

Learning interpretable latent autoencoder representations with annotations of feature sets

Sergei Rybakov^{1 2} Mohammad Lotfollahi^{1 3} Fabian J. Theis^{1 2} F. Alexander Wolf^{1 4}

Abstract

Existing methods for learning latent representations for single-cell RNA-seq data are based on autoencoders and factor models. However, representations learned by autoencoders are hard to interpret and representations learned by factor models have limited flexibility. Here, we introduce a framework for learning interpretable autoencoders based on regularized linear decoders. It decomposes variation into interpretable components using prior knowledge in the form of annotated feature sets obtained from public databases. Through this, it provides an alternative to enrichment techniques and factor models for the task of explaining observed variation with biological knowledge. Benchmarking our model on two single-cell RNA-seq datasets, we demonstrate how our model outperforms an existing factor model regarding scalability while maintaining interpretability.

1. Introduction

Advances in single-cell technologies enabled constructing large cell atlases cells across different tissues and species (Regev et al., 2017; The Tabula Muris Consortium et al., 2019). In recent years, machine learning methods have been proposed to learn a compact latent representation to address gene expression denoising and data integration (Lopez et al., 2018; Eraslan et al., 2019), and perturbation modeling (Lotfollahi et al., 2019b;a). However, current methods are not able to incorporate prior-knowledge into their learning algorithms. This work focuses on representation learning by

exploiting prior-knowledge for single-cell data. Differences in gene expression between cells can be decomposed into observed and unobserved factors. These factors can include non-biological factors such as batch effect, or biological factors, which can often be related to existing knowledge about biological processes and pathways. Recently, (Svensson et al., 2020) introduced an interpretable factor model in the form of a linear autoencoder, however, these authors do neither include prior knowledge, nor aim at explaining variation using prior knowledge.

Among existing models, only the factorial single-cell latent variable model (Buettner et al., 2017) can jointly infer factors that capture different sources of single-cell transcriptome variations, including i) variation in expression attributable to pre-annotated sparse gene sets representing biological knowledge, ii) effects due to additional sparse factors that are meant to explain biological effects, and iii) dense factors that are expected to affect the expression profile of the majority of genes, and often represent technical confounders. This model also allows the assignment of genes to each annotated factor to be refined in a data-driven manner. By that, it offers a powerful alternative to traditional enrichment techniques, which are used to contextualize differential expression signatures with biological knowledge.

In f-scLVM, deterministic approximate Bayesian inference based on variational methods is used to approximate the posterior over all random variables of the model. The f-scLVM python package (slalom) is a custom implementation of the variational Bayesian scheme developed for the model, and we use it as main reference for our benchmarks. It is important to note that the variational Bayesian formulation of the method affects its scalability, and complicates the inference of values of the latent factors for new and out-of-sample data points after the model was already trained.

Here, we present a scalable alternative to f-scLVM to learn latent representations of single-cell RNA-seq data that exploit prior knowledge such as Gene Ontology, resulting in interpretable factors. Our frequentist alternative also allows easier inference of the latent factors for out-of-sample data.

¹Institute of Computational Biology, Helmholtz Center Munich, Neuherberg, Germany ²Department of Mathematics, Technical University of Munich, Munich, Germany ³School of Life Sciences Weihenstephan, Technical University of Munich, Munich, Germany ⁴Cellarity Inc., Cambridge, MA, USA. Correspondence to: Sergei Rybakov <sergei.rybakov@helmholtz-muenchen.de>, Fabian J. Theis <fabian.theis@helmholtz-muenchen.de>, F. Alexander Wolf <alex.wolf@helmholtz-muenchen.de>.

Learning interpretable latent representations with annotations of feature sets

2. Model

Our autoencoder can be formulated as:

$$Y \approx f_{\theta}(Y)W^T \quad (1)$$

Where $f_{\theta}(Y)$ is a neural network with learnable parameters θ . This network takes Y , the gene expression matrix of size $N \times G$, as an input and produces a matrix of factor loadings with size $N \times K$. W in the formula above is a learnable parameter matrix of size $G \times K$. A column of W corresponds to weights for all genes for a given factor. Analogously, a row of $f_{\theta}(Y)$ corresponds to loadings of all factors for a given cell. Also, note that the model has the nonlinear encoder $f_{\theta}(Y)$ and the linear decoder zW^T .

