a trade-off parameter allowing an easy adaptation to the different options.

We now present the ICA method from [12] that we use to generate matrices A and B from the data matrix $X \in \mathbb{R}^{p \times n}$. The algorithm depends on a *spatiotemporal parameter* $\alpha \in [0, 1]$ that allows it to explore a continuum between imposing independence solely on A ($\alpha = 0$) and solely on B ($\alpha = 1$). The term "spatiotemporal" comes from the pixel-by-time data in medical imaging for which the concept was introduced [16].

The first step consists of centering the gene-by-sample data matrix X by subtracting the row and column means, followed by a dimensionality reduction by means of a K-truncated SVD, yielding a new matrix $\tilde{X} = U_K D_K V_K^T$. All the

possible decompositions of $\tilde{X}$ are given by $\tilde{X} = AB^T = AW^{-1}WB^T$ with W a $K \times K$ invertible matrix. The considered decomposition is then:

$$\tilde{X} = \underbrace{U_K D_K^{\alpha} W^{-1}}_{=:A} \underbrace{W D_K^{1-\alpha} V_K^T}_{=:B^T} \tag{2}$$

where W is restricted to the orthogonal group $\mathrm{O}(K) = \{W \in \mathbb{R}^{K \times K} : W^T W = I\}$. Consequently, the columns of A, resp. B, are structurally decorrelated when $\alpha = 0$, resp. $\alpha = 1$.

In the spirit of the JADE ICA algorithm [3], the objective function to minimize is of the form:

$$f_\alpha(W) = \alpha \sum_i \mathrm{Off}(C_i(B^T)) + (1-\alpha) \sum_i \mathrm{Off}(C_i(A^T)), \quad W \in \mathrm{O}(K)$$

where A and B depend on W through (2), $\mathrm{Off}(Y)$ returns the sum of squares of the off-diagonal elements of Y, and the C_i's are fourth-order cumulant matrices, satisfying the property $C_i(WM) = WC_i(M)W^T$. The minimization of f_α is thus a joint approximate diagonalization problem, which is addressed as in JADE using Jacobi rotations. The Jacobi algorithm is initialized with $W = I$, ensuring that both A and B initially have decorrelated columns.

2.2 Dataset normalization

The normalization process to remove batch effects is detailed in Algorithm 1. Matrices A and B are first computed (line 1), then we can use the components $B_{:,k}$ to remove possible batch effects. For this, we select the components $B_{:,k}$ that correlate with the batch. As batch is a categorical information and the components $B_{:,k}$ are continuous, the usual correlation formula (Pearson or Spearman) can not be used. To estimate which components are related to batch, we compute the R^2 value (line 2) that measures how well a variable x can predict a variable y in a linear model:

$$R^2(x, y) \equiv 1 - \frac{SS_{res}}{SS_{tot}}$$

where

- $SS_{tot} = \sum_i (y_i - \bar{y})^2$ is the sum of squares of the prediction errors if we take the mean $\bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_i$ as predictor or y,
- $SS_{res} = \sum_i (y_i - \hat{y}_i)^2$ is the sum of squares of the prediction errors if we use a linear model $\hat{y}_i = f(x_i)$ as predictor: if x is continuous the prediction model is a linear regression, if x is categorical we use a class mean.

The R^2 value indicates the proportion of the variance in y that can be predicted from x, and has the advantage to be usable with categorical or continuous variables. So the higher the R^2 value, the better the association between both variables. As the sub-datasets information is categorical, $R^2(\text{sub-datasets}, B_{:k})$

compares the prediction of B_{ik} by a general mean $\sum_j \frac{B_{jk}}{n}$ or by a sub-dataset mean $\sum_{j \in C_j} \frac{B_{jk}}{n}$ (where C_j represents all samples in the same sub-datasets as sample j).

If a component presents some correlation with the sub-datasets (line 3), then this component is selected. An additional step can be added in the process to check if the selected components do not correlate with some information of interest (lines 4-6, optional). The selected components are then removed from the matrix X to obtain a cleaned dataset (line 7).

