DropLasso: A robust variant of Lasso for single cell RNA-seq data

Beyrem Khalfaoui, Jean-Philippe Vert

DropLasso: A robust variant of Lasso for single cell RNA-seq data

Beyrem Khalfaoui [1,2] and Jean-Philippe Vert [3,1]

[1] MINES ParisTech, PSL Research University,
CBIO - Centre for Computational Biology, 75006 Paris, France
[2] Institut Curie, PSL Research University, INSERM, U900, 75005 Paris, France.
[3] Google Brain, 75009 Paris, France.
firstname.lastname@mines-paristech.fr

Abstract

Single-cell RNA sequencing (scRNA-seq) is a fast growing approach to measure the genome-wide transcriptome of many individual cells in parallel, but results in noisy data with many dropout events. Existing methods to learn molecular signatures from bulk transcriptomic data may therefore not be adapted to scRNA-seq data, in order to automatically classify individual cells into predefined classes.
We propose a new method called DropLasso to learn a molecular signature from scRNA-seq data. DropLasso extends the dropout regularisation technique, popular in neural network training, to estimate sparse linear models. It is well adapted to data corrupted by dropout noise, such as scRNA-seq data, and we clarify how it relates to elastic net regularisation. We provide promising results on simulated and real scRNA-seq data, suggesting that DropLasso may be better adapted than standard regularisations to infer molecular signatures from scRNA-seq data.
DropLasso is freely available as an R package at `https://github.com/jpvert/droplasso`

1 Introduction

The fast paced development of massively parallel sequencing technologies and protocols has made it possible to measure gene expression with more precision and less cost in recent years. Single-cell RNA sequencing (scRNA-seq), in particular, is a fast growing approach to measure the genome-wide transcriptome of many individual cells in parallel (Kolodziejczyk et al., 2015). By giving access to cell-to-cell variability, it represents a major advance compared to standard "bulk" RNA sequencing to investigate complex heterogeneous tissues (Macosko et al., 2015; Tasic et al., 2016; Zeisel et al., 2015; Villani et al., 2017) and study dynamic biological processes such as embryo development (Deng et al., 2014) and cancer (Patel et al., 2014).

The analysis of scRNA-seq data is however challenging and raises a number of specific modelling and computational issues (Ozsolak and Milos, 2011; Bacher and Kendziorski, 2016). In particular, since a tiny amount of RNA is present in each cell, a large fraction of polyadenylated RNA can be stochastically lost during sample preparation steps including cell lysis, reverse transcription or amplification. As a result, many genes fail to be detected even though they are expressed, a type of errors usually referred to as *dropouts*. In a standard scRNA-seq experiment it is common to observe more than 80% of genes with no apparent expression in each single cell, an important proportion of which are in fact dropout errors (Kharchenko et al., 2014). The presence of so many zeros in the raw data can have significant impact on the downstream analysis and biological conclusions, and has given rise to new statistical models for data normalisation and visualisation (Pierson and Yau, 2015; Risso et al., 2018) or gene differential analysis (Kharchenko et al., 2014).

Besides exploratory analysis and gene-per-gene differential analysis, a promising use of scRNA-seq technology is to automatically classify individual cells into pre-specified classes, such as particular cell types in a cancer tissue. This requires to establish cell type specific "molecular signatures" that could be shared and used consistently across laboratories, just like standard molecular signatures are commonly used to classify tumour samples into subtypes from bulk transcriptomic data (Ramaswamy et al., 2001; Sørlie et al., 2001,

2003). From a methodological point of view, molecular signatures are based on a *supervised analysis*, where a model is trained to associate each genome-wide transcriptomic profile to a particular class, using a set of profiles with class annotation to select the genes in the signature and fit the parameters of the models. While the classes themselves may be the result of an unsupervised analysis, just like breast cancer subtypes which were initially defined from a first unsupervised clustering analysis of a set of tumours (Perou et al., 2000), the development of a signature to classify any new sample into one of the classes is generally based on a method for supervised classification or regression.

Signatures based on a few selected genes, such as the 70-gene signature for breast cancer prognosis of van de Vijver et al. (2002), are particularly useful both for interpretability of the signature, and to limit the risk of overfitting the training set. Many techniques exist to train molecular signatures on bulk transcriptomic data (Haury and Vert, 2010), however, they may not be adapted to scRNA-seq data due to the inflation of zeros resulting from dropout events.

Interestingly and independently, the term "dropout" has also gained popularity in the machine learning community in recent years, as a powerful technique to regularise deep neural networks (Srivastava et al., 2014). Dropout regularisation works by randomly removing connexions or nodes during parameter optimisation of a neural network. On a simple linear model (a.k.a. single-layer neural network), this is equivalent to randomly creating some dropout noise to the training examples, i.e., to randomly set some features to zeros in the training examples (Wager et al., 2013; Baldi and Sadowski, 2013). Several explanations have been proposed for the empirical success of dropout regularisation. Srivastava et al. (2014) motivated the technique as a way to perform an ensemble average of many neural networks, likely to reduce the generalisation error by reducing the variance of the estimator, similar to other ensemble averaging techniques like bagging (Breiman, 1996) or random forests (Breiman, 2001). Another justification for the relevance of dropout regularisation, particularly in the linear model case, is that it performs an intrinsic data-dependent regularisation of the estimator (Wager et al., 2013; Baldi and Sadowski, 2013) which is particularly interesting in the presence of rare but important features. Yet another justification for dropout regularisation, particularly relevant for us, is that it can be interpreted as a *data augmentation* technique, a general method that amounts to adding virtual training examples by applying some transformation to the actual training examples, such as rotations of images or corruption by some Gaussian noise; the hypothesis being that the class should not change after transformation. Data augmentation has a long history in machine learning (e.g., Schölkopf et al., 1996), and is a key ingredient of many modern successful applications of machine learning such as image classification (Krizhevsky et al., 2012). As shown by van der Maaten et al. (2013), dropout regularisation in the linear model case can be interpreted as a data augmentation technique, where corruption by dropout noise enforces the model to be robust to dropout events in the test data, e.g., to blanking of some pixels on images or to removal of some words in a document. Wager et al. (2014) show that in some cases, data augmentation with dropout noise allows to train model that should be insensitive to such noise more efficiently than without.

Since scRNA-seq data are inherently corrupted by dropout noise, we therefore propose that dropout regularisation may be a sound approach to make the predictive model robust to this form of noise, and consequently to improve their generalisation performance on scRNA-seq supervised classification. Since plain dropout regularisation does not lead to feature selection and to the identification of a limited number of genes to form a molecular signature, we furthermore propose an extension of dropout regularisation, which we call *DropLasso* regularisation, obtained by adding a sparsity-inducing ℓ_1 regularisation to the objective function of the dropout regularisation, just like *lasso* regression adds an ℓ_1 penalty to a mean squared error criterion in order to estimate a sparse model (Tibshirani, 1996). We show that the ℓ_1 penalty can be integrated in the standard stochastic gradient algorithm used to implement dropout regularisation, resulting in a scalable stochastic *proximal* gradient descent formulation of DropLasso. We also clarify the regularisation property of DropLasso, and show that it is to elastic net regularisation what plain dropout regularisation is to the plain ridge regularisation. Finally, we provide promising results on simulated and real scRNA-seq data, suggesting that specific regularisations like DropLasso may be better adapted than standard regularisations to infer molecular signatures from scRNA-seq data.

