

Jake Crawford dissertation title

This manuscript ([permalink](#)) was automatically generated from [greenelab/jake_dissertation@547acc2](https://greenelab.github.io/jake_dissertation@547acc2) on August 28, 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

✉ — Correspondence possible via [GitHub Issues](#)

Abstract

Chapter 1

- Modeling strategies (copy existing review)
- Machine learning for cancer transcriptomics (add some new text)

Chapter 2: optimization strongly influences model selection in transcriptomic prediction

This chapter has been posted as a preprint on bioRxiv (<https://www.biorxiv.org/content/10.1101/2023.06.26.546586v1>) and submitted for publication at Bioinformatics Advances as “Optimizer’s dilemma: optimization strongly influences model selection in transcriptomic prediction”.

Contributions: I designed and ran the experiments, created the figures, wrote the initial draft of the manuscript, and edited the manuscript. Maria Chikina gave feedback on an initial version of the manuscript, gave guidance on experimental design, and edited the manuscript. Casey S. Greene gave feedback and guidance on experiments, and edited the manuscript.

Abstract

Motivation

Most models can be fit to data using various optimization approaches. While model choice is frequently reported in machine-learning-based research, optimizers are not often noted. We applied two different implementations of LASSO logistic regression implemented in Python’s scikit-learn package, using two different optimization approaches (coordinate descent and stochastic gradient descent), to predict driver mutation presence or absence from gene expression across 84 pan-cancer driver genes. Across varying levels of regularization, we compared performance and model sparsity between optimizers.

Results

After model selection and tuning, we found that coordinate descent (implemented in the `liblinear` library) and SGD tended to perform comparably. `liblinear` models required more extensive tuning of regularization strength, performing best for high model sparsities (more nonzero coefficients), but did not require selection of a learning rate parameter. SGD models required tuning of the learning rate to perform well, but generally performed more robustly across different model sparsities as regularization strength decreased. Given these tradeoffs, we believe that the choice of optimizers should be clearly reported as a part of the model selection and validation process, to allow readers and reviewers to better understand the context in which results have been generated.

Availability and implementation

The code used to carry out the analyses in this study is available at https://github.com/greenelab/pancancer-evaluation/tree/master/01_stratified_classification.

Performance/regularization strength curves for all genes in the Vogelstein et al. 2013 dataset are available at <https://doi.org/10.6084/m9.figshare.22728644>.

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, after tuning the learning rate, models fit using SGD tended to perform well across both higher and lower regularization strengths, and overfitting was less common.

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 well 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 or 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

Data download and preprocessing

To generate binary mutated/non-mutated gene labels for our machine learning model, we used mutation calls for TCGA Pan-Cancer Atlas samples from MC3 [9] and copy number threshold calls from GISTIC2.0 [10]. MC3 mutation calls were downloaded from the Genomic Data Commons (GDC) of the National Cancer Institute, at <https://gdc.cancer.gov/about-data/publications/pancanatlas>.

Thresholded copy number calls are from an older version of the GDC data and are available here: https://figshare.com/articles/dataset/TCGA_PanCanAtlas_Copy_Number_Data/6144122. We removed hypermutated samples, defined as two or more standard deviations above the mean non-silent somatic mutation count, from our dataset to reduce the number of false positives (i.e., non-driver mutations). Any sample with either a non-silent somatic variant or a copy number variation (copy number gain in the target gene for oncogenes and copy number loss in the target gene for tumor

suppressor genes) was included in the positive set; all remaining samples were considered negative for mutation in the target gene.

RNA sequencing data for TCGA was downloaded from GDC at the same link provided above for the Pan-Cancer Atlas. We discarded non-protein-coding genes and genes that failed to map and removed tumors that were measured from multiple sites. After filtering to remove hypermutated samples and taking the intersection of samples with both mutation and gene expression data, 9074 total TCGA samples remained.

Cancer gene set construction

In order to study mutation status classification for a diverse set of cancer driver genes, we started with the set of 125 frequently altered genes from Vogelstein et al. [8] (all genes from Table S2A). For each target gene, in order to ensure that the training dataset was reasonably balanced (i.e., that there would be enough mutated samples to train an effective classifier), we included only cancer types with at least 15 mutated samples and at least 5% mutated samples, which we refer to here as “valid” cancer types. In some cases, this resulted in genes with no valid cancer types, which we dropped from the analysis. Out of the 125 genes originally listed in the Vogelstein et al. cancer gene set, we retained 84 target genes after filtering for valid cancer types.

Classifier setup and optimizer comparison details

We trained logistic regression classifiers to predict whether or not a given sample had a mutational event in a given target gene using gene expression features as explanatory variables. Based on our previous work, gene expression is generally effective for this problem across many target genes, although other -omics types can be equally effective in many cases [11]. Our model was trained on gene expression data (X) to predict mutation presence or absence (y) in a target gene. To control for varying mutation burden per sample and to adjust for potential cancer type-specific expression patterns, we included one-hot encoded cancer type and $\log_{10}(\text{sample mutation count})$ in the model as covariates. Since gene expression datasets tend to have many dimensions and comparatively few samples, we used a LASSO penalty to perform feature selection [12]. LASSO logistic regression has the advantage of generating sparse models (some or most coefficients are 0), as well as having a single tunable hyperparameter which can be easily interpreted as an indicator of regularization strength, or model complexity.

To compare model selection across optimizers, we first split the “valid” cancer types into train (75%) and test (25%) sets. We then split the training data into “subtrain” (66% of the training set) data to train the model on, and “holdout” (33% of the training set) data to perform model selection, i.e. to use to select the best-performing regularization parameter, and the best-performing learning rate for SGD in the cases where multiple learning rates were considered. In each case, these splits were stratified by cancer type, i.e. each split had as close as possible to equal proportions of each cancer type included in the dataset for the given driver gene.

LASSO parameter range selection and comparison between optimizers

The scikit-learn implementations of coordinate descent (in `liblinear` / `LogisticRegression`) and stochastic gradient descent (in `SGDClassifier`) use slightly different parameterizations of the LASSO regularization strength parameter. `liblinear`’s logistic regression solver optimizes the following loss function:

$$\hat{w} = \operatorname{argmin}_w (C \cdot \ell(X, y; w)) + \|w\|_1$$

where $\ell(X, y; w)$ denotes the negative log-likelihood of the observed data (X, y) given a particular choice of feature weights w . `SGDClassifier` optimizes the following loss function:

$$\hat{w} = \operatorname{argmin}_w \ell(X, y; w) + \alpha \|w\|_1$$

which is equivalent with the exception of the LASSO parameter which is formulated slightly differently, as $\alpha = \frac{1}{C}$. The result of this slight difference in parameterization is that `liblinear` C values vary inversely with regularization strength (higher values = less regularization, or greater model complexity) and `SGDClassifier` α values vary directly with regularization strength (lower values = less regularization, or greater model complexity).

For the `liblinear` optimizer, we trained models using C values evenly spaced on a logarithmic scale between $(10^{-3}, 10^7)$; i.e. the output of `numpy.logspace(-3, 7, 21)`. For the SGD optimizer, we trained models using the inverse range of α values between $(10^{-7}, 10^3)$, or `numpy.logspace(-7, 3, 21)`. These hyperparameter ranges were intended to give evenly distributed coverage across genes that included “underfit” models (predicting only the mean or using very few features, poor performance on all datasets), “overfit” models (performing perfectly on training data but comparatively poorly on cross-validation and test data), and a wide variety of models in between that typically included the best fits to the cross-validation and test data.

For ease of visual comparison in our figures, we plot the SGD α parameter directly, and the `liblinear` C parameter inversely (i.e. $\frac{1}{C}$). This orients the x-axes of the relevant plots in the same direction: lower values represent lower regularization strength or higher model complexity, and higher values represent higher regularization strength or lower model complexity, for both optimizers.

SGD learning rate selection

scikit-learn’s `SGDClassifier` provides four built-in approaches to learning rate scheduling: `constant` (a single, constant learning rate), `optimal` (a learning rate with an initial value selected using a heuristic based on the regularization parameter and the data loss, that decreases across epochs), `invscaling` (a learning rate that decreases exponentially by epoch), and `adaptive` (a learning rate that starts at a constant value, which is divided by 5 each time the training loss fails to decrease for 5 straight epochs). The `optimal` learning rate schedule is used by default.

When we compared these four approaches, we used a constant learning rate of 0.0005, and an initial learning rate of 0.1 for the `adaptive` and `invscaling` schedules. We also tested a fifth approach that we called “`constant_search`”, in which we tested a range of constant learning rates in a grid search on a validation dataset, then evaluated the model on the test data using the best-performing constant learning rate by validation AUPR. For the grid search, we used the following range of constant learning rates: {0.000005, 0.00001, 0.00005, 0.0001, 0.0005, 0.001, 0.01}. Unless otherwise specified, results for SGD in the main paper figures used the `constant_search` approach, which performed the best in our comparison between schedulers.