Algorithm 1 ICA based normalization

Require: X $(p \times n)$ the aggregated dataset to be normalized, c (n) a categorical variable indicating the sub-datasets, α the spatio-temporal parameter for the ICA method, $t \in [0, 1]$ the threshold to consider a component associated to c, [optional] c_2 categorical/continuous information that we want to preserve

1: $A, B \leftarrow ICA(X, \alpha)$
2: $R \leftarrow R2(c, B)$
3: $ix \leftarrow which(R \geq t)$
4: $R_2 \leftarrow R2(c_2, B)$ ▷ optional
5: $ix_2 \leftarrow which(R_2 \geq R)$ ▷ optional
6: $ix \leftarrow ix \setminus ix_2$ ▷ optional
7: $X_n \leftarrow X - A[:, ix] * B[:, ix]^T$

3 Results

We tested our normalization method on breast cancer expression. We combined different datasets which can be accessed under GEO numbers GSE2034 [18] and GSE5327 [11], GSE7390 [4], GSE2990 [15], GSE3494 [10], GSE6532 [9] and GSE21653 [13]. All datasets were summarized with MAS5 and represented in log2 scale, except GSE6532 which was already summarized with RMA. With those datasets come different pieces of information: age of the patient, grade and size of the tumor, if a lymphatic node is affected, the estrogen receptor status, the treatment, the subtype, two values estimating the relapse risk (scoreGene76 and scoreODX), and one estimating the proliferation (scoreProlif). The last four are values computed from a model and the expression values, and so are more directly dependent on the dataset.

We took estrogen-receptor status (ER) prediction as the classification task. ER is thus our phenotype of interest, and other pieces of information will be called external information later in this paper. We removed the samples (or genes) with missing information which gives an aggregated dataset of 1361 samples for 22276 genes. The repartition of the ER status is described in Table 1. Proportion of ER positive samples depends on the dataset but is always in majority.

Sub-dataset	1	2	3	4	5	6
ER = 0	135	64	34	34	40	110
ER = 1	209	134	149	213	86	153
% of 1's	0.61	0.68	0.81	0.86	0.68	0.58

Table 1. Repartition of ER status in the sub-datasets

As can be seen on the first subplot in Fig. 1 the expression values in the aggregated dataset are clearly associated to the sub-datasets. Many genes have a strong association with the sub-datasets, but the maximal R^2 value between a gene and the ER status is about 0.25.

3.1 Comparison with centering-scaling, ComBat, and SVD based methods

Many methods exist that aim to remove the unwanted variation coming from the batch effects. We compare our method to three different approaches: the very simple standardization method, the well-used ComBat method and an SVD-based method that have a similar approach to our proposition.

The simplest way to normalize a dataset in order to remove batch effect is to standardize each sub-dataset separately. That is, for each gene in each sub-dataset, the expression values are centered and divided by their standard deviation.

Another widely used but more complex method is ComBat [5]. The expression value of gene g for sample j in batch i is modeled as $Y_{ijg} = \alpha_g + X\beta_g + \gamma_{ig} + \delta_{ig}\epsilon_{ijg}$ where α_g is the overall gene expression, and X represents the sample conditions. The error term ϵ_{ijg} is assumed to follow a normal distribution $N(0, \sigma_g^2)$. Additive and multiplicative batch effects are represented by parameters γ_{ig} and δ_{ig}. ComBat uses a bayesian approach to model the different parameters, and then removes the batch effects from the data to obtain the clean data $Y^*_{ijg} = \hat{\epsilon}_{ijg} + \hat{\alpha}_g + X\hat{\beta}_g$.

The third method, which we term SVD, is similar to [1]. The main difference is that a singular value decomposition is computed instead of an independent component analysis. As it is not clear how to systematically infer which components to remove in [1], we use our R^2 criterion.