2 Methods

2.1 Setting and notations

We consider the supervised machine learning setting, where we observe a series of n pairs of the form $(x_i, y_i)_{i=1,\dots,n}$. For each $i \in [1, n]$, $x_i \in \mathbb{R}^d$ represents the gene expression levels for d genes measured in the i-th cell by scRNA-seq, and $y_i \in \mathbb{R}$ or $\{-1, 1\}$ is a label to represent a discrete category or a real number associated to the i-th cell, e.g., a phenotype of interest such as normal vs tumour cell, or an index of progression in the cell cycle. For $i \in [1, n]$ and $j \in [1, d]$, we denote by $x_{i,j} \in \mathbb{R}$ the expression level of gene j in cell i. From this training set of n annotated cells, the goal of supervised learning is to estimate a function to predict the label of any new, unseen cell from its transcriptomic profile. We restrict ourselves to linear models $f_w : \mathbb{R}^d \to \mathbb{R}$, for any $w \in \mathbb{R}^d$, of the form

$$\forall u \in \mathbb{R}^d, \quad f_w(u) = \sum_{i=1}^{d} w_i u_i \,.$$

To estimate a model on the training set, a popular approach is to follow a penalised maximum likelihood or empirical risk minimisation principle and to solve an objective function of the form

$$\min_{w \in \mathbb{R}^d} \left\{ \frac{1}{n} \sum_{i=1}^{n} L(w, x_i, y_i) + \lambda \Omega(w) \right\}, \tag{1}$$

where $L(w, x_i, y_i)$ is a loss function to assess how well f_w predicts y_i from x_i, Ω is an (optional) penalty to control overfitting in high dimensions, and $\lambda > 0$ is a regularisation parameter to control the balance between under- and overfitting. Examples of classical loss functions include the square loss:

$$L_{\text{square}}(w, x_i, y_i) = \left(y_i - \sum_{j=1}^{d} w_j x_{i,j} \right)^2 ,$$

and the logistic loss:

$$L_{\text{logistic}}(w, x_i, y_i) = \log \left(1 + \exp(-y_i \sum_{j=1}^{d} w_j x_{i,j}) \right) ,$$

which are popular losses when y_i is respectively a continuous ($y_i \in \mathbb{R}$) or discrete ($y_i \in \{-1, 1\}$) label. As for the regularisation term $\Omega(w)$ in (1), popular choices include the ridge penalty (Hoerl and Kennard, 1970):

$$\Omega_{\text{ridge}}(w) = \|w\|_2^2 = \sum_{i=1}^{d} w_i^2 \,,$$

and the lasso penalty (Tibshirani, 1996):

$$\Omega_{\text{lasso}}(w) = \|w\|_1 = \sum_{i=1}^{d} |w_i| \,.$$

The properties, advantages and drawbacks of ridge and lasso penalties have been theoretically studied under different assumptions and regimes. The lasso penalty additionally allows feature selection by producing sparse solutions, i.e., vectors w with many zeros; this is useful to in many bioinformatics applications to select "molecular signatures", i.e., predictive models based on the expression of a limited number of genes only. It is known however that lasso can be unstable in particular when there are several highly correlated features in the data. It also cannot select more features than the number of observations and its accuracy is often dominated by that of ridge. For these reasons, another popular penalty is elastic net, which encompasses the advantages of both penalties Zou and Hastie (2005) :

$$\Omega_{\text{elastic net}}(w) = \alpha \|w\|_2^2 + (1 - \alpha)\|w\|_1 \,,$$

where $\alpha \in [0, 1]$ allows to interpolate between the lasso ($\alpha = 0$) and the ridge ($\alpha = 1$) penalties.

2.2 DropLasso

For scRNA-seq data subject to dropout noise, we propose a new model to train a sparse linear model robust to the noise by artificially augmenting the training set with new examples corrupted by dropout. Formally, given a vector $u \in \mathbb{R}^d$ and a dropout mask $\delta \in \{0,1\}^d$, we consider the corrupted pattern $\delta \odot u \in \mathbb{R}^d$ obtained by entry-wise multiplication $(\delta \odot u)_i = \delta_i u_i$. In order to consider all possible dropout masks, we make δ a random variable with independent entries following a Bernoulli distribution of parameter $p \in [0,1]$, i.e., $P(\delta_i = 1) = p$, and consider the following DropLasso regularisation for any $\lambda > 0$, $p \in [0,1]$ and loss function L:

$$\min_{w \in \mathbb{R}^d} \left(\frac{1}{n} \sum_{i=1}^{n} \mathop{\mathbb{E}}_{\delta_i \sim B(p)^d} L(w, \delta_i \odot \frac{x_{i,}}{p}, y_i) + \lambda \left\| w \right\|_1 \right) . \tag{2}$$

In this equation, the expectation over the dropout mask corresponds to an average of 2^d terms. The division by p in the term x_i/p is here to ensure that, on average, the inner product between w and $\delta_i \odot \frac{x_{i,}}{p}$ is independent of p, because:

$$\begin{aligned} \mathop{\mathbb{E}}_{\delta_i \sim B(p)^d} \sum_{j=1}^{d} w_j \left(\delta_i \odot \frac{x_{i,}}{p} \right)_j &= \sum_{j=1}^{d} \mathop{\mathbb{E}}_{\delta_{i,j} \sim B(p)} w_j \delta_{i,j} \frac{x_{i,j}}{p} \\ &= \sum_{j=1}^{d} w_j x_{i,j} \, . \end{aligned}$$

When $p = 1$ and $\lambda > 0$, the only mask with positive probability is the constant mask with all entries equal to 1, which performs no dropout corruption. In that case, DropLasso (2) therefore boils down to standard lasso. When $\lambda = 0$ and $p < 1$, on the other hand, DropLasso boils down to the standard dropout regularisation proposed by Srivastava et al. (2014) and studied, among others, by Wager et al. (2013); Baldi and Sadowski (2013); van der Maaten et al. (2013). In general, DropLasso interpolates between lasso and dropout. For $\lambda > 0$, it inherits from lasso regularisation the ability to select features associated with ℓ_1 regularisation (Bach et al., 2011). We therefore propose DropLasso as a good candidate to select molecular signatures (thanks to the sparsity-inducing ℓ_1 regularisation) for data corrupted with dropout noise, in particular scRNA-seq data (thanks to the dropout data augmentation).

2.3 Algorithm

For any convex loss function L such as the square or logistic losses, DropLasso (2) is a non-smooth convex optimisation problem whose global minimum can be found by generic solvers for convex programs. Due to the dropout corruption, the total number of terms in the sum in (2) is $n \times 2^d$. This is usually prohibitive as soon as d is more than a few, e.g., in practical applications when d is easily of order 10^4 (number of genes). Hence the objective function (2) can simply not be computed exactly for a single candidate model w, and even less optimised by methods like gradient descent.

To solve (2), we instead propose to follow a stochastic gradient approach to exploit the particular structure of the model, in particular the fact that it is fast and easy to generate a sample randomly corrupted by dropout noise. A similar approach is used for standard dropout regularisation when L is differentiable w.r.t. w (Srivastava et al., 2014), however in our case we additionally need to take care of the non-differentiable ℓ_1 norm; this can be handled by a forward-backward algorithm which, plugged in the stochastic gradient loop, leads to the proximal stochastic gradient descent algorithm presented in Algorithm 1. The fact that Algorithm 1 is correct, i.e., converges to the solution of (2), follows under weak conditions from general results on stochastic approximations and proximal stochastic gradient descent algorithms (Robbins and Siegmund, 1971; Atchadé et al., 2017).