Results

`liblinear` and SGD LASSO models perform comparably, but `liblinear` is sensitive to regularization strength

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), achieving a variety of different levels of model sparsity (Supplementary Figure 4). We repeated model fitting/evaluation across 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 [13]. We first evaluated models for KRAS mutation prediction. As model complexity increases (more nonzero coefficients) for the `liblinear` optimizer, we observed 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 observed consistent performance as model complexity increases, with models having no nonzero coefficients performing comparably to the best (Figure 1B). In this case, top performance for SGD (a regularization parameter of 10^{-1}) is slightly better than top performance for `liblinear` (a regularization parameter of $1 / 3.16 \times 10^2$): we observed a mean test AUPR of 0.722 for SGD vs. mean AUPR of 0.692 for `liblinear`.

To determine how relative performance trends with `liblinear` tend to compare across the genes in the Vogelstein dataset at large, we looked at the difference in performance between optimizers for the best-performing models for each gene (Figure 1C). The distribution is centered around 0 and more or less symmetrical, suggesting that across the gene set, `liblinear` and SGD tend to perform comparably to one another. We saw that for 52/84 genes, performance for the best-performing model was better using SGD than `liblinear`, and for the other 32 genes performance was better using `liblinear`. In order to quantify whether the overfitting tendencies (or lack thereof) also hold across the gene set, we plotted the difference in performance between the best-performing model and the largest (least regularized) model; classifiers with a large difference in performance exhibit strong overfitting, and classifiers with a small difference in performance do not overfit (Figure 1D). For SGD, the least regularized models tend to perform comparably to the best-performing models, whereas for `liblinear` the distribution is wider suggesting that overfitting is more common.

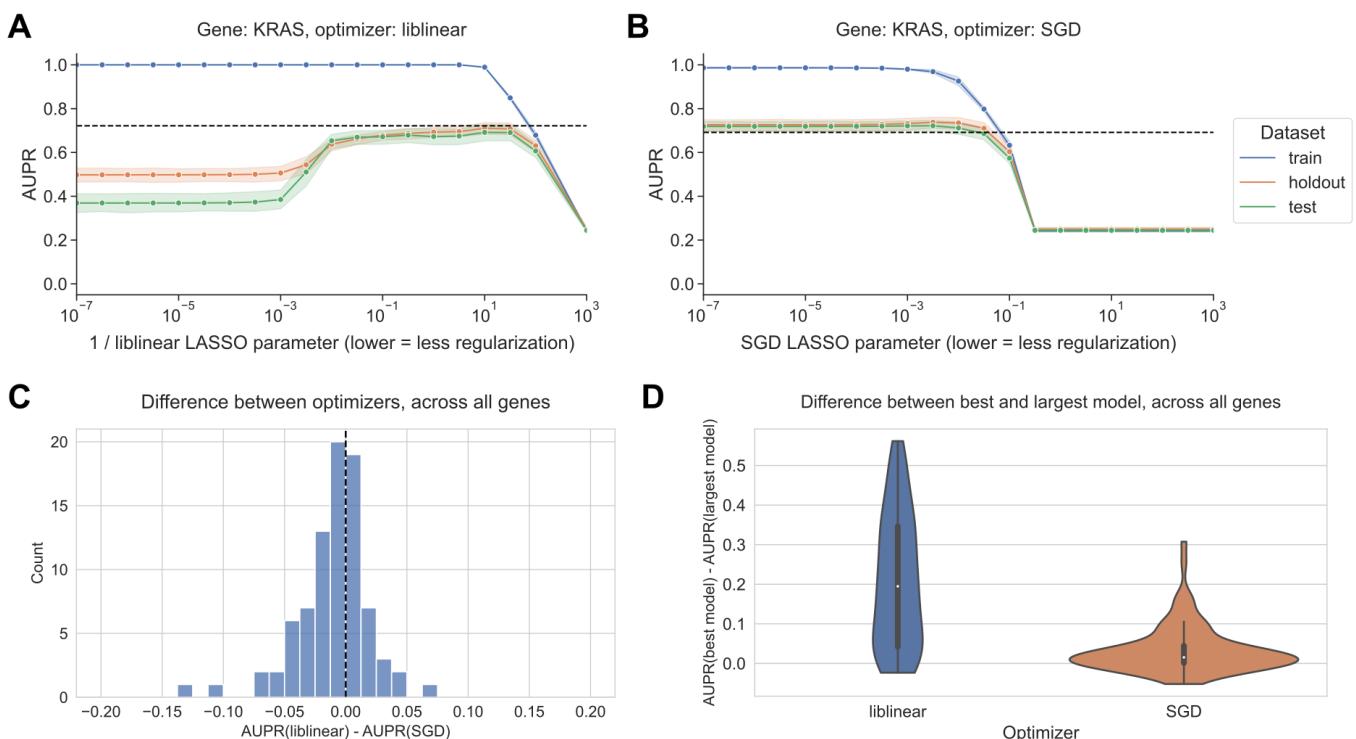


Figure 1: **A.** Performance vs. inverse regularization parameter for KRAS mutation status prediction, using the liblinear coordinate descent optimizer. Dotted lines indicate top performance value of the opposite optimizer. **B.** Performance vs. regularization parameter for KRAS mutation status prediction, using the SGD optimizer. “Holdout” dataset is used for SGD learning rate selection, “test” data is completely held out from model selection and used for evaluation. **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. **D.** Distribution of performance difference between best-performing model and largest (least regularized) model, for liblinear and SGD, across all 84 genes. Smaller numbers on the y-axis indicate less overfitting, and larger numbers indicate more overfitting.

SGD is sensitive to learning rate selection

The SGD results shown in Figure 1 select the best-performing learning rate using a grid search on the holdout dataset, independently for each regularization parameter. We also compared against other learning rate scheduling approaches implemented in scikit-learn (see Methods for implementation details and grid search specifications). For KRAS mutation prediction, we observed that the choice of initial learning rate and scheduling approach affects performance significantly, and other approaches to selecting the learning rate performed poorly relative to liblinear (black dotted lines in Figure 2) and to the grid search. We did not observe an improvement in performance over liblinear or the grid search for learning rate schedulers that decrease across epochs (Figure 2A, C, and D), nor did we see comparable performance when we selected a single constant learning rate for all levels of regularization without the preceding grid search (Figure 2B). Notably, scikit-learn’s default “optimal” learning rate schedule performed relatively poorly for this problem, suggesting that tuning the learning rate and selecting a well-performing scheduler is a critical component of applying SGD successfully for this problem (Figure 2D). We observed similar trends across all genes in the Vogelstein gene set, with other learning rate scheduling approaches performing poorly in aggregate relative to both liblinear and SGD with the learning rate grid search (Supplementary Figure 5).

