Optimizers manuscript

This manuscript (permalink) was automatically generated from greenelab/optimizer-manuscript@c5471d9 on May 1, 2023.

Authors

Jake Crawford

© 0000-0001-6207-0782 · ♥ jjc2718 · ♥ jjc2718

Genomics and Computational Biology Graduate Group, Perelman School of Medicine, University of Pennsylvania, Philadelphia, PA, USA

• Casey S. Greene ✓

Department of Biochemistry and Molecular Genetics, University of Colorado School of Medicine, Aurora, CO, USA; Center for Health AI, University of Colorado School of Medicine, Aurora, CO, USA

■ — Correspondence possible via <u>GitHub Issues</u> or email to Casey S. Greene <casey.s.greene@cuanschutz.edu>.

Abstract

Introduction

Gene expression profiles are widely used to classify samples or patients into relevant groups or categories, both preclinically [1,2] and clinically [3,4]. To extract informative gene features and to perform classification, a diverse array of algorithms exist, and different algorithms perform well across varying datasets and tasks [1]. Even within a given model class, multiple optimization methods can often be applied to find well-performing model parameters or to optimize a model's loss function. One commonly used example is logistic regression. The widely used scikit-learn Python package for machine learning [5] provides two modules for fitting logistic regression classifiers:

LogisticRegression, which uses the liblinear coordinate descent method [6] to find parameters that optimize the logistic loss function, and SGDClassifier, which uses stochastic gradient descent [7] to optimize the same loss function.

Using scikit-learn, we compared the liblinear (coordinate descent) and SGD optimization techniques for prediction of driver mutation status in tumor samples, across a wide variety of genes implicated in cancer initiation and development [8]. We applied LASSO (L1-regularized) logistic regression, and tuned the strength of the regularization to compare model selection between optimizers. We found that across a variety of models (i.e. varying regularization strengths), the training dynamics of the optimizers were considerably different: models fit using liblinear tended to perform best at fairly high regularization strengths (100-1000 nonzero features in the model) and overfit easily with low regularization strengths. On the other hand, models fit using stochastic gradient descent tended to perform best at fairly low regularization strengths (10000+ nonzero features in the model), and overfitting was uncommon.

Our results caution against viewing optimizer choice as a "black box" component of machine learning modeling. The observation that LASSO logistic regression models fit using SGD tended to perform best for low levels of regularization, across diverse driver genes, runs counter to conventional wisdom in machine learning for high-dimensional data which generally states that explicit regularization and/or feature selection is necessary. Comparing optimizers/model implementations directly is rare in applications of machine learning for genomics, and our work shows that this choice can affect generalization and interpretation properties of the model significantly. Based on our results, we recommend considering the appropriate optimization approach carefully based on the goals of each individual analysis.

Methods

Results

For each of the 125 driver genes from the Vogelstein et al. 2013 paper, we trained models to predict mutation status (presence or absence) from RNA-seq data, derived from the TCGA Pan-Cancer Atlas. For each optimizer, we trained LASSO logistic regression models across a variety of regularization parameters (see Methods for parameter range details), for 4 cross-validation splits x 2 replicates (random seeds) for a total of 8 different models per parameter. Cross-validation splits were stratified by cancer type.

Previous work has shown that pan-cancer classifiers of Ras mutation status are accurate and biologically informative [9]. As model complexity increases (more nonzero coefficients) for the liblinear optimizer, we observe that performance increases then decreases, corresponding to overfitting for high model complexities/numbers of nonzero coefficients (Figure 1A). On the other hand, for the SGD optimizer, we observe an increase in performance as model complexity increases, with models having no nonzero coefficients performing the best (Figure 1B). In this case, top performance for SGD (the largest bin, i.e. furthest right on the x-axis) is slightly worse than top performance for liblinear (the third smallest bin): we observed a mean test AUPR of 0.618 for SGD vs. mean AUPR of 0.688 for liblinear. As model complexity varies, similar performance trends tend to hold across a variety of driver genes in the Vogelstein dataset, and for a variety of approaches to quantifying model complexity (see Supplementary Data).