We can easily see that for $p = 1$, our algorithm becomes a classical stochastic proximal descent algorithm. On the other hand when $\lambda = 0$, the soft threshholding operator becomes the identity and we turn back to the stochastic gradient descent with the dropout trick.

When $p = 1$ (no dropout), it is known that the solution of (2) is sparse, and is even 0 when λ is larger than a value λ_{max} than can be used as initial value when one wants to compute the set of solutions over a

Algorithm 1 Solving DropLasso

Require: Training set $(x_i, y_i)_{i=1,\dots,n}$, initialisation $w_0 \in \mathbb{R}^d$, initial learning rate $\gamma_0 > 0$, learning rate decay $\beta > 0$, number of passes $n_{passes} \in \mathbb{N}$, $\lambda \geq 0$, $p \in [0, 1]$

procedure DropLasso
  $w^0 \leftarrow w_0$
  $t \leftarrow 0$
  for $iter = 1$ `to` n_{passes} **do**
    $\pi \leftarrow$ random permutation of $[1, n]$ ▷ Shuffle training set
    for $i = 1$ `to` n **do** ▷ (Mini-)batch also possible
      $\gamma_t \leftarrow \gamma_0/(1 + \beta t)$
      Sample $\delta \sim Bernoulli(p)^d$
      $z \leftarrow \delta \odot x_{\pi(i)}/p$
      $w^{t+1} \leftarrow S_{\gamma_t \lambda}(w^t - \gamma_t \nabla_w L(w^t, z, y_{\pi(i)}))$ ▷ $S_{\gamma_t\lambda}$is the soft-thresholding operator
      $t \leftarrow t + 1$
    end for
  end for
return w^t
end procedure

decreasing grid of values for λ. Interestingly, this property also holds when $p < 1$, with the same λ_{max} value which therefore does not depend on p:

Theorem 1. *For a loss function of the form $L(w, x, y) = \ell_y(w^\top x)$ where ℓ_y is convex and differentiable at 0 for all y, $w = 0$ is solution of (2) if and only if $\lambda \geq \lambda_{\max}$ with*

$$\lambda_{\max} = \left\| \frac{1}{n} \sum_{i=1}^{n} \ell'_{y_i}(0) x_i \right\|_\infty . \tag{3}$$

Proof. Under the assumptions of the theorem, the function $w \to F(w)$ with

$$F(w) = \frac{1}{n} \sum_{i=1}^{n} \mathop{\mathbb{E}}_{\delta_i \sim B(p)^d} L(w, \delta_i \odot \frac{x_i}{p}, y_i) = \frac{1}{n} \sum_{i=1}^{n} \mathop{\mathbb{E}}_{\delta_i \sim B(p)^d} \ell_{y_i} \left(w^\top (\delta_i \odot \frac{x_i}{p}) \right)$$

is convex and its subdifferential is

$$\partial F(w) = \frac{1}{n} \sum_{i=1}^{n} \mathop{\mathbb{E}}_{\delta_i \sim B(p)^d} \partial \ell_{y_i} \left(w^\top (\delta_i \odot \frac{x_i}{p}) \right) \delta_i \odot \frac{x_i}{p} . \tag{4}$$

At $w = 0$, this simplifies to

$$\partial F(0) = \frac{1}{n} \sum_{i=1}^{n} \mathop{\mathbb{E}}_{\delta_i \sim B(p)^d} \ell'_{y_i}(0) \, \delta_i \odot \frac{x_i}{p} = \frac{1}{n} \sum_{i=1}^{n} \ell'_{y_i}(0) \, x_i .$$

Besides, the subdifferential of $w \to \|w\|_1$ at $w = 0$ is $\partial \|\cdot\|_1(0) = \{u : \|u\|_\infty \leq 1\}$. Using the standard characterization that w is solution of the convex problem (2) if and only if $0 \in \partial \left(F + \lambda \|\cdot\|_1\right)(w)$, we get that $w = 0$ is a solution of (2) if and only if $-\partial F(0) \in \lambda \partial \|\cdot\|_1(0)$, or equivalently $\|\partial F(0)\|_\infty \leq \lambda$. The theorem follows by using (4). □

In practice, for the square loss $\ell_y(u) = (u - y)^2$, we get $\ell'_y(0) = -2y$; and for the logistic loss $\ell_y(u) = \ln(1 + e^{-yu})$, we get $\ell'_y(0) = -y/2$. Taking

$$S = \frac{1}{n} \sum_{i=1}^{n} y_i x_i ,$$

we therefore have the following $\lambda_{\max}$ values for respectively the square and logistic losses:

$$\lambda_{\max}^{\text{square}} = 2\|S\|_\infty\,, \quad \lambda_{\max}^{\text{logistic}} = \frac{\|S\|_\infty}{2}\,.$$

In order to get the regularization path of DropLasso, i.e., the set solutions (2) when λ varies for a fixed p, we therefore first fix a grid of values to test for λ in an interval $[\lambda_{\min}, \lambda_{\max}]$ where $\lambda_{\max}$ is given by (3) and, for example $\lambda_{\min} = \lambda_{\max}/100$. We then iteratively solve (2) using Algorithm 1 for decreasing values of λ using warm restart, i.e., taking the solution for the previous λ as initialization for the next λ. Since 0 is the solution for $\lambda = \lambda_{\max}$, we initialize the first optimization with $w_0 = 0$.

2.4 DropLasso and elastic net

As we already mentioned, DropLasso interpolates between lasso ($p = 1, \lambda > 0$) and dropout ($p \in [0,1]$, $\lambda = 0$). On the other hand, dropout regularisation is known to be related to ridge regularisation (Wager et al., 2013; Baldi and Sadowski, 2013); in particular, for the square loss, dropout regularisation boils down to ridge regression after proper normalisation of the data, while for more general losses it can be approximated by reweighted version of ridge regression. Here we show that DropLasso largely inherits these properties, and in a sense is to elastic net what dropout is to ridge.