Effects of the different normalization methods on the association between genes and sub-datasets or ER are visible on Fig. 1. Compared to the initial values, all methods about double the maximum R^2 value associated to the ER factor (from 0.25 to 0.5). Effects on association with sub-datasets are more different. The centering-scaling approach and ComBat remove all association with the sub-datasets. The methods based on matrix factorization are less sharp, the SVD one keeping the higher association with sub-datasets.

In the remaining of this section, we compare all four normalization methods (centering-scaling, ComBat, SVD and ICA based) and the case without normalization regarding possible batch effects. First we look in Section 3.2 at how the

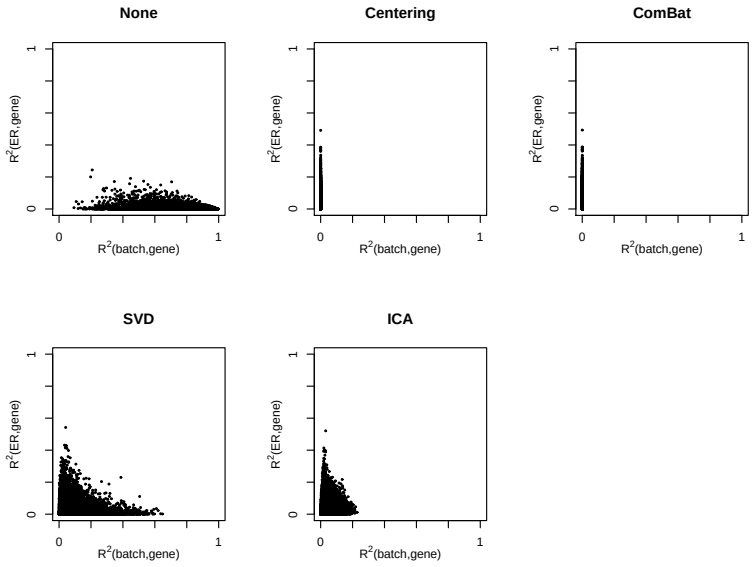

Fig. 1. R^2 values between the gene expression values and the sub-datasets versus the ER factor, for different normalizations of aggregated dataset.

method works and can be interpreted regarding the external information we have access to. This is only possible for the factorization methods. In a second time, we compare in Section 3.3 the results obtained in the context of a classification task.

3.2 Spatio-temporal ICA to model sources of variations

As a first step to validate our approach, we computed the ICA factorization of the unnormalized aggregated dataset for different values of α, yielding the components $A(\alpha)$ and $B(\alpha)$ as in Equation 1.

The maximal R^2 values between components of the matrix B and the external information (i.e. $\max_i R^2(\text{info}, B_{:i})$) are represented on Fig. 2. Information related to the sub-datasets appears to be captured quite well in at least one component $B_{:i}$. The quality of the recovering of the external information depends on the α value. If we compare with the SVD components, ICA is at least as good as SVD to recover the external information. For some factors like subType, scoreGene76, scoreProlif, scoreODX, treatment, and even ER in a smaller measure, the ICA factorization improves the modeling.

Influence of sub-datasets appears to be captured in the SVD decomposition, and in all values of α in ICA. However if we examine the relation between components and external information the behaviors differ. On Fig. 3 we represented for

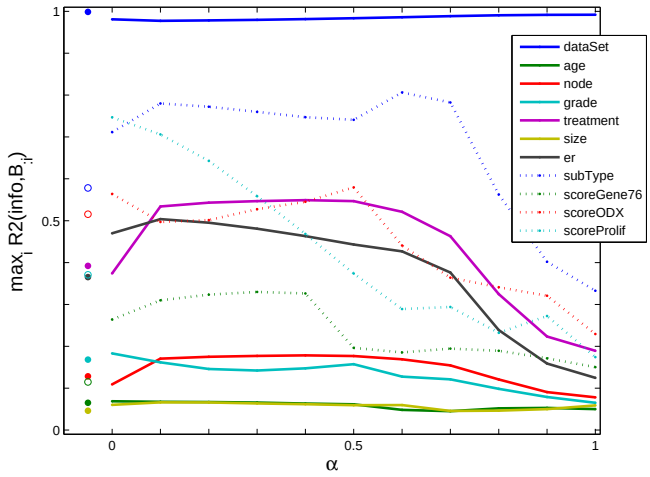

Fig. 2. Maximal R^2 values between components of the ICA factorization and the external information depending on α. The isolated dots on the left hand side give the same information but for the SVD decomposition.

different decompositions the R^2 values between all components and the external information, and the correlation between components themselves.

