Comparing classic, deep and semi-supervised learning for whole-transcriptome breast cancer subtyping

Francisco Cristovao*, Arif Canakoglu*, Mark Carman, Silvia Cascianelli, Luca Nanni, Pietro Pinoli*, Marco Masseroli

Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Italy
first.last@polimi.it
**corresponding author*

Keywords: Deep Learning, Breast Cancer, Gene Expression, Semi-Supervised Learning, Variational Autoencoder.

Abstract. We investigate the important clinical problem of predicting prognosis-related breast cancer (BRCA) molecular subtypes using whole-transcriptome information present in The Cancer Genome Atlas Project (TCGA) dataset. From a Machine Learning perspective, the data is both high-dimensional with over nineteen thousand features, and extremely small with about one thousand labeled instances in total. To deal with the paucity of information we compare classical, deep and semi-supervised learning approaches on the subtyping task. Specifically, we compare L_1-regularised Logistic Regression, a 2-hidden layer Feed Forward Neural Network and a Variational Autoencoder based semi-supervised learner that makes use of pan-cancer TCGA data as well as BRCA data from a second source. We find that the classical supervised technique performs at least as well as the deep and semi-supervised learning techniques, although learning curve analysis suggests that insufficient unlabeled data may be being provided for the chosen semi-supervised learning technique to be effective.

1 Scientific Background

Over the last two decades, an accurate classification into prognostically relevant molecular subtypes has been recognized as crucial for deeper understanding of BRCA heterogeneity, improving patient outcome prediction, developing tailored treatments and supporting therapeutic decision making [1, 2]. A significant body of evidence has confirmed the prognostic meaning and predictive ability of the intrinsic molecular subtypes: *Luminal A, Luminal B, Her2-enriched, Basal and Normal-like* [2, 3], which were discovered in the early 2000s through unsupervised hierarchical clustering on BRCA microarray gene expression profiles [1]. To date, the subtypes are commonly identified using the PAM50 method [4], which implements the Prediction Analysis for Microarrays (PAM) classification algorithm and examines specifically the differential expression of a signature of 50 genes. Yet, many other genes could play relevant roles in defining discriminant patterns of gene expression across intrinsic subtypes. Consequently, genome-wide analysis of RNA-Sequence data could yield a substantial contribution by taking advantage of larger gene expression spaces. Recently, the number of publicly available BRCA samples profiled with RNA-Sequencing has dramatically increased. And although the PAM50 technique has been extensively adopted to categorize microarray and PCR-based gene expression data, only recently it has been applied to some RNA-Seq BRCA datasets [5]. Thus, only a small portion of available BRCA RNA-Seq gene expression data is labeled with intrinsic subtypes and hence usable for supervised learning. Furthermore, two main issues can affect classifier performance in learning intrinsic subtypes from these available RNA-Seq data: 1) the number of instances usable for training is always much smaller than the huge amount of genes in the feature space; and 2) the limits and uncertainties of the PAM50 method are inherited to some extent by any supervised method trained with PAM50 labeled data.

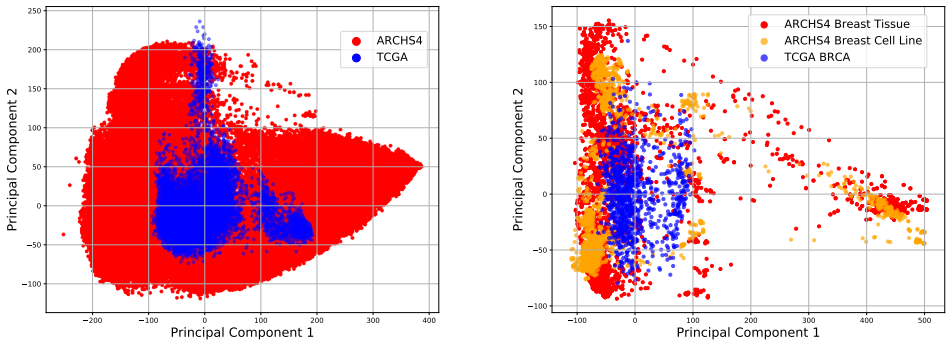

Figure 1: First two components of Principal Component Analysis of the two datasets used in this study. All samples (left); only Breast cancer samples (right).