Let us start with the square loss. In that case we have the following:

Theorem 2. *If the data are scaled so that*

$$\forall j \in [1,d]\,, \quad \frac{1}{n}\sum_{i=1}^n x_{i,j}^2 = 1\,,$$

then solving the DropLasso problem (2) with parameters λ and p and the square loss L_{square} is equivalent to solving the elastic net problem

$$\min_{w\in\mathbb{R}^d} \frac{1}{n}\sum_{i=1}^n L_{square}(w, x_i, y_i) + \lambda_{enet}\left(\alpha_{enet}\|w\|_2^2 + (1-\alpha_{enet})\|w\|_1\right)\,,$$

with

$$\lambda_{enet} = \lambda + \frac{1-p}{p} \quad \textit{and} \quad \alpha_{enet} = \frac{1-p}{1-p+\lambda p}\,.$$

Proof. By developing the error function and marginalising over the Bernoulli variables, we can rewrite the objective function of (2) as follows:

$$\begin{aligned}
&\frac{1}{n}\sum_{i=1}^n \mathop{\mathbb{E}}_{\delta_i\sim B(p)^d} L_{\text{square}}(w, \delta_i \odot \frac{x_i,}{p}, y_i) + \lambda\,\|w\|_1 \\
&= \frac{1}{n}\sum_{i=1}^n \mathop{\mathbb{E}}_{\delta_i\sim B(p)^d} \left(y_i - \sum_{j=1}^d w_j\delta_{i,j}\frac{x_{i,j}}{p}\right)^2 + \lambda\,\|w\|_1 \\
&= \frac{1}{n}\sum_{i=1}^n \left(y_i - \sum_{j=1}^d w_j x_{i,j}\right)^2 + \frac{1}{n}\sum_{i=1}^n\sum_{j=1}^d w_j^2 x_{i,j}^2 \mathrm{Var}\left(\frac{\delta_{i,j}}{p}\right) + \lambda\,\|w\|_1 \\
&= \frac{1}{n}\sum_{i=1}^n L_{\text{square}}(w, x_i, y_i) + \frac{1-p}{p}\sum_{j=1}^d \left(\frac{1}{n}\sum_{i=1}^n x_{i,j}^2\right) w_j^2 + \lambda\,\|w\|_1 \\
&= \frac{1}{n}\sum_{i=1}^n L_{\text{square}}(w, x_i, y_i) + \frac{1-p}{p}\|w\|_2^2 + \lambda\,\|w\|_1\,,
\end{aligned}$$

and Theorem 2 easily follows by identifying λ_{enet} and α_{enet} from this equation. □

We note that conversely, in order to solve an elastic net problem with parameters λ_{enet} and α_{enet}, one can equivalently solve a DropLasso problem with parameters

$$\lambda = \lambda_{\text{enet}}\,(1 - \alpha_{\text{enet}}) \quad \text{and} \quad p = \frac{1}{1 + \lambda_{\text{enet}}\alpha_{\text{enet}}}\,.$$

When the data are not scaled as in Theorem 2, then instead of a standard elastic net penalty the DropLasso problem with square loss is equivalent to a modified elastic net problem where the ℓ_2 norm is weighted by the vector of mean squared norm of each column in the data matrix.

In the case of the logistic loss, we can also adapt a result of Wager et al. (2013) which relates dropout to an adaptive version of ridge regression:

Property 1. *: For the logistic loss, DropLasso can be approximated when the dropout probability p is close to 1 by an adaptive version of elastic net that automatically scales the data but also that encourages more confident predictions.*

Proof. Writing the Taylor expansion for the logistic loss up to the second order when the dropout is small (p close to 1), we obtain the following quadratic approximation to the dropout loss on a point:

$$\begin{aligned} L(w, \delta_{\mathbf{i}} \odot \frac{x_{i,}}{p}, y_i) \simeq\; & L(w, x_{i,}, y_i) \\ & + \sum_{j=1}^{d} \frac{\partial L(w, x_{i,}, y)}{\partial x_{i,j}} \left(\frac{\delta_{i,j}}{p} - 1\right) x_{i,j} \\ & + \frac{1}{2} \sum_{j=1}^{d} \sum_{k=1}^{d} \frac{\partial^2 L(w, x_{i,}, y)}{\partial x_{i,j} \partial x_{i,k}} \left(\frac{\delta_{i,j}}{p} - 1\right) \left(\frac{\delta_{i,k}}{p} - 1\right) x_{i,j} x_{i,k}\,. \end{aligned}$$

Taking the expectation with respect to $\delta_i \sim B(p)^d$, the first order term cancels out since $\mathbb{E}\delta_{i,j} = p$ for all $j \in [1, d]$. The off-diagonal second-order term also disappear because $\delta_{i,j}$ and $\delta_{i,k}$ are independent for $j \neq k$. Noting that for $\delta \sim B(p)$ it holds that

$$\mathbb{E}\left(\frac{\delta}{p} - 1\right)^2 = \frac{1-p}{p}\,,$$

and that for the logistic loss,

$$\frac{\partial^2 L_{\text{logistic}}(w, x_{i,}, y)}{\partial x_{i,j}^2} = \pi_i(1 - \pi_i) w_j^2\,,$$

where $\pi = e^{w^\top x_i} / \left(1 + e^{w^\top x_i}\right) = P_w(Y = 1 \,|\, X = x_i)$ under the logistic model parametrized by w, we finally get the following quadratic approximation:

$$L_{\text{logistic}}(w, \delta_{\mathbf{i}} \odot \frac{x_{i,}}{p}, y_i) \simeq L_{\text{logistic}}(w, x_{i,}, y_i) + \frac{1-p}{2p} \sum_{j=1}^{d} \pi_i(1 - \pi_i) w_j^2 x_{i,j}^2\,.$$

We finally get the following approximation to the DropLasso objective function:

$$\begin{aligned} & \frac{1}{n} \sum_{i=1}^{n} \mathop{\mathbb{E}}_{\delta_i \sim B(p)^d} L_{\text{logistic}}(w, \delta_i \odot \frac{x_{i,}}{p}, y_i) + \lambda \left\|w\right\|_1 \\ & \simeq \frac{1}{n} \sum_{i=1}^{n} L_{\text{logistic}}(w, x_{i,}, y_i) + \frac{1-p}{2p} \sum_{j=1}^{d} \gamma_j w_j^2 + \lambda \left\|w\right\|_1\,, \end{aligned}$$

where for $j \in [1, d]$,

$$\gamma_j = \sum_{i=1}^{n} \frac{\partial^2 L(w, x_{i,}, y)}{\partial^2 w^\top x_{i,j}} . x_{i,j} = \sum_{i=1}^{n} \pi_i(1 - \pi_i) x_{i,j}^2\,.$$

□

This shows that with the logistic loss, that the ridge penalty corresponding to the approximation of the Droplasso is controlled both by the size of the features $x_{i,j}^2$, but also by the fact that the prediction for each sample is confident or not. In fact γ_j is maximal when $\pi_i = 0.5$ for all $i \in [1, n]$, which means that the model is not confident about the examples it is learnt with.

3 Results

3.1 Simulation results

We first investigate the performance of DropLasso on simulated data, and compare it to standard dropout and elastic net regularisation. We design a toy simulation to illustrate in particular how corruption by dropout noise impacts the performances of the different methods. The simulation goes as follow :

- We set the dimension to $d = 20$.
- Each sample is a random vector $z \in \mathbb{N}^d$ with entries following a Poisson distribution with parameter $\pi = 1$. The data variables are independent.
- The "true" model is a logistic model with sparse weight vector $w \in \mathbb{R}^d$ satisfying $w_i = +10, i = 1 \dots d_1$, $w_i = -10, i = (d_1 + 1) \dots 2d_1$, and $w_i = 0$ for $i = (2d_1 + 1), \dots, d$. d_1 here is fixed to 2 and thus we have 4 active predictors (with signal) in this simulation.
- Using w as the true underlying model and z as the true observations, we simulate a label $y \sim \text{Bernoulli}(1/(1 + \exp(-\sum_{j=1}^{d} w_j z_j)))$.
- We introduce corruption by dropout events by multiplying entry-wise z with an i.i.d Bernoulli variables δ with probability q.