SVD gives two uncorrelated components highly associated to sub-datasets. ICA allows to increase the number of components associated to sub-dataset, especially for α values close to 0. However, some of those components are redundant: for $\alpha = 0$, components 1,2,4,6,8,9 have high R^2 values. But components 4 and 9 are correlated with component 2, and component 8 with component 1. Increasing the value of α imposes more and more independence on B and so enable to get rid of the redundancy between components. A good trade-off between recovering external information and avoiding redundancy would be an intermediate value of α.

3.3 Validation by impact on classification

Classification process description To compare the different methods, we used them in a whole process of classification task. We predicted the ER status using an SVM classifier. The whole process is described in Algorithm 2. The first step is to normalize the aggregated dataset X with the chosen method (here, centering-scaling, ComBat, SVD or ICA based normalization), then center and scale it to be sure to treat all features with the same weight (line 2). Training and testing sets are then separated (line 3). A basic feature selection is performed by selecting the 10 genes with the best association with the ER label based on a Wilcoxon test (line 4). A standard SVM model is trained based on those genes (lines 5 to 8). The labels of testing set are finally predicted using the SVM

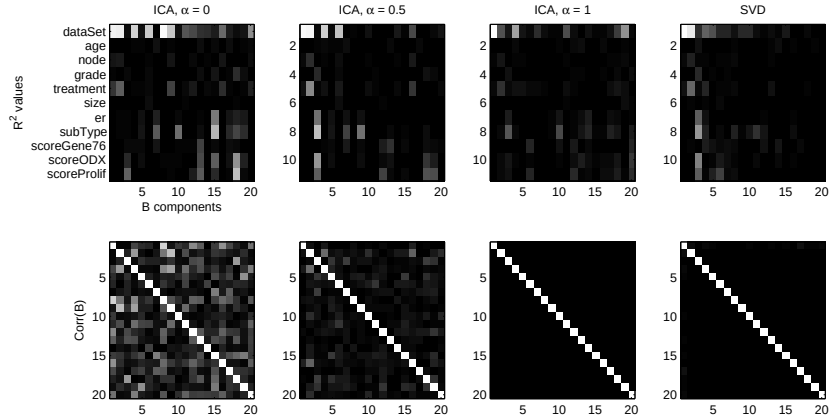

Fig. 3. Top: R^2 values between components resulting from different factorizations and the external information. Bottom: correlation between the different components. Black implies a null correlation (0), and white a perfect one (1).

model (line 9). To keep the SVM model simple, we used a linear kernel, the cost parameter is fixed using the heuristic implemented in the LiblineaR package, and the classes weights are set to $[1 - p_0, 1 - p_1]$ where p_i gives the proportion of samples in class i. In ICA and SVD based normalizations, we computed the $K = 20$ first components and removed the components with an R^2 value higher than $t = 0.5$.

Algorithm 2 Classification process

Require: X $(p \times n)$ aggregated matrix of genes expression, y (n) the label to predict, c (n) the sub-dataset information

1: $y_{tr}, y_{te} \leftarrow y$
2: $X_n \leftarrow normalize(X, c, y_{tr})$
3: $X_{tr}, X_{te} \leftarrow X_n$;
4: $idx_{bestGenes} \leftarrow Wilcoxon(y_{tr}, X_{tr})$
5: $X_{SVM} \leftarrow X_{tr}[idx_{bestGenes}, :]$
6: $c \leftarrow heuristic(X_{SVM})$
7: $w \leftarrow 1 - [\sharp y_{tr} == 1 \ \sharp y_{tr} == 0]$
8: $model_{SVM} \leftarrow SVM(X_{SVM}, y_{tr}, c, w)$
9: $\hat{y_{te}} \leftarrow prediction(model_{SVM}, X_{te}[idx_{bestGenes}, :])$

Results on dataset We kept each time two sub-datasets out of six for testing, and trained the SVM model on the four other sub-datasets, for a total of $C_6^2 = 15$ experiments.