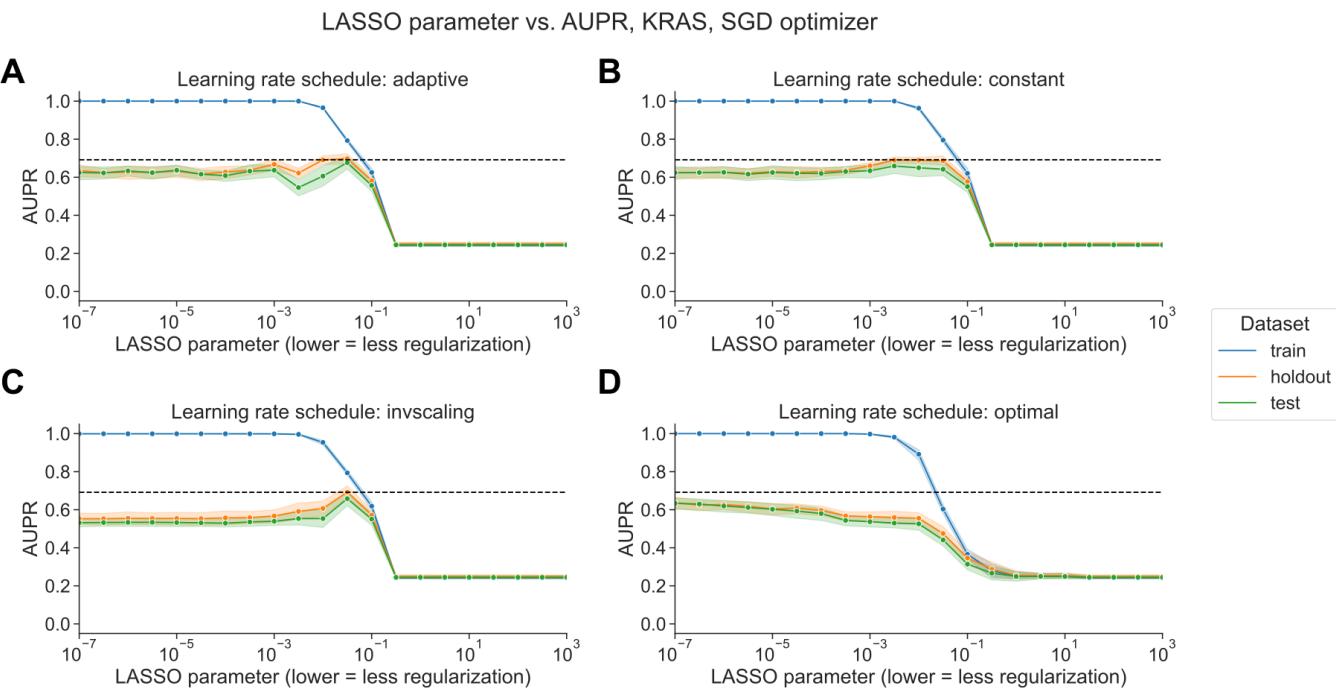


Figure 2: **A.** Performance vs. regularization parameter for KRAS mutation prediction, using SGD optimizer with adaptive learning rate scheduler. Dotted line indicates top performance value using liblinear, from Figure 1A. **B.** Performance vs. regularization parameter, using SGD optimizer with constant learning rate scheduler and a learning rate of 0.0005. **C.** Performance vs. regularization parameter, using SGD optimizer with inverse scaling learning rate scheduler. **D.** Performance vs. regularization parameter, using SGD optimizer with “optimal” learning rate scheduler.

liblinear and SGD result in different models, with varying loss dynamics

We sought to determine whether there was a difference in the sparsity of the models resulting from the different optimization schemes. In general across all genes, the best-performing SGD models mostly tend to have many nonzero coefficients, but with a distinct positive tail, sometimes having few nonzero coefficients. By contrast, the `liblinear` models are generally sparser with fewer than 2500 nonzero coefficients, out of ~16100 total input features, and a much narrower tail (Figure 3A). The sum of the coefficient magnitudes, however, tends to be smaller on average across all levels of regularization for SGD than for `liblinear` (Figure 3B). This effect is less pronounced for the other learning rate schedules shown in Figure 2, with the other options resulting in larger coefficient magnitudes (Supplementary Figure 6). These results suggest that the models fit by `liblinear` and SGD navigate the tradeoff between bias and variance in slightly different ways: `liblinear` tends to produce sparser models (more zero coefficients) as regularization increases, but if the learning rate is properly tuned, SGD coefficients tend to have smaller overall magnitudes as regularization increases.

We also compared the components of the loss function across different levels of regularization between optimizers. The LASSO logistic regression loss function can be broken down into a data-dependent component (the log-loss) and a parameter magnitude dependent component (the L1 penalty), which are added to get the total loss that is minimized by each optimizer; see Methods for additional details. As regularization strength decreases for `liblinear`, the data loss collapses to near 0, and the L1 penalty dominates the overall loss (Figure 3C). For SGD, on the other hand, the data loss decreases slightly as regularization strength decreases but remains relatively high (Figure 3D). Other SGD learning rate schedules have similar loss curves to the `liblinear` results, although this does not result in improved classification performance (Supplementary Figure 7).

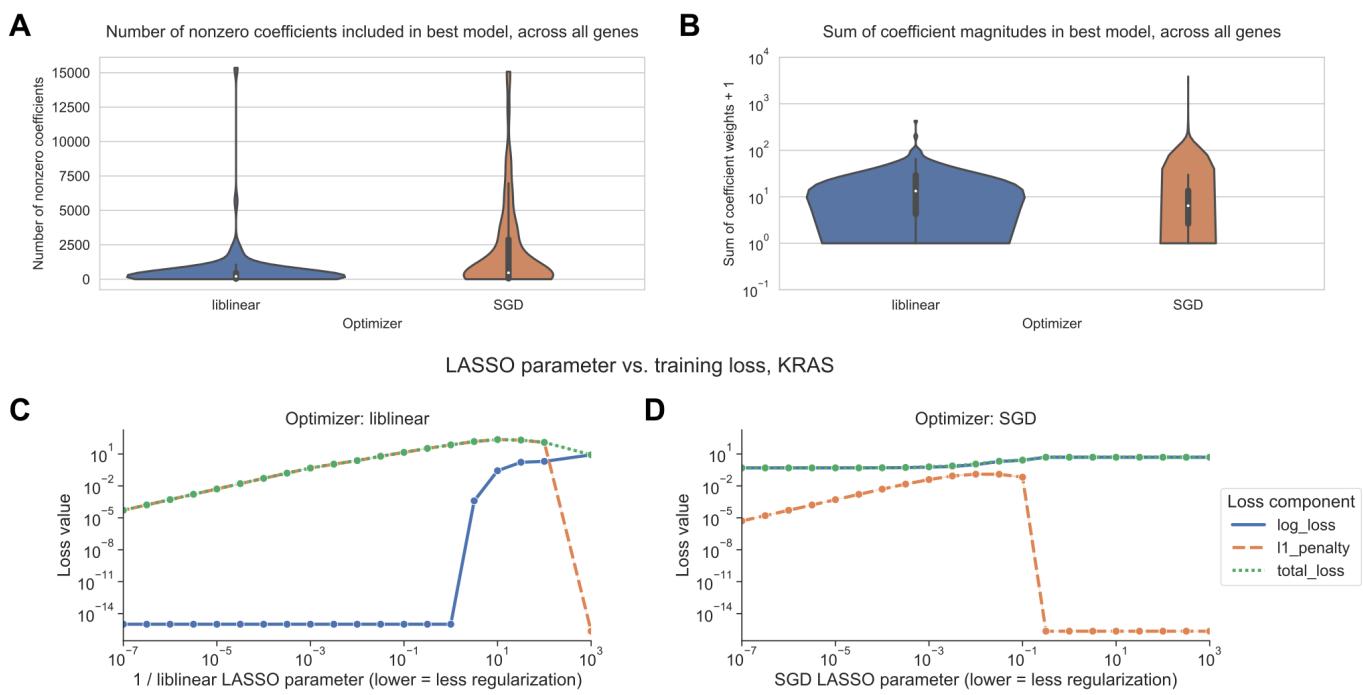


Figure 3: **A.** Distribution across genes of the number of nonzero coefficients included in best-performing LASSO logistic regression models. Violin plot density estimations are clipped at the ends of the observed data range, and boxes show the median/IQR. **B.** Distribution across genes of the sum of model coefficient weights for best-performing LASSO logistic regression models. **C.** Decomposition of loss function for models fit using `liblinear` across regularization levels. 0 values on the y-axis are rounded up to machine epsilon; i.e. 2.22×10^{-16} . **D.** Decomposition of loss function for models fit using SGD across regularization levels. 0 values on the y-axis are rounded up to machine epsilon; i.e. 2.22×10^{-16} .

Discussion

Our work shows that optimizer choice presents tradeoffs in model selection for cancer transcriptomics. We observed that LASSO logistic regression models for mutation status prediction fit

using stochastic gradient descent were highly sensitive to learning rate tuning, but they tended to perform robustly across diverse levels of regularization and sparsity. Coordinate descent implemented in `liblinear` did not require learning rate tuning, but generally performed best for a narrow range of fairly sparse models, overfitting as regularization strength decreased. Tuning of regularization strength for `liblinear`, and learning rate (and regularization strength to a lesser degree) for SGD, are critical steps which must be considered as part of analysis pipelines. The sensitivity we observed to these details highlights the importance of reporting exactly what optimizer was used, and how the relevant hyperparameters were selected, in studies that use machine learning models for transcriptomic data.

To our knowledge, the phenomenon we observed with SGD has not been documented in other applications of machine learning to genomic or transcriptomic data. In recent years, however, the broader machine learning research community has identified and characterized implicit regularization for SGD in many settings, including overparametrized or feature-rich problems as is often the case in transcriptomics [14,15,16]. The resistance we observed of SGD-optimized models to decreased performance on held-out data as model complexity increases is often termed “benign overfitting”: 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 attributed to optimization using SGD [16,17], and similar patterns have been observed for both linear models and deep neural networks [18,19].

Existing gene expression prediction benchmarks and pipelines typically use a single model implementation, and thus a single optimizer. We recommend thinking critically about optimizer choice, but this can be challenging for researchers 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 [20], which would presumably behave similarly to `liblinear`, but this is somewhat opaque in the `glmnet` documentation itself. Increased transparency and documentation in popular machine learning packages with respect to optimization, especially for models that are difficult to fit or sensitive to hyperparameter settings, would benefit new and unfamiliar users.

Related to what we see in our SGD-optimized models, there exist other problems in gene expression analysis where using all available features is comparable to, or 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 [21]. 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 [22]. In future work, it could be useful to explore if the coefficients found by `liblinear` and SGD emphasize the same pathways or functional gene sets, or if there are patterns to which mutation status classifiers (or other cancer transcriptomics classifiers) perform better with more/fewer nonzero coefficients.