We simulate $n = 100$ samples to train elastic net and DropLasso models, and evaluate their performance in terms of area under the receiving operator curve (AUC) on $10,000$ independent samples. Both models have two parameters, λ and α for elastic net, λ and p for DropLasso. We vary each parameter over a grid: α over 11 regularly spaced values between 0 and 1, p over the grid 0.6^n for $n = 0, \dots, 10$, and λ over a regular grid of 10 values between $\lambda_{\max}$ and $\lambda_{\max}/100$, where $\lambda_{\max}$ is the smallest value such that the solution of the optimization problem is the null model (see Theorem 1). All model are trained on the training set, and the best parameter set is chosen as the one that maximizes the AUC on an independent validation set of 10,000 samples; only the AUC of the best model is then reported on the test set. We repeat the whole procedure 1,000 times in order to estimate the variability of the performance of each method.

Table 1: Test AUC of elastic net and DropLasso regression on simulations with different amount of dropout noise on the training data. The $*$ indicates that a method significantly outperforms the other (i.e., $P < 0.05$ according to a paired t-test comparing the AUC over 1,000 repeats).

Noise rate	**Elastic net**	**DropLasso**
q=1	$0.974 \pm 0.006^*$	0.954 ± 0.012
q=0.4	0.641 ± 0.043	0.639 ± 0.027
q=0.2	0.554 ± 0.031	$0.561 \pm 0.021^*$

Table 1 shows the classification performance in terms of test AUC of elastic net and DropLasso, when we vary the amount of dropout noise in the training data. We first observe that, for both methods, the performance drastically decreases when dropout noise increases, confirming the difficulty induced by dropout events to learn predictive models. Second, we note that in the absence of noise, elastic net significantly outperforms DropLasso. However, when the amount of noise increases, both methods perform similarly (for $q = 0.4$), and ultimately DropLasso outperforms elastic net in the configuration with large dropout noise ($q = 0.2$). This confirms that DropLasso provides potential benefits in situations where data are corrupted by dropout noise.

3.2 Classification on Single Cell RNA-seq

We now turn on to real scRNA-seq data. To evaluate the performance of methods for supervised classification, we collected 4 publicly available scRNA-seq datasets amenable to this setting, as summarised in Table 2. These datasets were pre-processed by Soneson and Robinson (2017), and we downloaded them from the *conquer* website[1], a collection of consistently processed, analysis-ready and well documented publicly available scRNA-seq data sets. We used the preprocessed length-scaled transcripts per million mapped reads (see Soneson and Robinson, 2017, for details about data processing). These datasets were used by Soneson and Robinson (2017) to assess the performance of methods for gene differential analysis between classes of cells, and we follow the same splits of cells into classes for our experiments of supervised classification.We used the available sample annotations to create binary classification problems, as described in Table 2. Note that some datasets have more than two classes, in which case we created several binary classification problems.

Table 2: Description of the scRNA-seq data and the corresponding (binary) classification tasks.

Dataset	**Classification task**	**Variables**	**Samples**
EMTAB2805	Cell cycle phase: G1 vs G2M	18,979	96 ; 96
EMTAB2805	Cell cycle phase: S vs G1	18,740	96 ; 96
EMTAB2805	Cell cycle phase: S vs G2	18,873	96 ; 96
GSE45719	mid blastocyst vs 16-cell stage blastomere	22,059	50 ; 60
GSE45719	8-cell stage blastomere vs 16-cell stage blastomere	21,590	50 ; 60
GSE48968	BMDC 1h LPS vs 4h LPS Stimulation	16,439	95 ; 96
GSE48968	BMDC 4h LPS vs 6h LPS Stimulation	15,719	95 ; 96
GSE74596	NKT0 vs NKT17	15,642	45 ; 44
GSE74596	NKT0 vs NKT1	14,962	45 ; 46
GSE74596	NKT1 vs NKT2	16,135	46 ; 48

On each of the 10 resulting binary classification problems, we compare the performance of 5 regularisation methods for logistic regression: lasso, ridge, elastic net, dropout and DropLasso. We train the different models on 20% of the data chosen in such way that labels are balanced, choose the best hyper-parameter(s) for each on a 20% validation set, and finally evaluate the performance of the resulting models on the 60% remaining data. We search the best parameters for each method over the same grid as described for the simulation study above (except that lasso, ridge and dropout have a single parameter to tune). We report in Table 3 the averag test AUC corresponding to the best parameters.

Table 3: Mean test AUC score for different regularizations schemes, on different binary classification problems.)

Dataset	**dropout**	**DropLasso**	**elastic net**	**ridge**	**lasso**
EMTAB2805, G1 vs G2M	0.96	0.97	0.98	0.97	0.94
EMTAB2805, G1 vs S	0.98	0.97	0.98	0.98	0.91
EMTAB2805, S vs G2M	0.99	0.98	0.99	0.99	0.95
GSE45719, 16-cell vs Mid blastocyst	1.00	0.99	1.00	1.00	0.99
GSE45719, 16-cell vs 8-cell	0.98	0.95	0.97	0.98	0.72
GSE48968, 1h vs 4h	1.00	1.00	1.00	1.00	1.00
GSE48968, 4h vs 6h	0.84	0.84	0.86	0.85	0.79
GSE74596, NKT0 vs NKT17	1.00	1.00	0.99	0.99	1.00
GSE74596, NKT0 vs NKT1	1.00	1.00	1.00	1.00	0.99
GSE74596, NKT1 vs NKT2	0.98	0.98	0.99	0.99	0.98

The first observation is that the performances reached by all methods on all datasets are generally high, and can reach a perfect AUC score of 1 on some of the datasets. This suggests that the labels chosen in these datasets are sufficiently different in terms of transcriptomic profiles that they can be easily recognised most of the time. We still notice some differences in performance between datasets, with GSE48968 with

[1] http://imlspenticton.uzh.ch:3838/conquer/

1h-4h stimulation labels being the easiest dataset to classify while GSE48968 with 6h-4h stimulation labels is the most challenging, for all methods. This contrast of performance for the same dataset confirms that supervised learning on single cell data can be challenging when the labels are biologically close regardless of the preprocessing step. Soneson and Robinson (2017) also noticed a difference in signal-to-noise ratio between these datasets, in the context of gene differential analysis. Second, we observe that the lasso is clearly the worst performing method in terms of accuracy, while all other methods tend to have similar accuracies. To further analyze the relative performance of different methods, we perform statistical tests for each pair of methods on each dataset, and call a method a "winner" if it is statistically more accurate than the other method ($P < 0.05$ for a t-test on the test AUC). Figure 1 reports, for each method, the number of times it is a winner. The plot first confirms that lasso is the least performing method in terms of accuracy, and that elastic net and dropout are the methods that have the largest number of wins. Although it was expected that elastic net improves over the lasso in this high-dimensional data setting, where many genes are correlated through several regulatory networks (Abdelmoez et al., 2018), it is also interesting to see that elastic net slightly outperforms ridge indicating that at least some of these biological labels can be explained by a sparse model. Dropout outperforming ridge indicates that the adaptive regularisation that dropout introduces is relevant to this type of data. Finally, DropLasso only outperforms the lasso method, but in contrast with dropout (and ridge) does allow for feature selection and the discovery of potential biomarkers, which we study next.