The impact of α in the ICA based normalization on the results are illustrated on Fig. 4. An α closer to 1 tends to predict more positive labels. As discussed in Section 3.2, a value of $\alpha = 0.5$ appears to be a good compromise.

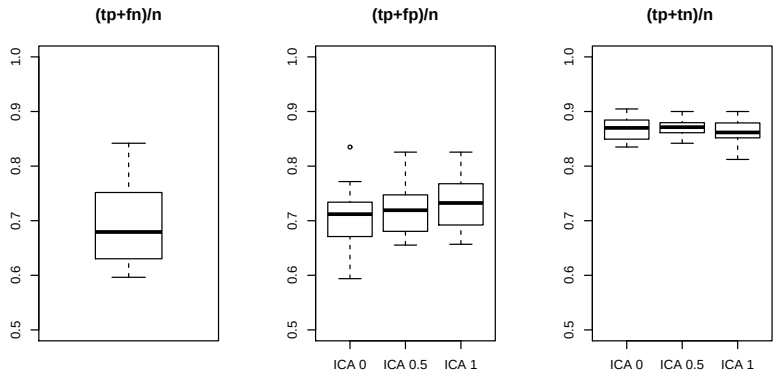

Fig. 4. Impact of α for the validating sets. On the left, proportion of positives in the sub-datasets. In the middle, proportion of positive predictions by the algorithm after applying an ICA based normalization. On the right, proportion of correct predictions.

The results on the testing set for the five methods (with $\alpha = 0.5$ for the ICA based) are shown on Fig. 5. The case without normalization appears to have a larger variance. ComBat has a smaller variance than the case without normalization, but bigger than the factorization methods. ICA and SVD are closer, ICA being slightly higher.

4 Conclusion

In the context of merging gene expression datasets to increase the number of samples analyzed and so the robustness of extracted information, we have proposed a method to remove batch effects. Inspired from existing methods, we have used a spatio-temporal independent component analysis to model those effects and remove it from the data. We have tested our method on a real breast cancer aggregated dataset in a classification task and compared it to other normalization methods. We have shown that our method can recover external information better than using a simple singular value decomposition. The spatio-temporal parameter α allows to adjust between modeling of external information and redundancy between components. By comparison with ComBat, the factorization approach enables to better understand what is removed in the cleaning process. Results on the classification task on the real dataset shows a slight improvement

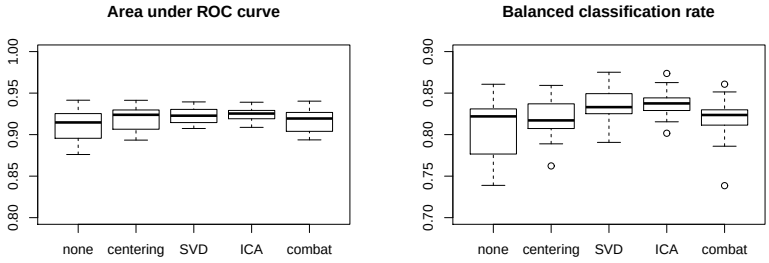

Fig. 5. Area under the ROC curve (obtained by varying the bias b in the separating hyperplane) and balanced classification rate ($0.5(\frac{TP}{TP+FN} + \frac{TN}{TN+FP})$) for the validating sets.

for the ICA based one. The next step would be to test it on a more difficult dataset where the labels correlate partially with the sub-datasets.