After the training phase, the encoder $f_{\theta}(Y)$ can also be used to project out-of-sample data to the learned latent space.

If there are observed covariates X , i.e. features that are known prior to training, such as batch or cell type. They can be incorporated additively into the model 1 as

$$Y \approx f_{\theta}(Y)W^T + d_{\phi_1}(e_{\phi_2}(X)) \quad (2)$$

Where $e_{\phi_2}(X)$ is an encoder network taking covariates matrix X and $d_{\phi_1}(\cdot)$ is a decoder network. In particular the linear model can be used in place of the decoder and encoder for the covariates X , that is $d_{\phi_1}(e_{\phi_2}(X)) = X\Phi_1^T$.

Only the form 1 (the model without observed covariates) will be discussed further.

There are three kinds of factors in the model.

1. **Annotated factors** correspond to gene sets from pathway databases, such as MSigDB (Subramanian, 2005) or Reactome (Jassal et al., 2019). For these factors we enforce structured sparsity informed by the gene sets.
2. **Sparse unannotated factors** represent biologically meaningful factors that don't have annotations. These factors are assumed to be generically sparse.
3. **Dense factors** correspond to effects on the expression of large numbers of genes, no sparsity is enforced for this type.

To model annotated and unannotated factors in W , we need to introduce the right kind of structured sparsity regularization for the columns of W .

The decomposition 1 is fitted to Y using regularized L2 loss

$$L(\theta, W) = \frac{1}{N} \sum_{n=1}^N \|y_n - f_{\theta}(y_n)W^T\|_2^2 + \frac{\lambda_0}{N} \sum_{n=1}^N \|f_{\theta}(y_n)\|_2^2 + R_{\lambda_1, \lambda_2, \lambda_3}(W) \quad (3)$$

Where λ_{0-3} are regularization hyperparameters, the term $R_{\lambda_1, \lambda_2, \lambda_3}(W)$ is a sparsity inducing regularization function, and y_n denotes the n -th row of the data matrix Y (n -th cell). $L(\theta, W)$ can be optimized using the mini-batch stochastic gradient descent algorithm.

$R_{\lambda_1, \lambda_2, \lambda_3}(W)$ in 3 consists of two additive terms

$$R_{\lambda_1, \lambda_2, \lambda_3}(W) = R_{\lambda_1, \lambda_2}^1(W) + \lambda_3 R^2(W) \quad (4)$$

The first term $R_{\lambda_1, \lambda_2}^1(W)$ induces structured sparsity on the level of individual genes in each factor (individual elements in each column of W).

$$R_{\lambda_1, \lambda_2}^1(W) = \lambda_1 \sum_{k_1} \|W_{:,k} \odot (1 - I_{:,k})\|_1 + \lambda_2 \sum_{k_2} \|W_{:,k}\|_1 \quad (5)$$

Where k_1, k_2 correspond to annotated factors and sparse unannotated factors respectively. $W_{:,k} \odot (1 - I_{:,k})$ means Hadamard (element-wise) product between the k -th column of the factor weights matrix W and the k -th column of the binary annotation matrix I . The matrix I , which is used for the annotated factors, can be formed from pathway databases, such as Reactome or MSigDB, with $I_{g,k} = 1$ if the gene g is present in the pathway k and $I_{g,k} = 0$ otherwise. This means that $\|W_{:,k} \odot (1 - I_{:,k})\|_1$ equals to the sum of the absolute values of weights for genes in the factor k that are inactive in the annotation of the factor k (genes that have $I_{g,k} = 0$).

The second regularization term $R^2(W)$ in 4 is responsible for deactivating unneeded factors. We use group lasso for this.

$$R^2(W) = \sum_k \|W_{:,k}\|_2 \quad (6)$$

The group lasso ensures that all genes in a factor (a column of the matrix W) are either included or excluded from the model together. This penalty exploits the non-differentiability of $\|W_{:,k}\|_2$ at $W_{:,k} = 0$; setting whole columns of W to exactly 0.

To minimize 3, the stochastic proximal gradient algorithm is used. This algorithm is employed here because the ordinary stochastic gradient descent algorithm doesn't account for the points of non-differentiability in $R_{\lambda_1, \lambda_2, \lambda_3}(W)$; thus, it can't enforce the required structured sparsity.