To determine if the relative performance improvement with liblinear tends to hold across the genes in the Vogelstein dataset at large, we compared performance for the best-performing models for each gene, between optimizers. Figure 1C shows the distribution of differences in performance across genes. The distribution is generally shifted to the right, suggesting that liblinear generally tends to outperform SGD. We saw that for 71/84 genes, performance for the best-performing model was better using liblinear than SGD, and for the other 13 genes performance was better using SGD.

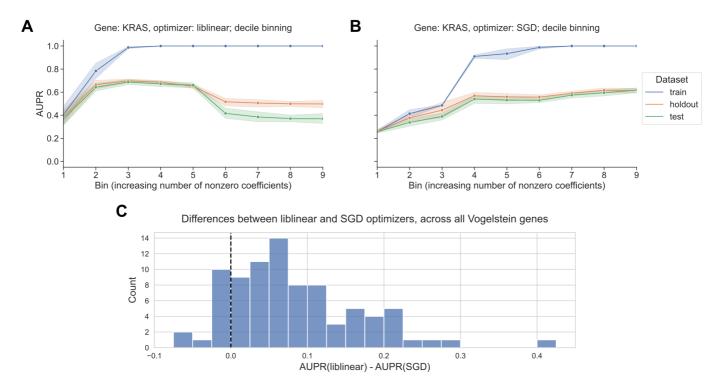
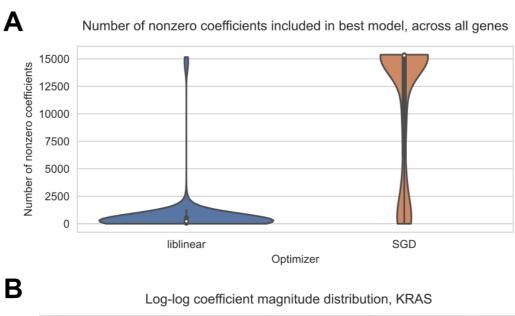


Figure 1: A. Performance vs. model complexity (number of nonzero coefficients) for KRAS mutation status prediction, for liblinear optimizer. Bins are derived from deciles of coefficient count distribution across optimizers; additional detail in Methods. "Holdout" dataset is used in panel C and following figures for best-performing model selection, "test"

data is completely held out from model selection and used for evaluation in panel C and following figures. **B.**Performance vs. model complexity (number of nonzero coefficients) for KRAS mutation status prediction, for SGD optimizer. **C.** Distribution of performance difference between best-performing model for liblinear and SGD optimizers, across all 84 genes in Vogelstein driver gene set. Positive numbers on the x-axis indicate better performance using liblinear, and negative numbers indicate better performance using SGD.

We next sought to determine whether there was a difference in the magnitudes of coefficients in the models resulting from the different optimization schemes. Following up on the trend in Figure 1, where we saw that the best-performing SGD model had many nonzero coefficients, we also see that in general across all genes, the best-performing SGD models tend to be bimodal, sometimes having few nonzero coefficients but often having many/all nonzero coefficients (Figure 2A). By contrast, the liblinear models are almost always much sparser with fewer than 2500 nonzero coefficients, out of ~16100 total input features.

Despite the SGD models performing best with many nonzero coefficients, it could be the case that many of the coefficients could be "effectively" 0, or uninformative to the final model. However, Figure 2B provides evidence that this is not the case, with most coefficients in the best-performing KRAS mutation prediction model using SGD being considerably larger than the coefficients in the best-performing model using liblinear, and very few close to 0. This emphasizes that the different optimization methods result in fundamentally different models, relying on different numbers of features with nonzero coefficients in different magnitudes, rather than converging to similar models.



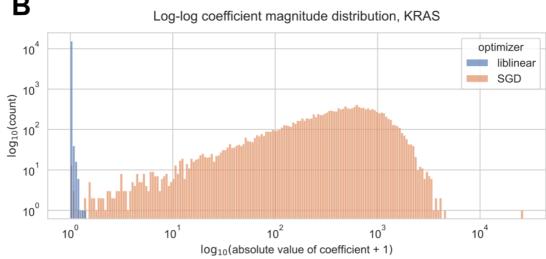


Figure 2: A. Distribution across genes of the number of nonzero coefficients included in best-performing LASSO logistic regression models. Violin plot density estimations are clipped at the ends of the observed data range, and boxes show the median/IQR. **B.** Distribution of coefficient magnitudes for a single KRAS mutation prediction model (random seed 42,

first cross-validation split), colored by optimizer. The x-axis shows the base-10 logarithm of the absolute value of each coefficient + 1 (since some coefficients are exactly 0), and the y-axis shows the base-10 log of the count of coefficients in each bin. Other random seeds and cross-validation splits are similar.