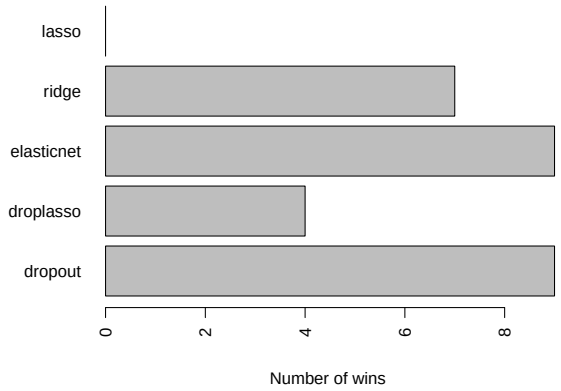

Figure 1: Number of significant wins for each method over all datasets

Table 4 shows the average number of selected features for each method on each classification problem. Selected features are defined by having nonzero coefficients in the corresponding model after fixing its parameters. We use a sensitivity threshold $\epsilon = 10^{-8}$ to account for potential convergence issues (coefficients below this threshold are considered as null). According to Table 4, lasso is the method with the highest values of sparsity (that is selecting the most compact sets of features for the classification task) with an average selected set size of 6.63, coming before DropLasso with an average of 676. It is interesting that elastic net does perform feature selection but with a much bigger average selected size of $11,869$. Ridge and dropout do not perform feature selection if we do not account for coefficients below the threshold.

Providing a compact set of features that can discriminate the task labels with high accuracy is important not only for computational time and memory footprint but more importantly for the interpretability of the model and the identification of a minimal set of features or a *molecular signature* of the observed phenotype. Using the reported results in the previous tables, we compare in Figure 2 the trade off presented by the different methods between accuracy, as evaluated by mean AUC for each dataset, and model sparsity that can be defined by the proportion of features not selected for each dataset, where each point is the best validated model for one method on one dataset. Figure 2 confirms the fact that most accurate models are not sparse, and presents DropLasso as the method that trades off best sparsity and accuracy, presenting a more sparse alternative to elastic net, and a more accurate alternative to lasso.

Table 4: Average number of selected variables for the different models

Dataset	Variables	dropout	DropLasso	elastic net	ridge	lasso
EMTAB2805, G1 vs G2M	18,979	18,973	274	13,117	14,597	7
EMTAB2805, G1 vs S	18,740	18,733	291	13,089	14,606	7
EMTAB2805, S vs G2M	18,979	18,867	41	8,193	13,088	6
GSE45719, 16-cell vs Mid blastocyst	22,059	21,965	4	19,747	19,747	3
GSE45719, 16-cell vs 8-cell	21,590	21,413	4,892	17,133	21,393	7
GSE48968, 1h vs 4h	16,439	16,431	18	7,071	10,139	7
GSE48968, 4h vs 6h	15,719	15,711	594	8,994	12,998	14
GSE74596, NKT0 vs NKT17	15,642	15,416	60	7,000	8,758	5
GSE74596, NKT0 vs NKT1	14,962	14,806	33	6,364	6,364	5
GSE74596, NKT1 vs NKT2	16,135	16,020	55	7,368	9,148	5

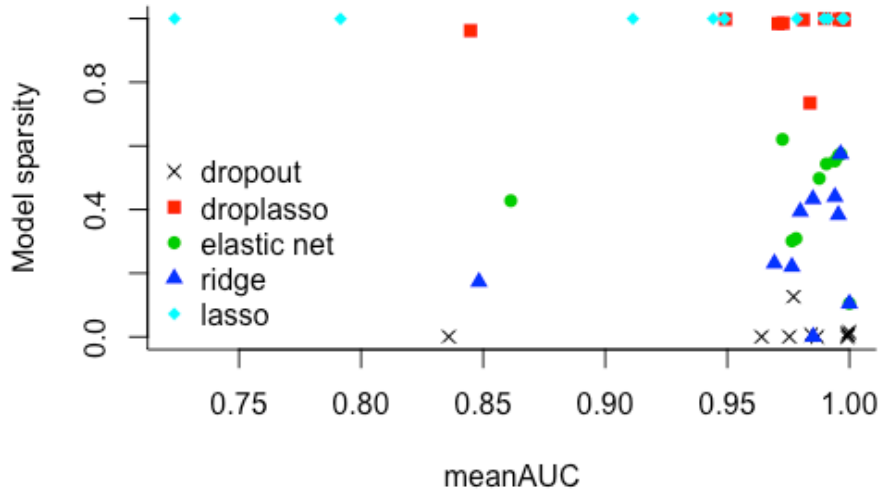

Figure 2: Scatter plot of mean AUC against mean model sparsity for different models, across the different datasets. Each point represents a method tested on one of the classification problems.

3.2.1 Biological significance of the selected features:

To conclude this section, we now evaluate the biological relevance of the gene lists or the molecular signatures estimated by the two methods that consistently provided sparse models, that is the lasso and DropLasso regularisation. We first illustrate this comparison on the first dataset, EMTAB2805, where the goal is to discriminate mice cells at the G1 from the G2M cell cycle stages. To this end, we retrain the different methods with the parameters corresponding to the best accuracy but this time on all the samples, and then we perform a Gene Ontology enrichment analysis using DAVID (Huang et al., 2009) on the subset of genes with non-zero coefficients for each method.

For this dataset and the best tuning parameters, DAVID identifies 24 genes selected by DropLasso and 5 genes selected by lasso. While the analysis of the genes selected by DropLasso shows enrichment in the functional term "positive regulation of mitotic cell cycle", the genes selected by the lasso method do not include the terms "cell division", "cell cycle" or "mitosis". Among the genes selected by DropLasso, 5 genes were related to the functional term "cell cycle" and 2 genes were related to the term "cell division". It is interesting to notice first that 4 out of 5 genes selected by lasso were related to ATP synthesis which underlies the potential importance of the relationship between energy and the cell cycle, as reviewed in (Salazar-Roa and Malumbres, 2017), and second that all the genes selected by lasso were also selected by DropLasso, which

shows that DropLasso potentially allows for the discovery of novel biomarkers.

The enrichment analysis on the GSE48968 dataset, where the goal is to discriminate between primary mice dendritic cells exposed to 1 hour LPS stimulation and 4 hours stimulation, identifies 8 genes selected by DropLasso and 4 genes selected by lasso. Although both sets were enriched with the term "response to virus", DropLasso set shows enrichment for "immune response", "inflammatory response" and "cellular response to lipopolysaccharide", as it also interestingly shows enrichment for the terms "defense response to Gram-negative bacterium" and " cellular response to tumor necrosis factor" , as it is known that lipopolysaccharide stimulates the production of tumor necrosis factor (TNF)-α (Barsig et al., 1995; Ogikubo et al., 2004). While the analysis of lasso selected genes does not reveal any enriched functional annotation cluster, one cluster is enriched in the DropLasso genes set and appears to be mainly related to cytokines and chemokine which were previously shown to have very altered profiles by LPS stimulation (Medvedev et al., 2000; Kopydlowski et al., 1999; Johnston et al., 1998). Interestingly, here again all the genes selected by lasso are also selected by DropLasso.