In such a complex scenario, we implemented two baseline supervised methods to perform RNA-Seq BRCA sample classification into intrinsic subtypes: an L_1-regularised Logistic Regression and a Fully Connected Feed Forward Neural Network, for which we examined several architectures. Furthermore, we considered semi-supervised learning techniques, both to leverage on available unlabeled RNA-Seq BRCA samples and to evaluate the possible gain of including deep learning methods to tackle the BRCA intrinsic subtyping task. Particularly, we focused on Variational Autoencoders (VAEs) and Conditional Variational Autoencoders (CVAEs), since they can learn better continuous well-structured latent spaces by mixing deep learning with Bayesian inference. Hence, in this study we investigated the performance of these innovative deep approaches compared with baseline methods, exploring different architectures, hyper-parameters and regularisation techniques. Furthermore, for each approach we evaluated to what extent its accuracy is influenced by the dimension of the available labeled training samples, as to better assess the role and contribution of the semi-supervised learning methods.

2 Materials and Methods

2.1 Datasets

We used RNA-Seq data from the TCGA and ARCHS4 [6] public datasets. Both of them were downloaded from the ARCHS4 website as raw read counts of HiSeq 2000, HiSeq 2500 and NextSeq 500 platforms. We used the expression data for the 19,036 genes that are common to both datasets. For all the TCGA samples, subtype labels were traced on cBioPortal[1] and mainly come from PAM50 classification performed by Ciriello et al [5], leading to a total of 1053 labeled samples (546 *Luminal A*, 208 *Luminal B*, 179 *Basal*, 81 *HER2* and 39 *Normal-like*). This classification is not available in ARCHS4 experiments, which we used as unlabeled data. In order to use the data from both sources, we computed the *reads per million* (RPM) of each gene g_i in each sample s_j as: $RPM = \frac{\#\ reads\ mapped\ to\ g_i}{total\ reads\ for\ sample\ s_j} * 10^6$, and then applied log and min-max normalization.

After normalization, we performed Principle Component Analysis (PCA) on the combined dataset (TCGA+ARCHS4) and also on the respective breast cancer subsets, visualizing the first two components in Figure 1 to check their compatibility. Given the significant overlap between the first two components in both cases, no highly significant batch effects exist; thus, we used the two different datasets in the same experiment.

2.2 Supervised learning

We first explored traditional supervised learning approaches, which by definition resort to labeled data in order to learn the model parameters. In the following paragraphs we describe the two supervised methods used.

L_1-regularised Logistic Regression: In order to deal effectively with the very large number of input dimensions (19,036 genes) with respect to a very small number of

[1] https://www.cbioportal.org/

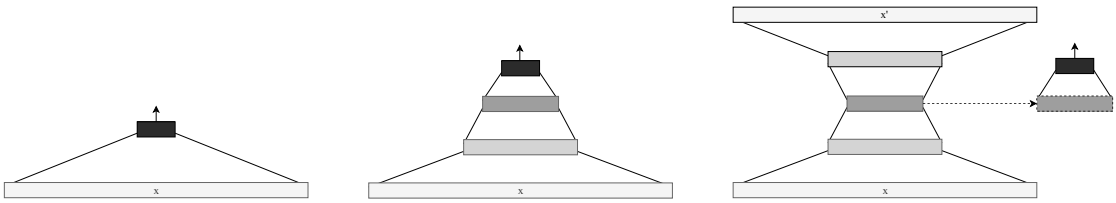

Figure 2: The three network architectures under comparison: Logistic Regression with L_1-regularisation (left), a Fully-connected Feed Forward Neural Network (centre), and a Variational Autoencoder with additional Softmax layer (right).

training instances (817 samples), we used multi-class Logistic Regression (LR) with a sparsity inducing L_1 penalty to prevent over-fitting to the training data. L_1-regularised LR minimizes the following cost function:

$$\mathcal{L}(\theta) = -\left[\sum_{i=1}^{n}\sum_{k=1}^{K} 1\left\{y^{(i)}=k\right\} \log \frac{\exp(\theta^{(k)\top}x^{(i)})}{\sum_{j=1}^{K}\exp(\theta^{(j)\top}x^{(i)})}\right] + \lambda \sum_{j=1}^{K} ||\theta^{(j)}||_1 \qquad (1)$$

where n and K denote the number of instances and classes, respectively, $\theta^{(k)}$ is the vector of model parameters for class k, λ is the shrinking factor of the regularisation term and $y^{(i)}, x^{(i)}$ are the class and feature vectors of the i-th sample, respectively. We used the L_1-regularised LR model implemented in the scikit-learn Python package[2].