Lets denote the part of the objective function $L(\theta, W)$ (3) without the regularization $R_{\lambda_1, \lambda_2, \lambda_3}(W)$ by $F(\theta, W)$. Then, to minimize the objective function 3, we use the update scheme

$$\begin{aligned} \theta^{(t+1)} &= \theta^{(t)} - \eta \nabla_{\theta} \hat{F}(\theta, W) \\ W^{(t+1)} &= \eta R_{\lambda_1, \lambda_2, \lambda_3}^{\text{Prox}}(W^{(t)} - \eta \nabla_W \hat{F}(\theta, W)) \end{aligned} \quad (7)$$

Learning interpretable latent representations with annotations of feature sets

Where $\hat{F}(\theta, W)$ denote the function $F(\theta, W)$ calculated for a mini-batch of samples, η is a learning rate. $\eta R_{\lambda_1, \lambda_2, \lambda_3}^{\text{Prox}}(\cdot)$ denotes the proximal operator of $R_{\lambda_1, \lambda_2, \lambda_3}(W)$

$$\eta R_{\lambda_1, \lambda_2, \lambda_3}^{\text{Prox}}(V) = \arg \min_L \left[\frac{1}{2} \|L - V\|_F^2 + \eta R_{\lambda_1, \lambda_2, \lambda_3}(L) \right] \quad (8)$$

Using the result from (Yu, 2013), it can be shown that the proximal operator 8 has a closed-form expression, which is a composition of the closed-form expressions for the proximal operators corresponding to $R_{\lambda_1, \lambda_2}^1(W)$ and $\lambda_3 R^2(W)$.

3. Results

To validate the model, we considered a dataset where some sources of variation are known. In (Kang et al., 2017) 8 Lupus patients were stimulated with interferon (IFN) β . We expect to see upregulation of the pathways related to interferon signaling.

The autoencoder with only annotated factors was trained on the dataset from the paper with 1k highly variable genes selected, using the Reactome database for the annotated factors. The scatter plots of factor loadings of the annotated factors corresponding to the gene sets from Reactome “INTERFERON_SIGNALING” and “SIGNALING_BY_THE_B_CELL_RECEPTOR_BCR” (Figure 1a) show clear separation of stimulated cells from control cells.

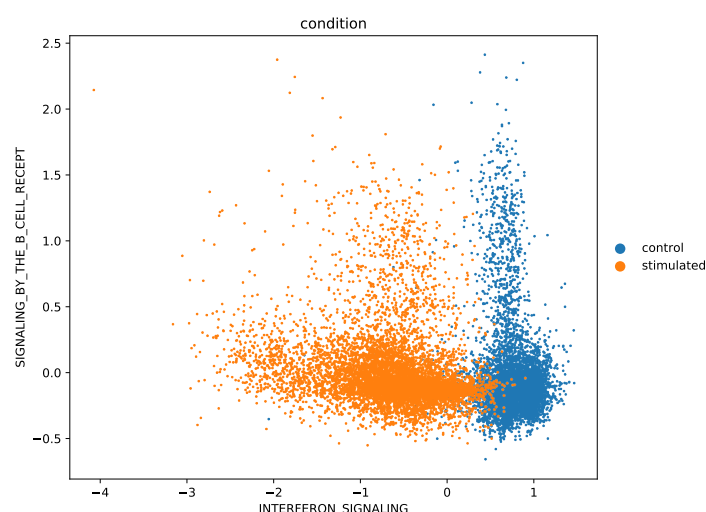
Next, we sought to analyze celltype-specific pathways by using the pathway related to B cells. We can also see a clear separation of B cells from the rest (Figure 1b).

Also we compared our results with f-scLVM. The python implementation of f-scLVM (slalom) was used for the comparison. In the table below the time to train both models is given.