Data and code availability

The data analyzed during this study were previously published as part of the TCGA Pan-Cancer Atlas project [23], and are available from the NIH NCI Genomic Data Commons (GDC). The scripts used to download and preprocess the datasets for this study are available at https://github.com/greenelab/pancancer-evaluation/tree/master/00_process_data, and the code used to carry out the analyses in this study is available at https://github.com/greenelab/pancancer-evaluation/tree/master/01_stratified_classification, both under the open-source BSD 3-clause license. Equivalent versions of Figure 1A and 1B for all 84 genes in the Vogelstein et al. 2013 gene set are available on Figshare at <https://doi.org/10.6084/m9.figshare.22728644>, under a CC0 license. This manuscript was written using Manubot [24] and is available on GitHub at

<https://github.com/greenelab/optimizer-manuscript> under the CC0-1.0 license. This research was supported in part by the University of Pittsburgh Center for Research Computing through the resources provided. Specifically, this work used the HTC cluster, which is supported by NIH award number S10OD028483.

Supplementary Material

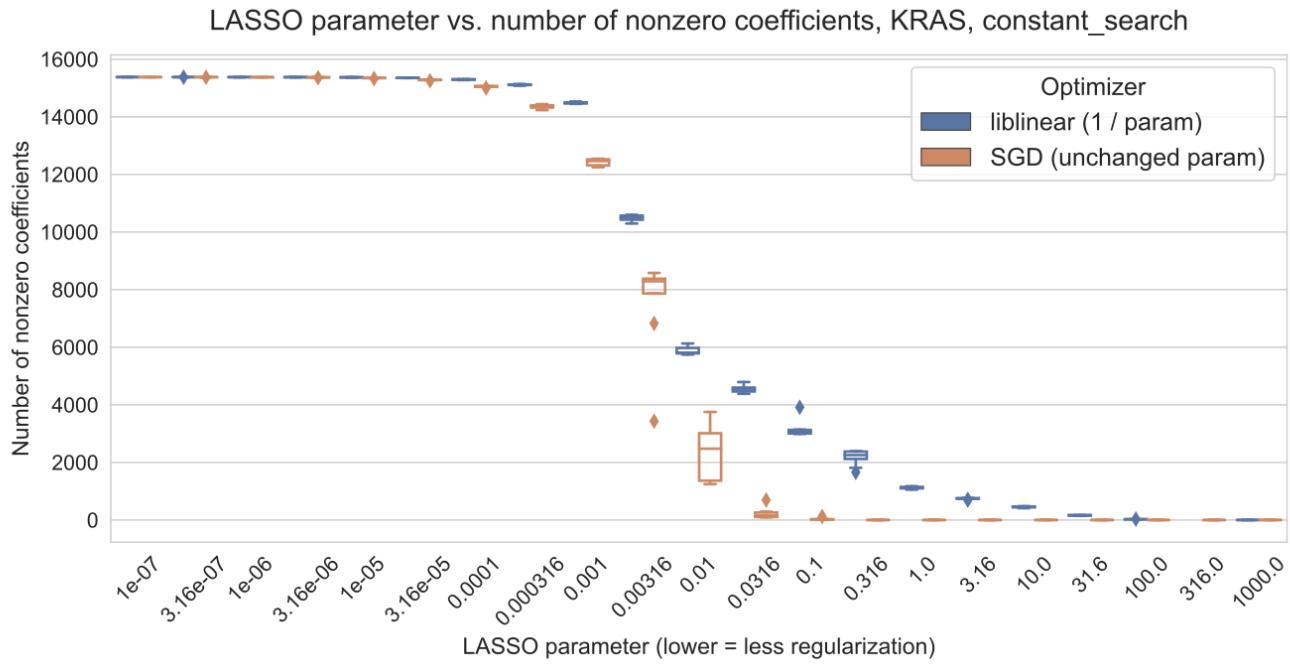


Figure 4: Number of nonzero coefficients (model sparsity) across varying regularization parameter settings for KRAS mutation prediction using SGD and liblinear optimizers.

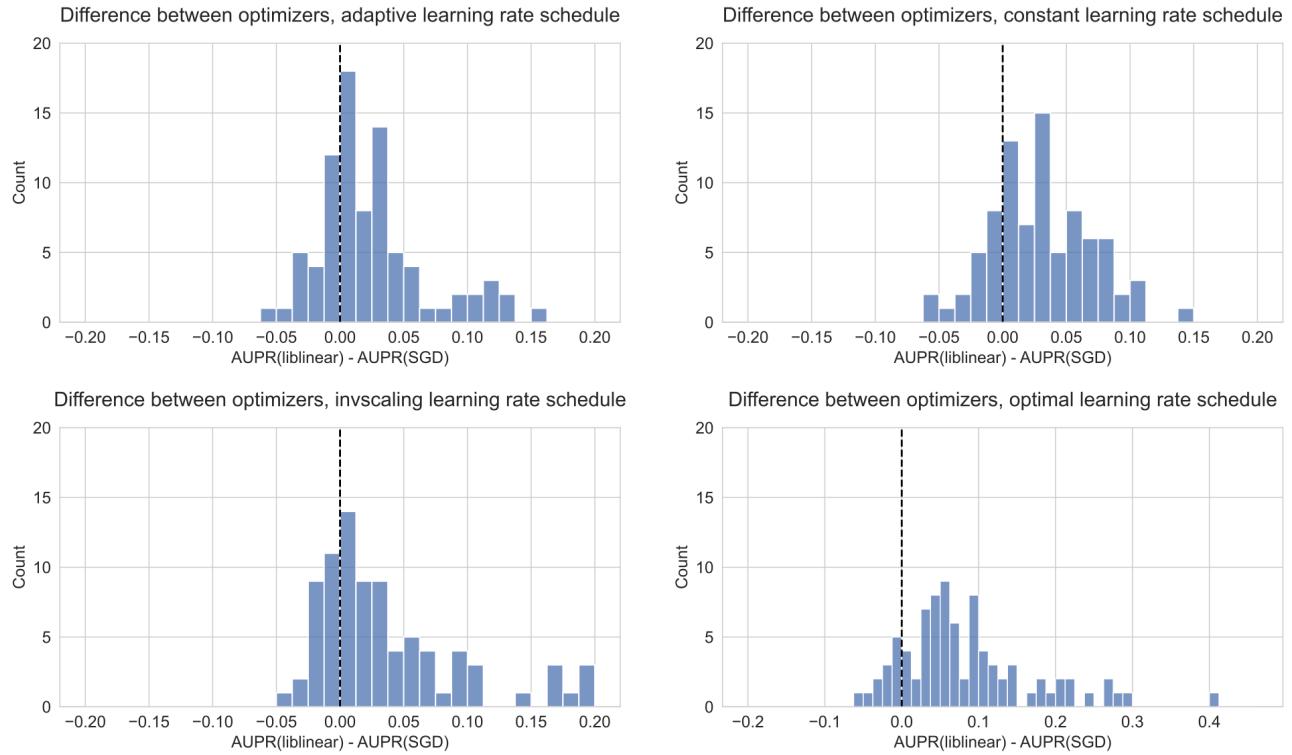


Figure 5: Distribution of performance difference between best-performing model for `liblinear` and SGD optimizers, across all 84 genes in Vogelstein driver gene set, for varying SGD learning rate schedulers. Positive numbers on the x-axis indicate better performance using `liblinear`, and negative numbers indicate better performance using SGD.