Feed-Forward Neural Network: In the recent years powerful toolkits have been developed for Neural Network approaches to classification. These toolkits make use of sophisticated techniques (such as dropout, batch normalization, fast stochastic gradient descent routines, and extensive use of validation data) that can effectively control over-fitting in over-parameterized models. Thus, we compare the LR approach with fully-connected Feed-Forward Neural Networks of varying structures (Figure 2) (number and dimensions of hidden units), with ReLu as activation function for the hidden units and sigmoid for the output ones. The networks were implemented in Keras[3], and trained using Categorical Cross Entropy loss: $\mathcal{L}(\theta) = -\sum_{i=1}^{n}\sum_{k=1}^{K} 1\left\{y^{(i)}=k\right\} \log \hat{p}(k|x^{(i)};\theta)$ where $\hat{p}(k|x^{(i)};\theta)$ denotes the model's predicted probability for class k on instance i.

2.3 Semi-supervised learning

For the cancer subtype prediction, we have very few labeled gene expression instances to work with (817 labeled samples) compared to the very large feature space of genes to consider. One method for mitigating this problem consists in making use of available unlabeled gene expression data, using *semi*-supervised learning techniques. Semi-supervised learning refers to a set of machine learning methodologies that leverage (usually) large quantities of unlabeled samples in conjunction with typically small amounts of labeled data to improve the model performance on prediction tasks. The usefulness of semi-supervised learning is based on the assumption of *continuity* [7] between the unlabeled and labeled data, which requires that data points lying nearby in the feature space tend to have the same label.

Semi-supervised learning techniques often make use of clustering or dimensionality-reduction techniques to model the feature space, such that the class label information from the small number of labeled training instances can be generalized to unlabeled parts of the feature space.

Variational Autoencoder: A popular deep learning method for modeling unlabeled data is the Variational Autoencoder (VAE) [8]. Autoencoders are neural networks that are trained to perform dimensionality reduction. The network is structured to map high-dimensional input data down into a low-dimensional representation and then back out

[2] https://scikit-learn.org/stable/

[3] https://www.tensorflow.org/guide/keras/

Architecture	Unlabeled Data	Labeled Data	Accuracy (std-dev)
Logistic Regression with L_1-reg (19k→5)	–	TCGA BRCA	0.884 (±0.031)
Feed Forward NN (19k→300→100→5)	–	TCGA BRCA	0.876 (±0.027)
Feed Forward NN (19k→100→20→5)	–	TCGA BRCA	0.867 (±0.036)
VAE + Softmax (19k→300→100→5)	TCGA	TCGA BRCA	0.865 (±0.029)
VAE + Softmax (19k→100→20→5)	TCGA	TCGA BRCA	0.863 (±0.019)
VAE + Softmax (19k→300→100→5)	ARCHS4 BRCA	TCGA BRCA	0.875 (±0.022)
VAE + Softmax (19k→100→20→5)	ARCHS4 BRCA	TCGA BRCA	0.872 (±0.035)
CVAE + Softmax (19k→300→100→5)	TCGA	TCGA BRCA	0.851 (±0.022)
CVAE + Softmax (19k→100→20→5)	TCGA	TCGA BRCA	0.764 (±0.113)

Table 1: Comparison of prediction performance on validation data for different models, architectures and datasets, using 5-fold cross-validation. Numbers in parenthesis denote the dimension of the layers in the neural network architectures.

to the original dimension. Weights are learnt such that the reconstructed data is as close to the original input data as possible. *Variational* Autoencoders include an additional stochastic sampling step over the low-dimensional representation, before generating the output. This sampling process provides superior regularisation and interpretability of the latent representation.

We made use of a Variational Autoencoder for semi-supervised learning as follows. We first trained the autoencoder in an unsupervised manner to minimise reconstruction error in terms of binary cross-entropy[4]. We then added a Softmax layer to the low-dimensional representation (the concatenation of the mean and variance vectors[5]) and trained the weights of the Softmax to maximise prediction performance on the labeled BRCA data (see the network architecture on the right in Figure 2). While learning the Softmax weights we also fine-tuned the encoder component of the autoencoder on the supervised task since that was observed to improve performance markedly over keeping the encoder weights fixed[6].

Conditional Variational Autoencoder: As an extension of Variational Autoencoders, Conditional Variational Autoencoders (CVAE) [9] differ from Variational Autoencoders as they allow us to condition the induced low-dimensional feature space on different types of inputs. In our case, by conditioning on tissue type, we aimed at obtaining a feature space for the BRCA samples that is more suitable for the classification task, when compared with normal VAEs. The procedure to use CVAEs for classification was the same as the one explained for normal VAEs.