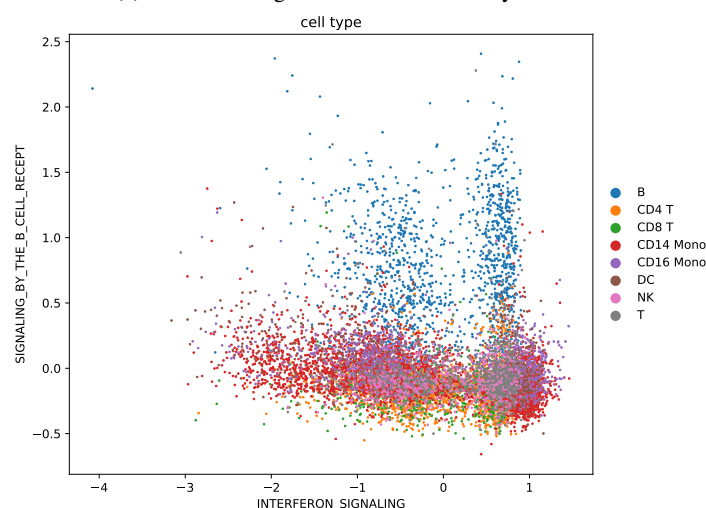
	Kang17 13.5k cells	Macosko15 13.5k cells
f-scLVM (slalom)	2203	1629
Interpretable AE (this work)	8	7

Table 1. Runtime comparison in minutes.

The time is provided for the 13.5k cell subset of the retina dataset from Macosko et al. (2015) and for the dataset from Kang et al. (2017). For both datasets the genes that are not in the annotations from Reactome were filtered out, and also highly variable genes were selected to limit the number of genes to about 1k. It can be seen from the table that slalom requires 270 times as long as the interpretable AE. More importantly, while it is still manageable for datasets of this size, it becomes near impossible to use slalom for much larger datasets, which are common in single cell genomics.



(a) Factor loadings for 2 terms colored by condition



(b) Factor loadings for 2 terms colored by cell type

Figure 1. Factor loadings in the interpretable AE.

In order to check how well both models explain variation in the Kang et al. (2017) dataset, several terms related to interferon beta were selected, and their loadings were used individually to train uni-variate binary logistic regression on the experimental condition (stimulated vs. control) as class label (Figure 2). Two setups of f-scLVM were used for comparison: f-scLVM without any dense and sparse unannotated factors and f-scLVM with 3 dense factors (default setting). The autoencoder provides better accuracy, and thus better separation, across all factors except for “ANTIVIRAL_MECHANISM_BY_IFN_STIMULATED_GENES”.

The figure 2 shows the resulting accuracies for training data of logistic regressions trained with the loadings of the individual factors. Two setups of f-scLVM were used for comparison - f-scLVM without any dense and sparse unan-

Learning interpretable latent representations with annotations of feature sets

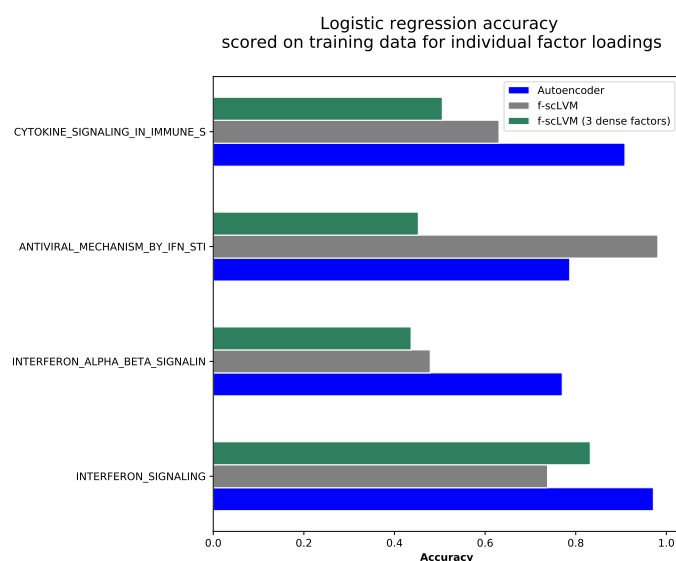


Figure 2. Accuracies of logistic regressions trained on loadings of individual factors vs condition (control, stimulated) for 3 models - autoencoder, f-scLVM without dense and sparse unannotated factors, f-scLVM with 3 dense factors (default setting).

notated factors and f-scLVM with 3 dense factors (standard settings). The autoencoder provides better accuracy, and thus better separation, across all factors except for "ANTIVIRAL_MECHANISM_BY_IFN_STIMULATED_GENES".