LASSO parameter vs. sum of coefficient weights, KRAS

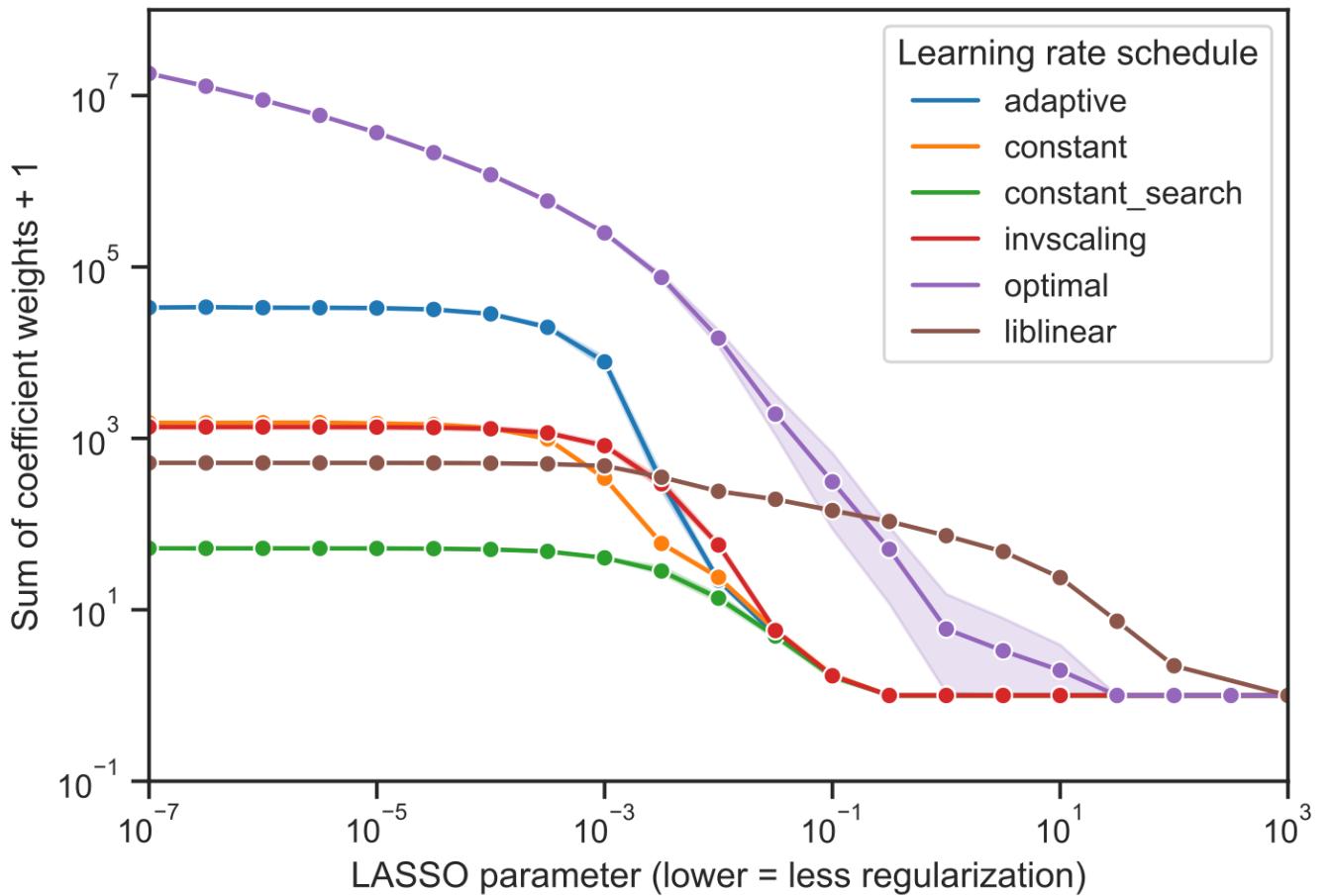


Figure 6: Sum of absolute value of coefficients + 1 for KRAS mutation prediction using SGD and `liblinear` optimizers, with varying learning rate schedules for SGD. Similar to the figures in the main paper, the `liblinear` x-axis represents the inverse of the C regularization parameter; SGD x-axes represent the untransformed α parameter.

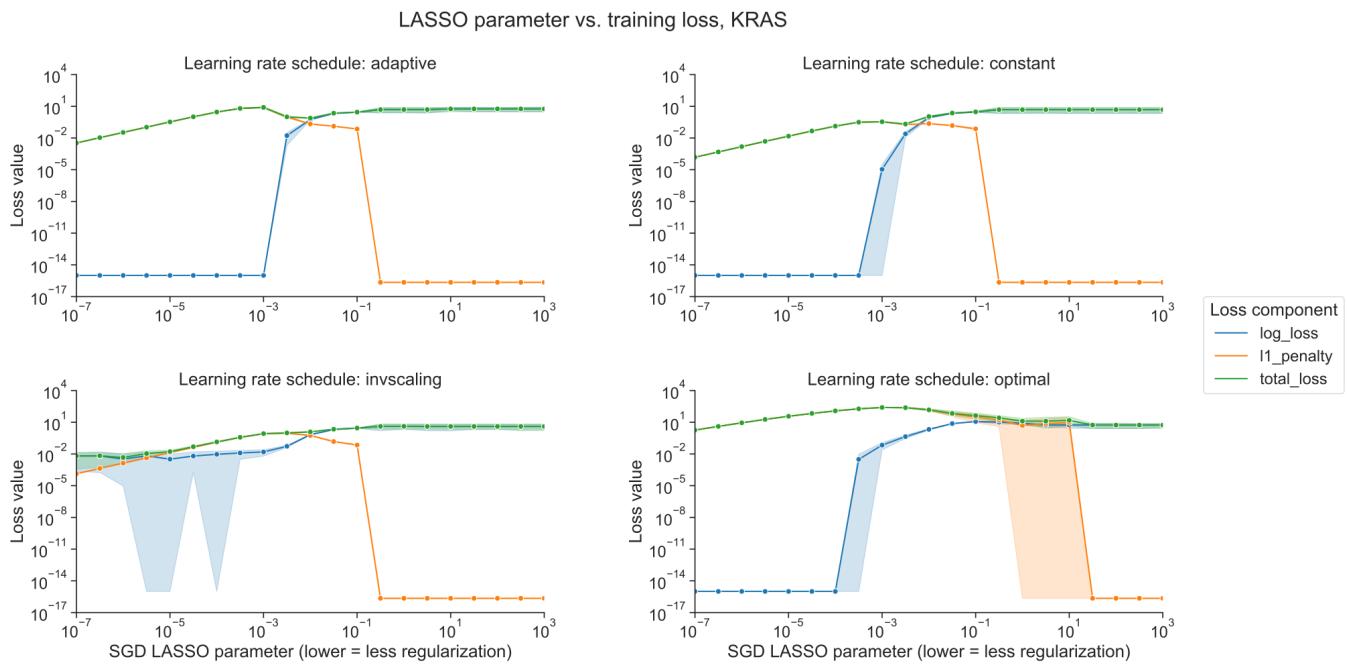


Figure 7: Decomposition of loss function into data loss and L1 penalty components for KRAS mutation prediction using SGD optimizer, across regularization levels, using varying learning rate schedulers. 0 values on the y-axis are rounded up to machine epsilon, i.e. 2.22×10^{-16} .

Chapter 3: Widespread redundancy in -omics profiles of cancer mutation states

This chapter has been published in *Genome Biology* (<https://doi.org/10.1186/s13059-022-02705-y>) as “Widespread redundancy in -omics profiles of cancer mutation states”.

Contributions: JC: conceptualization, methodology, software, visualization, writing - original draft, writing - review and editing BCC: methodology, writing - review and editing MC: methodology, writing - review and editing CSG: conceptualization, funding acquisition, methodology, supervision, writing - review and editing

Abstract

Background

In studies of cellular function in cancer, researchers are increasingly able to choose from many -omics assays as functional readouts. Choosing the correct readout for a given study can be difficult, and which layer of cellular function is most suitable to capture the relevant signal remains unclear.

Results

We consider prediction of cancer mutation status (presence or absence) from functional -omics data as a representative problem that presents an opportunity to quantify and compare the ability of different -omics readouts to capture signals of dysregulation in cancer. From the TCGA Pan-Cancer Atlas that contains genetic alteration data, we focus on RNA sequencing, DNA methylation arrays, reverse phase protein arrays (RPPA), microRNA, and somatic mutational signatures as -omics readouts. Across a collection of genes recurrently mutated in cancer, RNA sequencing tends to be the

most effective predictor of mutation state. We find that one or more other data types for many of the genes are approximately equally effective predictors. Performance is more variable between mutations than that between data types for the same mutation, and there is little difference between the top data types. We also find that combining data types into a single multi-omics model provides little or no improvement in predictive ability over the best individual data type.

Conclusions

Based on our results, for the design of studies focused on the functional outcomes of cancer mutations, there are often multiple -omics types that can serve as effective readouts, although gene expression seems to be a reasonable default option.

Background

Although cancer can be initiated and driven by many different genetic alterations, these tend to converge on a limited number of pathways or signaling processes [25]. As driver mutation status alone confers limited prognostic information, a comprehensive understanding of how diverse genetic alterations perturb central pathways is vital to precision medicine and biomarker identification efforts [26,27]. While many methods exist to distinguish driver mutations from passenger mutations based on genomic sequence characteristics [28,29,30], until recently it has been a challenge to connect driver mutations to downstream changes in gene expression and cellular function within individual tumor samples.

The Cancer Genome Atlas (TCGA) Pan-Cancer Atlas provides uniformly processed, multi-platform -omics measurements across tens of thousands of samples from 33 cancer types [23]. Enabled by this publicly available data, a growing body of work on linking the presence of driving genetic alterations in cancer to downstream gene expression changes has emerged. Recent studies have considered Ras pathway alteration status in colorectal cancer [31], alteration status across many cancer types in Ras genes [13,32], *TP53* [33], and *PIK3CA* [34], and alteration status across cancer types in frequently mutated genes [35]. More broadly, other groups have drawn on similar ideas to distinguish between the functional effects of different alterations in the same driver gene [36], to link alterations with similar gene expression signatures within cancer types [37], and to identify trans-acting expression quantitative trait loci (trans-eQTLs) in germline genetic studies [38].