3 Experiments

In the following subsections we discuss the experiments performed during this study.

3.1 Experimental settings

Evaluation metrics: We compute Accuracy on the validation and test datasets, defined as the proportion of correct predictions made: $Acc = \frac{1}{n}\sum_{i=1}^{n} 1\left\{y^{(i)} = \hat{y}(x^{(i)})\right\}$ where $\hat{y}(x^{(i)})$ and $y^{(i)}$ denote the predicted and true class for the i^{th} datapoint.

Hyperparameter tuning and cross-validation: The various parameters of the shallow and deep learning models include the regularisation parameter, number of epochs, learning rate and dropout rates. An extensive parameter search was done for all of the considered architectures, specifically for Logistic Regression: different values of the regularisation parameter ($\lambda = 10^i, i \in \{-3, -2, \ldots, 4\}$); and for deep learning models: different learning rates (0.0005, 0.001, 0.01), number of epochs (25, 50, 75, 100) and input/hidden dropout rates (0, 0.2, 0.4, 0.6, 0.8).

[4]We experimented with MSE, but observed better prediction performance with binary cross-entropy.

[5]We also investigated sampling the hidden representation, but observed no performance improvement.

[6]We leave an investigation of combined classification + autoencoder loss to future work.

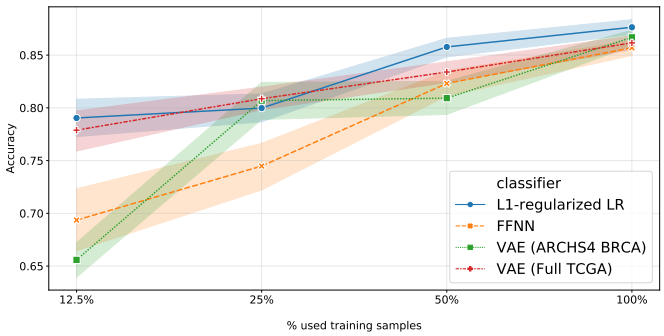

Figure 3: Learning curves for the evaluated classifiers on the validation data.

All considered models were evaluated using 5-fold cross-validation[7], reporting as aggregated performance score the mean accuracy across folds.

3.2 Experimental results

Model selection: Table 1 contains the cross-validation results for each considered model, using the corresponding best hyperparameters found. We note that regularised LR provides the highest overall average accuracy across the validation data, but the performance for all other models (except for CVAE) lies within one standard deviation.

Sensitivity to quantity of training data: In Figure 3, we provide learning curves on the validation data for four of the methods under comparison (L_1-regularised Logistic Regression, a Feed-Forward Neural Network, and a Variational Autoencoder trained on the ARCHS4 BRCA subset and the pan-cancer TCGA datasets)[8]. The curves show the effect on performance of reducing the amount of labeled training data available to the algorithm, while keeping the same proportion of BRCA subtypes. We note that performance in all cases increases with the amount of labeled training data. Importantly, however, at low quantities of training data the *relative* performance of the semi-supervised methods (particularly using the pan-cancer TCGA data) improves with respect to logistic regression. A possible explanation for this effect is that for small amounts of labeled data, the *relatively* larger amount of unlabeled data is able to provide a stronger regularisation effect (extracting relevant correlation information) across the whole-transcriptome, which can be exploited by the (weaker) subsequent classifier. If the case, this would suggest that larger quantities of unlabeled data may be needed to provide performance benefits from semi-supervised learning.

Performance on held-out test data: Having selected the most effective methods and architectures on the validation data[8], we evaluated them on the test data (Table 2), which was a held-out subset of the TCGA BRCA data, consisting of 236 samples. The higher accuracy results on the test data can be justified by minor differences on the distribution of the classes between the two subsets. However, since they were labeled by independent laboratories using slightly different pipelines, we decided not to re-balance them.

Architecture	**Unlabeled Data**	**Labeled Data**	**Accuracy**
Logistic Regression with L_1-reg (19k→5)	–	TCGA BRCA	0.936
Feed Forward NN (19k→300→100→5)	–	TCGA BRCA	0.907
VAE + Softmax (19k→300→100→5)	TCGA	TCGA BRCA	0.911
VAE + Softmax (19k→300→100→5)	ARCHS4 BRCA	TCGA BRCA	0.903

Table 2: Comparison of prediction performance on test data for chosen architectures.