Finally, the enrichment analysis on the GSE74596 with the classification task between natural killer T cell subsets (NKT0 vs NKT1) shows some differences in the selected genes by DropLasso and lasso, where some genes selected by lasso are not selected by DropLasso (3 out of 6 identified genes by lasso). While both methods are mostly enriched with the same terms: "CTL mediated immune response against target cells" and "Ras-Independent pathway in NK cell-mediated cytotoxicity", DropLasso set additionally shows enrichment for two terms including the term "Immunoglobulin" and three terms including the term "major histocompatibility complex (MHC) " molecules, that are both related by definition to T-cells.

Overall, this short analysis of the molecular signatures estimated by lasso and DropLasso confirms that a small number of relevant genes tend to be selected by both methods, and the fact that DropLasso significantly outperforms lasso in AUC on most datasets confirms that its list of genes is likely to be more complete than that selected by lasso.

4 Discussion

ScRNA-seq is changing the way we study cellular heterogeneity and investigate a number of biological processes such as differentiation or tumourigenesis. Yet, as the throughput of scRNA-seq technologies increases and allows to process more and more cells simultaneously, it is likely that the amount of information captured in each individual cell will remain limited in the future and that dropout noise will continue to affect scRNA-seq (and other single-cell technologies).

Several techniques have been proposed to handle dropout noise in the context of data normalisation or gene differential expression analysis, and shown to outperform standard techniques widely used for bulk RNA-seq data analysis. In this paper we investigate a new setting which, we believe, will play an important role in the future: supervised classification of cell populations into pre-specified classes, and selection of molecular signatures for that purpose. Molecular signatures for the classification of tissues from bulk RNA-seq data has already had a tremendous impact in cancer research, and as more and more cell types are investigated and discovered with scRNA-seq it is likely that specific molecular signatures will be useful in the future to automatically sort cells into their classes.

DropLasso, the new technique we propose, borrows the recent idea of dropout regularisation from machine learning, and extends it to allow feature selection. While a parallel between dropout regularisation and (data-dependent) ridge regression has already been shown by Wager et al. (2013) and Baldi and Sadowski (2013), it is reassuring that we are able to extend this parallel to DropLasso and elastic net regularisation.

More interesting is the fact that, on both simulated and real data, we obtained promising results with DropLasso in terms of trade-off between accuracy and feature selection. They suggest that, again, specific models tailored to the data and noise can give an edge over generic models developed under different assumptions.

The intuition behind why dropout (and DropLasso) perform well on scRNA-seq data, however, remains a bit unclear. Our main motivation to use them in this context was to see them as data augmentation techniques, where training data are corrupted according to the noise we assume in the data. While we believe this is fundamentally the reason why we obtain promising results, alternative explanations for the success of dropout have been proposed, and may also play a role in the context of scRNA-seq. They include for example the interpretation of dropout as a regulariser similar to a data-dependent weighted version of

ridge regularisation, which works well in the presence of rare but important features (Wager et al., 2013); it would be interesting to clarify if the regularisation induced by DropLasso on scRNA-seq data exploits some fundamental property of these data, and may be replaced by a more direct approach to model this.

Finally, this first study of dropout and DropLasso regularisation on biological data paves the way to many future direction. For example, it is known that the probability of dropout in scRNA-seq data depends on the gene expression level (Kharchenko et al., 2014; Risso et al., 2018). It would therefore be interesting to study both theoretically and empirically if a dropout regularisation following a similar pattern may be useful. Second, instead of independently perturbing the different features one may create a correlation between the dropout events in different genes. Creating a correlation may be a way to create new regularisation by generating a *structured* dropout noise. It may for example be possible to derive a correlation structure for dropout noise from prior knowledge about gene annotations or gene networks in order to enforce a structure in the molecular signature, just like structured ridge and lasso penalties have been used to promote structure molecular signatures with bulk transcriptomes (Rapaport et al., 2007; Jacob et al., 2009).

Funding

This work has been supported by the European Research Council (grant ERC-SMAC-280032).

References

Abdelmoez, M. N., Iida, K., Oguchi, Y., Nishikii, H., Yokokawa, R., Kotera, H., Uemura, S., Santiago, J. G., and Shintaku, H. (2018). Sinc-seq: correlation of transient gene expressions between nucleus and cytoplasm reflects single-cell physiology. *Genome biology*, 19(1):66.

Atchadé, Y. F., Fort, G., and Moulines, E. (2017). On perturbed proximal gradient algorithms. *Journal of Machine Learning Research*, 18:1–33.

Bach, F., Jenatton, R., Mairal, J., and Obozinski, G. (2011). Optimization with sparsity-inducing penalties. *Foundations and Trends® in Machine Learning*, 4(1):1–106.

Bacher, R. and Kendziorski, C. (2016). Design and computational analysis of single-cell RNA-sequencing experiments. *Genome Biology*, 17(63).

Baldi, P. and Sadowski, P. J. (2013). Understanding dropout. In Burges, C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K., editors, *Adv. Neural. Inform. Process Syst.*, volume 26, pages 2814–2822. Curran Associates, Inc.

Barsig, J., Küsters, S., Vogt, K., Volk, H.-D., Tiegs, G., and Wendel, A. (1995). Lipopolysaccharide-induced interleukin-10 in mice: role of endogenous tumor necrosis factor-α. *European journal of immunology*, 25(10):2888–2893.

Breiman, L. (1996). Bagging predictors. *Mach. Learn.*, 24(2):123–140.

Breiman, L. (2001). Random forests. *Mach. Learn.*, 45(1):5–32.

Deng, Q., Ramsköld, D., Reinius, B., and Sandberg, R. (2014). Single-cell RNA-seq reveals dynamic, random monoallelic gene expression in mammalian cells. *Science*, 343(6167):193–6.

Haury, A.-C. and Vert, J.-P. (2010). On the stability and interpretability of prognosis signatures in breast cancer. In *Proceedings of the Fourth International Workshop on Machine Learning in Systems Biology (MLSB10)*. To appear.

Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67.

Huang, D. W., Sherman, B. T., and Lempicki, R. A. (2009). Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists. *Nucl. Acids Res.*, 37:1–13.

Jacob, L., Obozinski, G., and Vert, J.-P. (2009). Group lasso with overlap and graph lasso. In *Proceedings of the 26th annual international conference on machine learning*, pages 433–440. ACM.

Johnston, C., Finkelstein, J., Gelein, R., and Oberdörster, G. (1998). Pulmonary cytokine and chemokine mrna levels after inhalation of lipopolysaccharide in c57bl/6 mice. *Toxicological Sciences*, 46(2):300–307.

Kharchenko, P. V., Silberstein, L., and Scadden, D. T. (2014). Bayesian approach to single-cell differential expression analysis. *Nat. Methods*, 11(7):740–742.

Kolodziejczyk, A. A., Kim, J. K., Svensson, V., Marioni, J. C., and Teichmann, S. A. (2015). The technology and biology of single-cell RNA sequencing. *Molecular Cell*, 58(4):610–620.

Kopydlowski, K. M., Salkowski, C. A., Cody, M. J., van Rooijen, N., Major, J., Hamilton, T. A., and Vogel, S. N. (1999). Regulation of macrophage chemokine expression by lipopolysaccharide in vitro and in vivo. *The Journal of Immunology*, 163(3):1537–1544.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In Pereira, F., Burges, C., Bottou, L., and Weinberger, K., editors, *Adv. Neural. Inform. Process Syst.*, volume 25, pages 1097–1105. Curran Associates, Inc.