These studies share a common thread: they each combine genomic (point mutation and copy number variation) data with transcriptomic (RNA sequencing) data within samples to interrogate the functional effects of genetic variation. RNA sequencing is ubiquitous and cheap, and its experimental and computational methods are relatively mature, making it a vital tool for generating insight into cancer pathology [39]. Some driver mutations, however, are known to act indirectly on gene expression through varying mechanisms. For example, oncogenic *IDH1* and *IDH2* mutations in glioma have been shown to interfere with histone demethylation, which results in increased DNA methylation and blocked cell differentiation [40,41,42,43]. Other genes implicated in aberrant DNA methylation in cancer include the TET family of genes [44] and *SETD2* [45]. Certain driver mutations, such as those in DNA damage repair genes, may lead to detectable patterns of somatic mutation [46]. Additionally, correlation between gene expression and protein abundance in cancer cell lines is limited, and proteomics data could correspond more directly to certain cancer phenotypes and pathway perturbations [47]. In these contexts and others, integrating different data modalities or combining multiple data modalities could be more effective than relying solely on gene expression as a functional signature.

Here, we compare -omics data types profiled in the TCGA Pan-Cancer Atlas to evaluate use as a multivariate functional readout of genetic alterations in cancer. We focus on gene expression (RNA sequencing data), DNA methylation (27K and 450K probe chips), reverse phase protein array (RPPA),

microRNA expression, and mutational signatures data [48] as possible readouts. Prior studies have identified univariate correlations of CpG site methylation [49,50] and correlations of RPPA protein profiles [51] with the presence or absence of certain driver mutations. Other relevant past work includes linking point mutations and copy number variants (CNVs) with changes in methylation and expression at individual genes [52,53] and identifying functional modules that are perturbed by somatic mutations [54,55]. However, direct comparison among different data types for this application is lacking, particularly in the multivariate case where we consider changes to -omics-derived gene signatures rather than individual genes in isolation.

We select a collection of potential cancer drivers with varying functions and roles in cancer development. We use mutation status in these genes as labels to train classifiers, using each of the data types listed as training data, in a pan-cancer setting; we follow similar methods to the elastic net logistic regression approach described in Way et al. 2018 [13] and Way et al. 2020 [35]. We show that there is considerable predictive signal for many genes relative to a cancer-type corrected baseline and that gene expression tends to provide good predictions of mutation state across most genes.

Surprisingly, we find that for a variety of genes, multiple data types are approximately equally effective predictors. We observe similar results for pan-cancer survival prediction across the same data types with little separation between the top-performing data types. In addition, we observe that combining data types into a single multi-omics model for mutation prediction provides little, if any, performance benefit over the most performant model using a single data type. Our results will help to inform the design of future functional genomics studies in cancer, suggesting that for many strong drivers with clear functional signatures, different -omics measurements can provide similar information content.

Methods

Mutation data download and preprocessing

To generate binary mutated/non-mutated gene labels for our machine learning model, we used mutation calls for TCGA samples from MC3 [9] and copy number threshold calls from GISTIC2.0 [10]. MC3 mutation calls were downloaded from the Genomic Data Commons (GDC) of the National Cancer Institute, at <https://gdc.cancer.gov/about-data/publications/pancanatlas>. Copy number threshold calls are from an older version of the GDC data, and are available here: https://figshare.com/articles/dataset/TCGA_PanCanAtlas_Copy_Number_Data/614412 [2]. We removed hypermutated samples (defined as five or more standard deviations above the mean non-silent somatic mutation count) from our dataset to reduce the number of false positives (i.e., non-driver mutations). After this filtering, 9,074 TCGA samples with mutation and copy number data remained. Any sample with a non-silent somatic variant in the target gene was included in the positive set. We also included copy number gains in the target gene for oncogenes and copy number losses in the target gene for tumor suppressor genes in the positive set; all remaining samples were considered negative for mutation in the target gene.

Omics data download and preprocessing

RNA sequencing, 27K and 450K methylation array, microRNA, and RPPA datasets for TCGA samples were all downloaded from GDC, at the same link provided above. Mutational signatures information for TCGA samples with whole-exome sequencing data was downloaded from the International Cancer Genome Consortium (ICGC) data portal, at https://dcc.icgc.org/releases/PCAWG/mutational_signatures/Signatures_in_Samples/SP_Signatures_in_Samples. For our experiments, we used only the “single base signature” (SBS) mutational signatures, generated in [48]. In general, before training classifiers or extracting PCA components from all of the data types, we standardized (took z-scores of) each column/feature of all

data types. For the RNA sequencing dataset, we generally used only the top 8,000 gene features by mean absolute deviation as predictors in our single-omics models, except where specified otherwise. For the RPPA, microRNA, and mutational signatures datasets, all columns/features were used.

To remove missing values from the methylation datasets, we removed the 10 samples with the most missing values, then performed mean imputation for probes with 1 or 2 values missing. All probes with missing values remaining after sample filtering and imputation were dropped from the analysis. This left us with 20,040 CpG probes in the 27K methylation dataset and 370,961 CpG probes in the 450K methylation dataset. For experiments where “raw” methylation data was used, we used the top 100,000 probes in the 450K dataset by mean absolute deviation for computational efficiency, and we used all of the 20,040 probes in the 27K dataset. For experiments where “compressed” methylation data was used, we used principal component analysis (PCA), as implemented in the `scikit-learn` Python library [56], to extract the top 5,000 principal components from the methylation datasets. We initially applied the beta-mixture quantile normalization (BMIQ) method [57] to correct for variability in signal intensity between type I and type II probes, but we observed that this had no effect on our results. We report uncorrected results in the main paper for simplicity.

Construction of a set of cancer genes

To get a comprehensive picture of classification performance across a wide variety of cancer-related genes, we integrated several curated gene sets from the literature into a single “merged” cancer gene set. The individual gene sets we integrated were from Vogelstein et al. [8] (all genes from Table S2A), Bailey et al. [58] (only genes annotated as “pan-cancer” drivers in Table S1), and the COSMIC Cancer Gene Census [59] (all Tier 1 genes annotated as “somatic”). In addition, the COSMIC CGC dataset contains 3 possible “roles in cancer” for each gene: oncogene, TSG, and fusion gene; for this analysis we dropped genes that are annotated only as fusion genes (i.e. no oncogene or TSG annotation). These filters resulted in a starting dataset of 511 cancer-related genes, which we reduced further for each experiment based on the number of mutated (i.e. positively labeled) samples as described in the next section.

Comparing data modalities

We made three main comparisons in this study: one between different sets of genes using only expression data, one comparing expression and DNA methylation data types, and one comparing all data types. This choice in comparisons was mainly due to sample size limitations, as running a single comparison using all data types would force us to use only samples that are profiled for every data type, which would discard a large number of samples that lack profiling on only one or a few data types. Thus, for each of the three comparisons, we used the intersection of TCGA samples having measurements for all of the datasets being compared in that experiment. This resulted in three distinct sets of samples: 9,074 samples shared between {expression, mutation} data, 7,981 samples shared between {expression, mutation, 27K methylation, 450K methylation}, and 5,226 samples shared between {expression, mutation, 27K methylation, 450K methylation, RPPA, microRNA, mutational signatures}. When we dropped samples between experiments as progressively more data types were added, we observed that the dropped samples had approximately the same cancer type proportions as the dataset as a whole. In other words, samples that were profiled for one data type but not another did not tend to come exclusively from one or a few cancer types. Exceptions included acute myeloid leukemia (LAML) which had no samples profiled in the RPPA data, and ovarian cancer (OV) which had only 8 samples with 450K methylation data. More detailed information on cancer type proportions profiled for each data type is provided in Additional File 1: Fig. S1 and Additional File 2.

For each target gene, in order to ensure that the training dataset was reasonably balanced (i.e. that there would be enough mutated samples to train an effective classifier), we included only cancer types with at least 15 mutated samples and at least 5% mutated samples, which we refer to here as

“valid” cancer types. After applying these filters, the number of valid cancer types remaining for each gene varied based on the set of samples used: more data types resulted in fewer shared samples, and fewer samples generally meant fewer valid cancer types. In some cases, this resulted in genes with no valid cancer types, which we dropped from the analysis. Out of the 511 genes from the “merged” cancer gene set described in the previous section, for the analysis using {expression, mutation} data we retained 268 target genes, for the {expression, mutation, 27k methylation, 450k methylation} analysis we retained 272 genes, and for the analysis using all data types we retained 217 genes.

We additionally explored mutation prediction from gene expression alone using three gene sets of equal size: the cancer-related genes from the merged dataset described previously, a set of frequently mutated genes in TCGA, and a set of random genes with mutations profiled by MC3. To match the size of the merged cancer gene set, we took the 268 most frequently mutated genes in TCGA as quantified by MC3, all of which had at least one valid cancer type. For the random gene set, we first filtered to the set of all genes with one or more valid cancer types by the same criteria (15 total samples mutated and at least 5% of samples mutated), then sampled 268 of the resulting 1,348 genes uniformly at random. Based on the results of the gene expression experiments, we used the merged cancer-related gene set for all subsequent experiments comparing -omics data types.