References

1. Alter, O., Brown, P. O., Botstein, D.: Singular Value Decomposition for Genome-Wide Expression Data Processing and Modeling. In: Proceedings of the National Academy of Sciences of the United States of America, vol. 97 (18), pp. 10101–10106 (2000)
2. Chen, C., Grennan, K., Badner, J., Zhang, D., Gershon, E., Jin, L., Liu, C.,: Removing batch effects in analysis of expression microarray data: an evaluation of six batch adjustment methods. PloS one, vol. 6 (2), e17238 (2011).
3. Cardoso, J.-F.:High-order contrasts for independent component analysis. Neural Comput., vol. 11 (1), pp 157–192 (1999).
4. Desmedt, C., et al.: Strong time dependence of the 76-gene prognostic signature for node-negative breast cancer patients in the TRANSBIG multicenter independent validation series. Clinical cancer research, vol. 13 (11), pp. 3207-3214 (2007).
5. Johnson, W., Li, C., Rabinovic, A.: Adjusting batch effects in microarray expression data using empirical Bayes methods. Biostatistics, vol. 8 (1), pp. 118–127 (2007)
6. Lazar, C., Meganck, S., Taminau, J., Steenhoff, D., Coletta, A., Molter, C., Weiss-Solís, D.Y., Duque, R., Bersini, H., Nowé, A.:Batch effect removal methods for microarray gene expression data integration: a survey. Briefings in bioinformatics, vol 14 (4), pp.469–490 (2013).
7. Leek, J. T et al. : Tackling the widespread and critical impact of batch effects in high-throughput data. Nat Rev Genet, vol. 11 (10), pp 733–739 (2010)
8. Leek, J. T., Storey, J. D.: Capturing Heterogeneity in Gene Expression Studies by Surrogate Variable Analysis. PLoS Genet, vol. 3 (9), e161 (2007)
9. Loi, S., et al: Definition of clinically distinct molecular subtypes in estrogen receptor–positive breast carcinomas through genomic grade. Journal of clinical oncology, vol. 25 (10), pp. 1239–1246 (2007).

10. Miller, L. D. et al.: An expression signature for p53 status in human breast cancer predicts mutation status, transcriptional effects, and patient survival. In: Proceedings of the National Academy of Sciences of the United States of America, vol. 102 (38), pp. 13550–13555 (2005).
11. Minn, A. J., et al.: Lung metastasis genes couple breast tumor size and metastatic spread. Proceedings of the National Academy of Sciences, vol. 104 (16), pp. 6740–6745 (2007).
12. Renard, E., Teschendorff, A. E., Absil P.-A. : Capturing confounding sources of variation in DNA methylation data by spatiotemporal independent component analysis. In: 22nd European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (2014)
13. Sabatier, R., Finetti, P., Cervera, N., Lambaudie, E., Esterni, B., Mamessier, E., Tallet, A., Chabannon, C., Extra, J.-M., Jacquemier, J., Viens, P., Birnbaum, D., and Bertucci, F.: A gene expression signature identifies two prognostic subgroups of basal breast cancer. Breast cancer research and treatment, vol. 126 (2), pp. 407-20 (2011).
14. Sainlez, M., Absil P.-A., Teschendorff, A.E.: Gene expression data analysis using spatiotemporal blind source separation. In: 17nd European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (2009)
15. Sotiriou, C., et al.: Gene expression profiling in breast cancer: understanding the molecular basis of histologic grade to improve prognosis. Journal of the National Cancer Institute, vol. 98 (4), pp. 262–272 (2006).
16. Stone, J. V., Porrill, J., Porter, N. R., Wilkinson, I. D.: Spatiotemporal independent component analysis of event-related fMRI data using skewed probability density functions. NeuroImage, vol. 15 (2), pp 407–421 (2002).
17. Teschendorff, A. E., Zhuang, J., Widschwendter, M.: Independent surrogate variable analysis to deconvolve confounding factors in large-scale microarray profiling studies. Bioinformatics, 27, 11, pp. 1496–1505 (2011)
18. Wang,Y., et al. :Gene-expression profiles to predict distant metastasis of lymph-node-negative primary breast cancer. The Lancet, vol. 365 (9460), pp. 671–679 (2005).