Confusion Matrices: In Table 3, the confusion matrices for the models evaluated over the test set are presented. It can be verified that the *Basal* samples are the easiest to classify, whereas *Normal-like* are the ones with a larger percentage of miss-classified

[7]We used scikit-learn StratifiedKFold method to preserve the percentage of samples for each subtype.
[8]The CVAE was dropped from further analysis due to poor performance on the validation data.

LR + L_1 predicted labels

Actual labels	Ba	H2	LA	LB	NL
Ba	43	0	0	0	0
H2	0	16	0	0	0
LA	0	1	126	4	0
LB	0	0	2	30	0
NL	0	3	4	1	6

FFNN predicted labels

Actual labels	Ba	H2	LA	LB	NL
Ba	43	0	0	0	0
H2	0	14	1	1	0
LA	0	0	119	11	1
LB	0	0	1	31	0
NL	0	1	5	1	7

VAE (ARCHS4) + Softmax predicted labels

Actual labels	Ba	H2	LA	LB	NL
Ba	41	0	0	1	1
H2	0	13	2	1	0
LA	0	0	123	8	0
LB	0	0	3	29	0
NL	0	0	7	0	7

VAE (TCGA) + Softmax predicted labels

Actual labels	Ba	H2	LA	LB	NL
Ba	42	0	0	1	0
H2	0	14	1	1	0
LA	0	3	121	7	0
LB	0	0	1	31	0
NL	0	1	6	0	7

Table 3: Confusion Matrices on the test data: Logistic Regression with L_1-regularisation (top left), Feed Forward NN (top right), VAE + Softmax trained on ARCHS4 (bottom left) and VAE + Softmax trained on TCGA (bottom right). Ba, H2, LA, LB and NL correspond to *Basal*, *Her2-enriched*, *Luminal A*, *Luminal B* and *Normal-like*, respectively.

samples. It is also shown that the wrongly classified *Luminal A* and *Luminal B* samples occur mostly among each other. All of the results obtained in this analysis are in accordance with the literature on the subject [2].

4 Conclusion

We investigated breast cancer subtype prediction from whole-transcriptome information present in TCGA, comparing L_1-regularised Logistic Regression, a 2-hidden layer Feed Forward Neural Network and a Variational Autoencoder based semi-supervised learner that makes use of pan-cancer TCGA data or BRCA data from ARCHS4. We found the regularised LR to perform at least as well as the deep and semi-supervised learning techniques, although learning curve analysis suggests that the latter may have been provided with insufficient unlabeled data for it to be effective. The source code of the implemented analysis is available at: https://github.com/DEIB-GECO/brca_subtype

Acknowledgments The results shown here are in part based on data generated by the TCGA Research Network[9]. This research is funded by the ERC Advanced Grant project 693174 GeCo (Data-Driven Genomic Computing), 2016-2021.

References

[1] T. Sorlie, et al.. "Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications". *Proc Natl Acad Sci USA*, vol. 98, no. 19, pp. 10869-10874, 2001.

[2] X. Dai, et al.. "Breast cancer intrinsic subtype classification, clinical use and future trends". *American Journal of Cancer Research*, vol. 5, no. 10, pp. 2929-2943, 2015.

[3] T. Sorlie, et al.. "Repeated observation of breast tumor subtypes in independent gene expression data sets". *Proc Natl Acad Sci USA*, vol. 100, no. 14, pp. 8418-8423, 2003.

[4] J.S. Parker, M. Mullins, M.C. Cheang et al.. "Supervised risk predictor of breast cancer based on intrinsic subtypes". *Journal of Clinical Oncology*, vol. 27, no. 8, pp. 1160-1167, 2009.

[5] G. Ciriello, M. Gatza, A.H. Beck et al.. "Comprehensive molecular portraits of invasive lobular breast cancer". *Cell*, vol. 163, no. 2, pp. 506-519, 2015.

[6] A. Lachmann, D. Torre, A.B. Keenan et al.. "Massive mining of publicly available RNA-seq data from human and mouse". *Nature Communications*, vol. 9, no. 1, 1366, 2018.

[7] Y. Bengio, A. Courville, P. Vincent. "Representation learning: A review and new perspectives". *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798-1828, 2013.

[8] D.P. Kingma and M. Welling. "Auto-encoding variational bayes". *Proceedings of the International Conference on Learning Representations (ICLR)*, arXiv: 1312.6114v10, pp. 1-14, 2014.

[9] K. Sohn, H. Lee, X. Yan. "Learning structured output representation using deep conditional generative models". *Advances in Neural Information Processing Systems (NIPS)*, pp. 3483-3491, 2015.

[9] https://www.cancer.gov/tcga/