Training classifiers to detect cancer mutations

We trained logistic regression classifiers to predict whether or not a given sample had a mutational event in a given target gene using data from various -omics datasets as explanatory variables. Our model was trained on -omics data (X) to predict mutation presence or absence (y) in a target gene. To control for varying mutation burden per sample and to adjust for potential cancer type-specific expression patterns, we included one-hot encoded cancer type and $\log_{10}(\text{sample mutation count})$ in the model as covariates. Since our -omics datasets tend to have many dimensions and comparatively few samples, we used an elastic net penalty to prevent overfitting [60] in line with the approach used in Way et al. 2018 [13] and Way et al. 2020 [35]. Elastic net logistic regression finds the feature weights $\hat{w} \in \mathbb{R}^p$ solving the following optimization problem:

$$\hat{w} = \operatorname{argmin}_w \ell(X, y; w) + \alpha\lambda\|w\|_1 + \frac{1}{2}\alpha(1-\lambda)\|w\|_2$$

where $i \in \{1, \dots, n\}$ denotes a sample in the dataset, $X_i \in \mathbb{R}^p$ denotes features (omics measurements) from the given sample, $y_i \in \{0, 1\}$ denotes the label (mutation presence/absence) for the given sample, and $\ell(\cdot)$ denotes the negative log-likelihood of the observed data given a particular choice of feature weights, i.e.

$$\ell(X, y; w) = - \sum_{i=1}^n y_i \log\left(\frac{1}{1 + e^{-w^\top X_i}}\right) + (1 - y_i) \log\left(1 - \frac{1}{1 + e^{-w^\top X_i}}\right)$$

This optimization problem leaves two hyperparameters to select: α (controlling the tradeoff between the data log-likelihood and the penalty on large feature weight values), and λ (controlling the tradeoff between the L1 penalty and L2 penalty on the weight values). Although the elastic net optimization problem does not have a closed form solution, the loss function is convex, and iterative optimization algorithms are commonly used for finding reasonable solutions. For fixed values of α and λ , we solved for \hat{w} using stochastic gradient descent, as implemented in `scikit-learn`'s `SGDClassifier` method.

Given weight values \hat{w} , it is straightforward to predict the probability of a positive label (mutation in the target gene) $P(y^* = 1 | X^*; \hat{w})$ for a test sample X^* :

$$P(y^* = 1 \mid X^*; \hat{w}) = \frac{1}{1 + e^{-\hat{w}^\top X^*}}$$

and the probability of no mutation in the target gene, $P(y^* = 0 \mid X^*; \hat{w})$, is given by (1 - the above quantity).

For each target gene, we evaluated model performance using two replicates of 4-fold cross-validation, where train and test splits were stratified by cancer type and sample type. That is, each training set/test set combination had equal proportions of each cancer type (BRCA, SKCM, COAD, etc.) and each sample type (primary tumor, recurrent tumor, etc.). To choose the elastic net hyperparameters, we used 3-fold nested cross-validation, with a grid search over the following hyperparameter ranges: $\lambda = [0.0, 0.05, 0.1, 0.3, 0.5, 0.7, 0.9, 1.0]$ and $\alpha = [0.0001, 0.001, 0.01, 0.1, 1, 10]$. Using the grid search results, for each evaluation fold we selected the set of hyperparameters with the optimal area under the precision-recall curve (AUPR), averaged over the three inner folds.

Evaluating mutation prediction classifiers

Area under the receiver-operator curve (AUROC) [61] and area under the precision-recall curve (AUPR) [62] are metrics that are frequently used to quantify classification performance for a continuous or probabilistic output, such as that provided by logistic regression. These metrics summarize performance across a variety of binary label thresholds, rather than requiring choice of a single threshold to determine positive or negative predictions. In the main text, we report results using AUPR, summarized using average precision. AUPR has been shown to distinguish between models more accurately than AUROC when there are few positively labeled samples [63,64]. As an additional correction for imbalanced labels, in many of the results in the main text we report the difference in AUPR between a classifier fit to true mutation labels and a classifier fit to data where the mutation labels are randomly permuted. In cases where mutation labels are highly imbalanced (very few mutated samples and many non-mutated samples), a classifier with permuted labels may perform well simply by chance, e.g. by predicting the negative/non-mutated class for most samples. To maintain the same label balance for the classifiers with permuted labels as the classifiers with the true labels, we permuted labels separately in the train and test sets for each cross-validation split. Additionally, to maintain the same label proportions within each cancer type after permuting the labels, we permuted labels independently for each cancer type.

Recall that for each target gene and each -omics dataset, we ran two replicates of 4-fold cross-validation, for a total of eight performance results. To make a statistical comparison between two models using these performance distributions, we used paired-sample *t*-tests, where performance measurements derived from the same cross-validation fold are considered paired measurements. We used this approach to compare a model trained on true labels with a model trained on permuted labels (addressing the question, “for the given gene using the given data type, can we predict mutation status better than random”), and to compare a model trained on data type A with a model trained on data type B (addressing the question, “for the given gene, can we make more effective mutation status predictions using data type A or data type B”).

We corrected for multiple tests using a Benjamini-Hochberg false discovery rate correction. For experiments where we chose a binary threshold for accepting/rejecting H_0 we set a conservative corrected threshold of $p = 0.001$; we were able to estimate the number of false positives by examining genes with better performance for permuted mutation labels than true labels. We chose this threshold to ensure that none of the observed false positive genes were considered significant, since we would never expect permuting labels to improve performance. However, our results were not sensitive to the choice of this threshold, and we display cutoffs of $p = 0.05$ and $p = 0.01$ in many of our plots as well.

Survival prediction using -omics datasets

As a complementary comparison to mutation prediction, we constructed predictors of patient survival using the clinical data available from the GDC, in the TCGA-CDR-SupplementalTableS1.xlsx file. Following the methods described in [65], as the clinical endpoint we used overall survival (OS), except in nine cancer types with few deaths observed where we used progression-free intervals (PFI) as the clinical endpoint (BRCA, DLBC, LGG, PCPG, PRAD, READ, TGCT, THCA and THYM). For prediction, we used Cox regression as implemented in the scikit-survival Python package [66], with patient age at diagnosis and $\log_{10}(\text{sample mutation count})$ included as covariates, as well as a one-hot encoded variable for cancer type in the pan-cancer case. To ensure that the per-feature information content was comparable between -omics data types, we preprocessed the -omics datasets using PCA and extracted the top k principal components; in the case where the number of features in the original dataset was less than k we used all available PCs (that is, we set $k = \min(p, k)$ where p is the number of features in the unprocessed dataset). For the pan-cancer models we plot results over multiple values of k : $k \in \{10, 100, 500, 1000, 5000\}$; for the individual cancer type models we set $k = 10$.

To model pan-cancer survival (results shown in main paper), we used the elastic net Cox regression implementation in scikit-survival (i.e. the CoxnetSurvivalAnalysis method). To select hyperparameters for the elastic net Cox regression model, we performed a grid search over $\lambda = [0.0, 0.05, 0.1, 0.3, 0.5, 0.7, 0.9, 1.0]$ and $\alpha = [0, 1e-5, 1e-4, 5e-4, 0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1, 10, 100, 1000]$. To select the regularization parameter α , we used the default selection procedure implemented in scikit-survival to determine a range of potential α values based on the data. This procedure begins by deriving the maximum α value as the smallest value for which all coefficients are 0 (call this α_{\max}), then it selects 100 possibilities for alpha spaced evenly on a log scale between α_{\max} and $0.01 \cdot \alpha_{\max}$. We found that for individual cancer types, this data-driven procedure resulted in more consistent and stable model convergence than choosing a fixed set of alphas to search over as in the pan-cancer survival prediction experiments.

We measured survival prediction performance using the censored concordance index (c-index) [[pubmed:8668867?](#)], which quantifies agreement between the order of survival time predictions and true outcomes for a held-out dataset; higher c-index values indicate more accurate survival prediction performance. Similar to the mutation prediction experiments, we calculated c-index values on held-out subsets of the data for two replicates of 4-fold cross-validation, resulting in eight performance measurements for each model. As a baseline, for both the pan-cancer and cancer type specific datasets, we constructed survival models using only non-omics covariates. For the pan-cancer data, covariates included patient age at diagnosis, $\log_{10}(\text{sample mutation count})$, and a one-hot encoded variable for sample cancer type. The cancer type-specific baseline models were the same, without the cancer type indicator, since all train and test samples were derived from the same cancer type.