Discussion

Our results suggest that even for the same model, LASSO logistic regression, optimizer choice can affect model selection and performance. Existing gene expression prediction benchmarks and pipelines typically use a single model implementation (and thus a single optimizer). To our knowledge, the phenomenon we observed with SGD has not been documented in other applications of ML to genomic or transcriptomic data. In the broader machine learning research community, however, similar patterns have been observed for both linear models and deep neural networks (e.g. [10,11]). This is often termed "benign overfitting": the idea that "overfit" models, in the sense that they fit the training data perfectly and perform worse on the test data, can still outperform models that do not fit the training data as well or that have stronger explicit regularization. Benign overfitting has been observed with, and attributed to, optimization using SGD, which is thought to provide a form of implicit regularization [12,13].

We recommend thinking critically about optimizer choice, but this can be challenging for users that are inexperienced with machine learning or unfamiliar with how certain models are fit under the hood. For example, R's glmnet package uses a cyclical coordinate descent algorithm to fit logistic regression models [14], which would presumably behave similarly to liblinear, but this is somewhat opaque in the glmnet documentation itself. LASSO logistic regression is a convex optimization problem, meaning there is a single unique optimum of the loss function in contrast to more complex models such as neural networks, but this optimum can be computationally intensive to find in practice and there is no closed-form solution [15]. Increased transparency and documentation in popular machine learning packages with respect to optimization, especially for models that are challenging to fit, would benefit new and unfamiliar users.

Similar to what we see in our SGD-optimized models, there exist other problems in gene expression analysis where using all available features is better than using a subset. For example, using the full gene set improves correlations between preclinical cancer models and their tissue of origin, as compared to selecting genes based on variability or tissue-specificity [16]. On the other hand, when predicting cell line viability from gene expression profiles, selecting features by Pearson correlation improves performance over using all features, similar to our liblinear classifiers [17]. An avenue of future work for our SGD classifiers would be to interpret the coefficients and compare them systematically to the coefficients found using liblinear. It could be useful to understand if the two optimization methods emphasize the same pathways or functional gene sets, or if there are patterns to which driver mutations perform better with more/fewer nonzero coefficients.

References

1. The ability to classify patients based on gene-expression data varies by algorithm and performance metric

Stephen R Piccolo, Avery Mecham, Nathan P Golightly, Jérémie L Johnson, Dustin B Miller *PLOS Computational Biology* (2022-03-11) https://doi.org/gr43qd

DOI: 10.1371/journal.pcbi.1009926 · PMID: 35275931 · PMCID: PMC8942277

2. Supervised learning is an accurate method for network-based gene classification

Renming Liu, Christopher A Mancuso, Anna Yannakopoulos, Kayla A Johnson, Arjun Krishnan *Bioinformatics* (2020-04-14) https://doi.org/gmvnfc

DOI: 10.1093/bioinformatics/btaa150 · PMID: 32129827 · PMCID: PMC7267831

3. Supervised Risk Predictor of Breast Cancer Based on Intrinsic Subtypes

Joel S Parker, Michael Mullins, Maggie CU Cheang, Samuel Leung, David Voduc, Tammi Vickery, Sherri Davies, Christiane Fauron, Xiaping He, Zhiyuan Hu, ... Philip S Bernard *Journal of Clinical Oncology* (2009-03-10) https://doi.org/c2688w

DOI: <u>10.1200/jco.2008.18.1370</u> · PMID: <u>19204204</u> · PMCID: <u>PMC2667820</u>