Macosko, E. Z., Basu, A., Satija, R., Nemesh, J., Shekhar, K., Goldman, M., Tirosh, I., Bialas, A. R., Kamitaki, N., Martersteck, E. M., Trombetta, J. J., Weitz, D. A., Sanes, J. R., Shalek, A. K., Regev, A., and McCarroll, S. A. (2015). Highly Parallel Genome-wide Expression Profiling of Individual Cells Using Nanoliter Droplets. *Cell*, 161(5):1202–1214.

Medvedev, A. E., Kopydlowski, K. M., and Vogel, S. N. (2000). Inhibition of lipopolysaccharide-induced signal transduction in endotoxin-tolerized mouse macrophages: dysregulation of cytokine, chemokine, and toll-like receptor 2 and 4 gene expression. *The Journal of Immunology*, 164(11):5564–5574.

Ogikubo, Y., Norimatsu, M., Sasaki, Y., Yasuda, A., Saegusa, J., and Tamura, Y. (2004). Effect of lipopolysaccharide (lps) injection on the immune responses of lps-sensitive mice. *Journal of veterinary medical science*, 66(10):1189–1193.

Ozsolak, F. and Milos, P. M. (2011). Rna sequencing: advances, challenges and opportunities. *Nature reviews genetics*, 12(2):87.

Patel, A. P., Tirosh, I., Trombetta, J. J., Shalek, A. K., Gillespie, S. M., Wakimoto, H., Cahill, D. P., Nahed, B. V., Curry, W. T., Martuza, R. L., Louis, D. N., Rozenblatt-Rosen, O., Suvà, M. L., Regev, A., and Bernstein, B. E. (2014). Single-cell RNA-seq highlights intratumoral heterogeneity in primary glioblastoma. *Science*, 344(6190):1396–1401.

Perou, C. M., Sørlie, T., Eisen, M. B., van de Rijn, M., Jeffrey, S. S., Rees, C. A., Pollack, J. R., Ross, D. T., Johnsen, H., Akslen, L. A., Fluge, O., Pergamenschikov, A., Williams, C., Zhu, S. X., Lønning, P. E., Børresen-Dale, A. L., Brown, P. O., and Botstein, D. (2000). Molecular portraits of human breast tumours. *Nature*, 406(6797):747–752.

Pierson, E. and Yau, C. (2015). Dimensionality reduction for zero-inflated single cell gene expression analysis. *Genome Biol.*, 16(241).

Ramaswamy, S., Tamayo, P., Rifkin, R., Mukherjee, S., Yeang, C.-H., Angelo, M., Ladd, C., Reich, M., Latulippe, E., and Mesirov, J. P. (2001). Multiclass cancer diagnosis using tumor gene expression signatures. *Proceedings of the National Academy of Sciences*, 98(26):15149–15154.

Rapaport, F., Zinovyev, A., Dutreix, M., Barillot, E., and Vert, J.-P. (2007). Classification of microarray data using gene networks. *BMC bioinformatics*, 8(1):35.

Risso, D., Perraudeau, F., Gribkova, S., Dudoit, S., and Vert, J.-P. (2018). A general and flexible method for signal extraction from single-cell rna-seq data. *Nature communications*, 9(1):284.

Robbins, H. and Siegmund, D. (1971). A convergence theorem for non negative almost supermartingales and some applications. In *Optimizing methods in statistics*, pages 233–257. Elsevier.

Salazar-Roa, M. and Malumbres, M. (2017). Fueling the cell division cycle. *Trends in cell biology*, 27(1):69–81.

Schölkopf, B., Burges, C., and Vapnik, V. (1996). Incorporating invariances in support vector learning machines. In *International Conference on Artificial Neural Networks*, pages 47–52. Springer.

Soneson, C. and Robinson, M. D. (2017). Bias, robustness and scalability in differential expression analysis of single-cell RNA-seq data. Technical Report 143289, bioRxiv.

Sørlie, T., Perou, C. M., Tibshirani, R., Aas, T., Geisler, S., Johnsen, H., Hastie, T., Eisen, M. B., van de Rijn, M., Jeffrey, S. S., Thorsen, T., Quist, H., Matese, J. C., Brown, P. O., Botstein, D., Eystein Lønning, P., and Børresen-Dale, A. L. (2001). Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications. *Proc. Natl. Acad. Sci. USA*, 98(19):10869–10874.

Sørlie, T., Tibshirani, R., Parker, J., Hastie, T., Marron, J., Nobel, A., Deng, S., Johnsen, H., Pesich, R., Geisler, S., Demeter, J., Perou, C., Lnning, P., Brown, P., Brresen-Dale, A., and Botstein, D. (2003). Repeated observation of breast tumor subtypes in independent gene expression data sets. *Proc. Natl. Acad. Sci. USA*, 100(14):8418–8423.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.

Tasic, B., Menon, V., Nguyen, T. N., Kim, T. K., Jarsky, T., Yao, Z., Levi, B., Gray, L. T., Sorensen, S. A., Dolbeare, T., Bertagnolli, D., Goldy, J., Shapovalova, N., Parry, S., Lee, C., Smith, K., Bernard, A., Madisen, L., Sunkin, S. M., Hawrylycz, M., Koch, C., and Zeng, H. (2016). Adult mouse cortical cell taxonomy revealed by single cell transcriptomics. *Nat. Neurosci.*, 19(2):335–346.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288.

van de Vijver, M. J., He, Y. D., van't Veer, L. J., Dai, H., Hart, A. A. M., Voskuil, D. W., Schreiber, G. J., Peterse, J. L., Roberts, C., Marton, M. J., Parrish, M., Atsma, D., Witteveen, A., Glas, A., Delahaye, L., van der Velde, T., Bartelink, H., Rodenhuis, S., Rutgers, E. T., Friend, S. H., and Bernards, R. (2002). A gene-expression signature as a predictor of survival in breast cancer. *N. Engl. J. Med.*, 347(25):1999–2009.

van der Maaten, L., Chen, M., Tyree, S., and Weinberger, K. Q. (2013). Learning with marginalized corrupted features. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, number 28 in JMLR Proceedings, pages 410–418. JMLR.org.

Villani, A.-C., Satija, R., Reynolds, G., Sarkizova, S., Shekhar, K., Fletcher, J., Griesbeck, M., Butler, A., Zheng, S., and Lazo, S. (2017). Single-cell RNA-seq reveals new types of human blood dendritic cells, monocytes, and progenitors. *Science*, 356(6335):eaah4573.

Wager, S., Fithian, W., Wang, S., and Liang, P. S. (2014). Altitude training: Strong bounds for single-layer dropout. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q., editors, *Adv. Neural. Inform. Process Syst.*, pages 100–108. Curran Associates, Inc.

Wager, S., Wang, S., and Liang, P. S. (2013). Dropout training as adaptive regularization. In Burges, C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K., editors, *Adv. Neural. Inform. Process Syst.*, volume 26, pages 351–359. Curran Associates, Inc.

Zeisel, A., Manchado, A. B. M., Codeluppi, S., Lonnerberg, P., La Manno, G., Jureus, A., Marques, S., Munguba, H., He, L., Betsholtz, C., Rolny, C., Castelo-Branco, G., Hjerling-Leffler, J., and Linnarsson, S. (2015). Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. *Science*, 347(6226):1138–42.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the Elastic Net. *J. R. Stat. Soc. Ser. B*, 67:301–320.