It is interesting that for both autoencoder model and f-scLVM with three dense factors the term which gives the highest accuracy (the best separating term) is "INTERFERON_SIGNALING" (see Figure 2 for the accuracies of classification, Figure 1a for the visualization). However, f-scLVM without dense factors selects "ANTIVIRAL_MECHANISM_BY_IFN_STIMULATED_GENES" for the separation (Figure 2). It is not clear why training f-scLVM with dense factors, which should account for confounding sources of variation, leads to significant changes in the loadings of the factors which should meaningfully explain biological variation in the data.

Next, we applied our model on a recent, comprising of immune and epithelial cells collected via bronchoalveolar lavage from healthy controls, and patients with moderate and severe COVID-19 Liao et al. (2020). The 10 most important pathways, explaining the variation in the dataset, by L2 norm of their parameter vectors (columns of W in 1) are

1. IMMUNE_SYSTEM
2. METABOLISM_OF_PROTEINS
3. HEMOSTASIS
4. TCA_CYCLE_AND_RESPIRATORY_ELECTRON_TRANSPORT
5. ADAPTIVE_IMMUNE_SYSTEM
6. 3_UTR_MEDIATED_TRANSLATIONAL_REGULATION
7. TRANSLATION

8. METABOLISM_OF_LIPIDS_AND_LIPOPROTEINS
9. INTERFERON_SIGNALING
10. MRNA_PROCESSING

In order to validate interpretability of the learned latent space of pathways, we again investigated factor loadings of the "INTERFERON_SIGNALING" term.

The figure 3 shows the separation of control and activated immune cells in moderate and severe COVID-19 in "INTERFERON_SIGNALING" term.

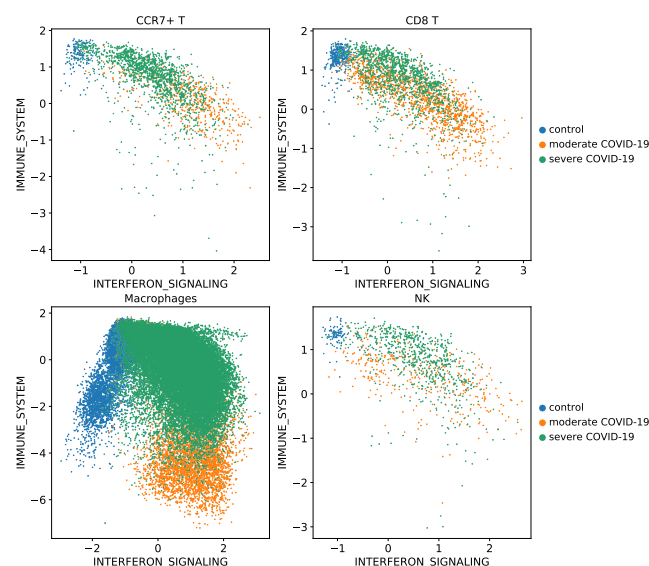


Figure 3. Factor loadings of 2 terms for the four cell types from the COVID-19 dataset

4. Discussion and future work

We presented an end-to-end trained autoencoder that can exploit biological knowledge from various databases (Subramanian, 2005; Jassal et al., 2019) to learning interpretable representations of data.

The current model can be extended by learning a more powerful decoder. This requires adding non-linearity to the decoder to learn more complex representations. However, non-linear functions can not easily be incorporated into to model while preserving feature factorization. Gradient-based feature importance methods have been shown to be applicable in this context for image data (Erion et al., 2019). We leave further extensions in this direction for future work.

The code for the model and the accompanying data can be obtained from <https://github.com/theislab/InterpretableAutoencoders>.