4. Prediction of adjuvant chemotherapy benefit in endocrine responsive, early breast cancer using multigene assays

Kathy S Albain, Soonmyung Paik, Laura van't Veer

The Breast (2009-10) https://doi.org/bp4rtw

DOI: <u>10.1016/s0960-9776(09)70290-5</u> · PMID: <u>19914534</u>

5. Scikit-learn: Machine Learning in Python

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, ... Édouard Duchesnay

Journal of Machine Learning Research (2011) http://jmlr.org/papers/v12/pedregosa11a.html

6. LIBLINEAR: A Library for Large Linear Classification

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, Chih-Jen Lin *Journal of Machine Learning Research* (2008) http://jmlr.org/papers/v9/fan08a.html

7. Online Learning and Stochastic Approximations

Leon Bottou

(1998) https://wiki.eecs.yorku.ca/course archive/2012-13/F/6328/ media/bottou-onlinelearning-98.pdf

8. Cancer Genome Landscapes

B Vogelstein, N Papadopoulos, VE Velculescu, S Zhou, LA Diaz, KW Kinzler *Science* (2013-03-28) https://doi.org/6rg

DOI: 10.1126/science.1235122 · PMID: 23539594 · PMCID: PMC3749880

9. Machine Learning Detects Pan-cancer Ras Pathway Activation in The Cancer Genome Atlas

Gregory P Way, Francisco Sanchez-Vega, Konnor La, Joshua Armenia, Walid K Chatila, Augustin Luna, Chris Sander, Andrew D Cherniack, Marco Mina, Giovanni Ciriello, ... Armaz Mariamidze *Cell Reports* (2018-04) https://doi.org/gfspsb

DOI: 10.1016/j.celrep.2018.03.046 · PMID: 29617658 · PMCID: PMC5918694

10. Benign overfitting in linear regression

Peter L Bartlett, Philip M Long, Gábor Lugosi, Alexander Tsigler

Proceedings of the National Academy of Sciences (2020-04-24) https://doi.org/gjgsxq
DOI: 10.1073/pnas.1907378117 • PMID: 23332161 • PMCID: PMC7720150

11. **Understanding deep learning requires rethinking generalization** Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, Oriol Vinyals *arXiv* (2017-02-28) https://arxiv.org/abs/1611.03530

12. **Understanding deep learning (still) requires rethinking generalization** Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, Oriol Vinyals *Communications of the ACM* (2021-02-22) https://doi.org/gh57fd DOI: 10.1145/3446776

13. **Benign Overfitting of Constant-Stepsize SGD for Linear Regression**Difan Zou, Jingfeng Wu, Vladimir Braverman, Quanquan Gu, Sham Kakade *Proceedings of Thirty Fourth Conference on Learning Theory* (2021-07-21) https://proceedings.mlr.press/v134/zou21a.html

14. **Regularization Paths for Generalized Linear Models via Coordinate Descent**Jerome Friedman, Trevor Hastie, Robert Tibshirani *Journal of Statistical Software* (2010) https://doi.org/bb3d DOI: 10.18637/jss.v033.i01

15. **Efficient L1 Regularized Logistic Regression**Su-In Lee, Honglak Lee, Pieter Abbeel, Andrew Y Ng AAAI 2006 (2006) https://ai.stanford.edu/~pabbeel//pubs/LeeLeeAbbeelNg_el1rlr_AAAI2006.pdf

16. Evaluating cancer cell line and patient-derived xenograft recapitulation of tumor and non-diseased tissue gene expression profiles

Avery S Williams, Elizabeth J Wilk, Jennifer L Fisher, Brittany N Lasseigne *Cold Spring Harbor Laboratory* (2023-04-13) https://doi.org/gr6jr4
DOI: 10.1101/2023.04.11.536431 · PMID: 37090499 · PMCID: PMC10120639

17. Gene expression has more power for predicting <i>in vitro</i> cancer cell vulnerabilities than genomics

Joshua M Dempster, John M Krill-Burger, James M McFarland, Allison Warren, Jesse S Boehm, Francisca Vazquez, William C Hahn, Todd R Golub, Aviad Tsherniak *Cold Spring Harbor Laboratory* (2020-02-24) https://doi.org/ghczbr

DOI: <u>10.1101/2020.02.21.959627</u>