Learning interpretable latent representations with annotations of feature sets

References

- The factorial single-cell latent variable model (slalom). <https://github.com/bioFAM/slalom>.
- Buettner, F., Pratanwanich, N., McCarthy, D., Marioni, J., and Stegle, O. f-sclvm: Scalable and versatile factor analysis for single-cell rna-seq. *Genome Biology*, 18, 12 2017. doi: 10.1186/s13059-017-1334-8.
- Eraslan, G., Simon, L. M., Mircea, M., Mueller, N. S., and Theis, F. J. Single-cell RNA-seq denoising using a deep count autoencoder. *Nature Communications*, 10 (1):390, January 2019. ISSN 2041-1723. doi: 10.1038/s41467-018-07931-2. URL <https://www.nature.com/articles/s41467-018-07931-2>. Number: 1 Publisher: Nature Publishing Group.
- Erion, G. G., Janizek, J. D., Sturmfels, P., Lundberg, S. M., and Lee, S.-I. Learning explainable models using attribution priors. *ArXiv*, abs/1906.10670, 2019.
- Jassal, B., Matthews, L., Viteri, G., Gong, C., Lorente, P., Fabregat, A., Sidiropoulos, K., Cook, J., Gillespie, M., Haw, R., Loney, F., May, B., Milacic, M., Rothfels, K., Sevilla, C., Shamovsky, V., Shorser, S., Varusai, T., Weiser, J., and D'Eustachio, P. The reactome pathway knowledgebase. *Nucleic acids research*, 48, 11 2019. doi: 10.1093/nar/gkz1031.
- Kang, H. M., Subramaniam, M., Targ, S., Nguyen, M., Maliskova, L., McCarthy, E. A., Wan, E., Wong, S. L., Byrnes, L. E., Lanata, C. M., Gate, R. E., Mostafavi, S., Marson, A., Zaitlen, N., Criswell, L. A., and Ye, C. J. Multiplexed droplet single-cell rna-sequencing using natural genetic variation. In *Nature Biotechnology*, 2017.
- Liao, M., Liu, Y., Yuan, J., Wen, Y., Xu, G., Zhao, J., Cheng, L., Li, J., Wang, X., Wang, F., Liu, L., Amit, I., Zhang, S., and Zhang, Z. Single-cell landscape of bronchoalveolar immune cells in patients with covid-19. *Nature Medicine*, 26, 06 2020. doi: 10.1038/s41591-020-0901-9.
- Lopez, R., Regier, J., Cole, M., Jordan, M., and Yosef, N. Bayesian Inference for a Generative Model of Transcriptome Profiles from Single-cell RNA Sequencing. preprint, Bioinformatics, March 2018. URL <http://biorxiv.org/lookup/doi/10.1101/292037>.
- Lotfollahi, M., Naghipourfar, M., Theis, F. J., and Wolf, F. A. Conditional out-of-sample generation for unpaired data using trVAE. *arXiv:1910.01791 [cs, eess, q-bio, stat]*, October 2019a. URL <http://arxiv.org/abs/1910.01791>. arXiv: 1910.01791.
- Lotfollahi, M., Wolf, F. A., and Theis, F. J. scGen predicts single-cell perturbation responses. *Nature Methods*, 16(8):715–721, August 2019b. ISSN 1548-7091, 1548-7105. doi: 10.1038/s41592-019-0494-8. URL <http://www.nature.com/articles/s41592-019-0494-8>. tex.ids: lotfollahiScGen-PredictsSinglecell2019a Number: 8 Publisher: Nature Publishing Group.
- Macosko, E., Basu, A., Satija, R., Nemesh, J., Shekhar, K., Goldman, M., Tirosh, I., Bialas, A., Kamitaki, N., Martersteck, E., Trombetta, J., Weitz, D., Sanes, J., Shalek, A., Regev, A., and Mccarroll, S. Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets. *Cell*, 161:1202–1214, 05 2015. doi: 10.1016/j.cell.2015.05.002.
- Regev, A., Teichmann, S. A., Lander, E. S., Amit, I., Benoist, C., Birney, E., Bodenmiller, B., Campbell, P., Carninci, P., Clatworthy, M., et al. Science forum: the human cell atlas. *Elife*, 6:e27041, 2017.
- Subramanian, A. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *proc natl acad sci u s a. Proceedings of the National Academy of Sciences*, 102:15545–15550, 10 2005.
- Svensson, V., Gayoso, A., Yosef, N., and Pachter, L. Interpretable factor models of single-cell rna-seq via variational autoencoders. *Bioinformatics (Oxford, England)*, 03 2020. doi: 10.1093/bioinformatics/btaa169.
- The Tabula Muris Consortium, Pisco, A. O., McGeever, A., Schaum, N., Karkanias, J., Neff, N. F., Darmanis, S., Wyss-Coray, T., and Quake, S. R. A Single Cell Transcriptomic Atlas Characterizes Aging Tissues in the Mouse. preprint, Cell Biology, June 2019. URL <http://biorxiv.org/lookup/doi/10.1101/661728>.
- Yu, Y. On decomposing the proximal map. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'13, pp. 91–99, Red Hook, NY, USA, 2013. Curran Associates Inc.