Multi-omics mutation prediction experiments

To predict mutation presence or absence in cancer genes using multiple data types simultaneously, we concatenated individual datasets into a large feature matrix, then used the same elastic net logistic regression method described previously. For this task, we considered only the gene expression, 27K methylation, and 450K methylation datasets. We used only these data types to limit the number of multi-omics combinations; the expression and methylation datasets resulted in the best overall performance across the single-omics experiments, so we limited combinations to those datasets. In the main text, we report results using the top 5,000 principal components for each dataset, which ensures that most variance is captured (approximately 95-98% of variance for each data type). In Additional File 1: Fig. S6, we also report results using “raw” features: for gene expression we used all

15,639 genes available in our RNA sequencing dataset, and for the 27K and 450K methylation datasets we used the top 20,000 CpG probes by mean absolute deviation.

To construct the multi-omics models, we considered each of the pairwise combinations of the datasets listed above, as well as a combination of all 3 datasets. When combining multiple datasets, we concatenated along the column axis and included covariates for cancer type and sample mutation burden as before. For all multi-omics experiments, we used only the samples from TCGA with data for all three data types (i.e. the same 7,981 samples used in the single-omics experiments comparing expression and methylation data types). We considered a limited subset of well-performing genes from the merged cancer gene set as target genes, including *EGFR*, *IDH1*, *KRAS*, *PIK3CA*, *SETD2*, and *TP53*. We selected these genes because we had previously observed that they have good predictive performance and because they represent a combination of alterations that have strong gene expression signatures (*KRAS*, *EGFR*, *IDH1*, *TP53*) and strong DNA methylation signatures (*IDH1*, *SETD2*, *TP53*).

For the experiments predicting mutation status using a 3-layer fully connected neural network, described in the Results section and Additional File 1: Fig. S7, we used the top 5,000 principal components as input for each data type. We selected hyperparameters for each of the 8 outer cross-validation splits using a single inner train/validation split and a random search over 20 hyperparameter combinations. The hyperparameter ranges that we sampled from in the random search were: `learning_rate: [0.1, 0.01, 0.001, 5e-4, 1e-4]`, `h1_size: [1000, 500, 250]`, `dropout: [0.5, 0.75, 0.9]`, `weight_decay: [0, 0.1, 1, 100]`. Here, `h1_size` refers to the size of the first hidden layer, and the size of the second hidden layer was always set to `h1_size / 2`. As in the elastic net grid search, we chose the combination of hyperparameters with the best AUPR on the validation set, and retrained the model with those hyperparameters to make predictions on the test set. We trained our networks with the Adam optimizer [67], with ReLU activation after the hidden layers and sigmoid activation to make predictions, and using binary cross-entropy as the loss function as implemented in the PyTorch `BCEWithLogitsLoss` function, through the `skorch` library which provides interoperability between PyTorch and scikit-learn.

Results

Using diverse data modalities to predict cancer alterations

We collected five different data modalities from cancer samples in the TCGA Pan-Cancer Atlas, capturing five steps of cellular function that are perturbed by genetic alterations in cancer (Figure 8A). These included gene expression (RNA-seq data), DNA methylation (27K and 450K Illumina BeadChip arrays), protein abundance (RPPA data), microRNA expression data, and patterns of somatic mutation (mutational signatures). To link these diverse data modalities to changes in mutation status, we used elastic net logistic regression to predict the presence or absence of mutations in cancer genes, using these readouts as predictive features (Figure 8B). We evaluated the resulting mutation status classifiers in a pan-cancer setting, preserving the proportions of each of the 33 cancer types in TCGA for eight train/test splits (4 folds x 2 replicates) in each of approximately 250 cancer genes (Figure 8C).

We sought to compare classifiers against a baseline where mutation labels are permuted (to identify genes whose mutation status correlates strongly with a functional signature in a given data type) and also to compare classifiers trained on true labels across different data types (to identify data types that are more or less predictive of mutations in a given gene). To account for variation between dataset splits in making these comparisons, we treat classification metrics from the eight train/test splits as performance distributions, which we compare using *t*-tests. We summarize performance across all genes in our cancer gene set using a similar approach to a volcano plot, in which each point is a gene. In our summary plots, the x-axis shows the magnitude of the change in the classification metric between conditions, and the y-axis shows the *p*-value for the associated *t*-test (Figure 8C).

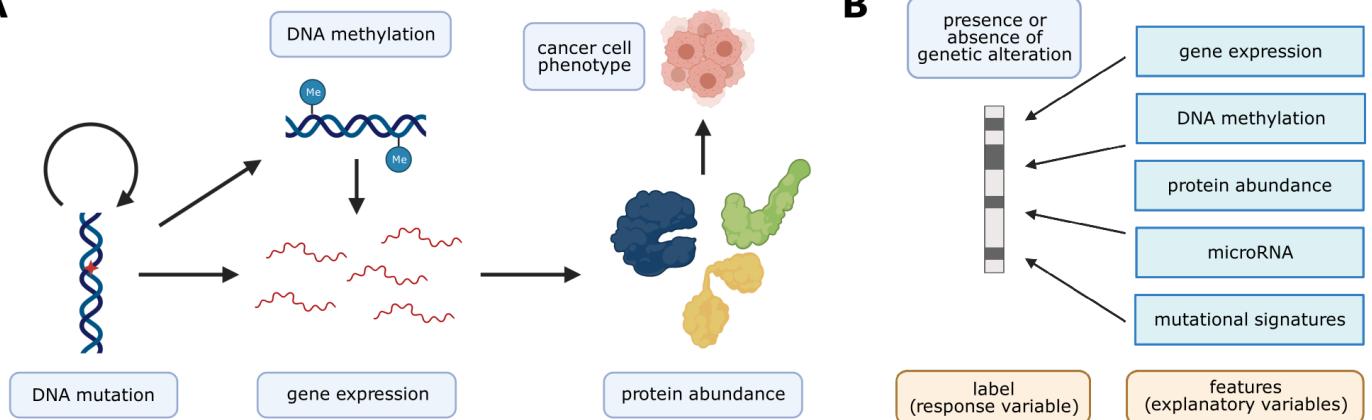
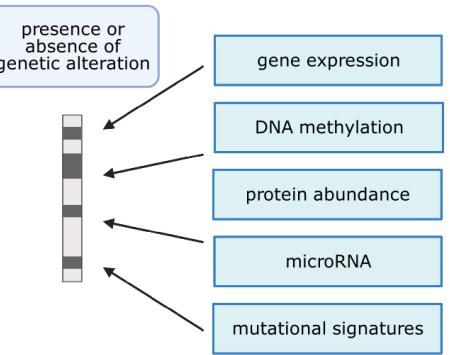
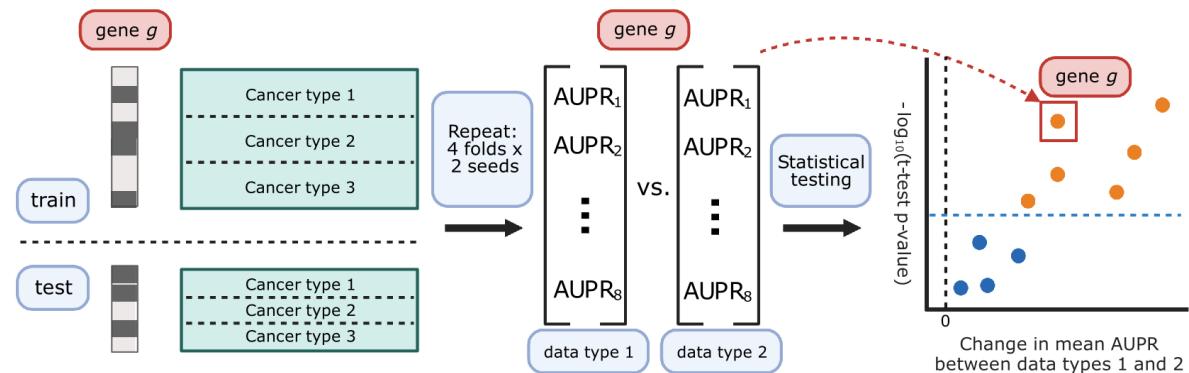
A**B****C**

Figure 8: **A.** Cancer mutations can perturb cellular function via a variety of cellular processes. Arrows represent major potential paths of information flow from a somatic mutation in DNA to its resulting cell phenotype; circular arrow represents the ability of certain mutations (e.g. in DNA damage repair genes) to alter somatic mutation patterns. Note that this does not reflect all possible relationships between cellular processes: for instance, changes in gene expression can lead to changes in somatic mutation rates. **B.** Predicting presence/absence of somatic alterations in cancer from diverse data modalities. In this study, we use functional readouts from TCGA as predictive features and the presence or absence of mutation in a given gene as labels. This reverses the primary direction of information flow shown in Panel A. **C.** Schematic of evaluation pipeline.

Selection of cancer-related genes improves predictive signal

As a baseline, we evaluated prediction of mutation status from gene expression data across several different gene sets. Past work has evaluated mutation prediction for the top 50 most mutated genes in TCGA [35], and we sought to extend this to a broader list of gene sets. To evaluate whether using known cancer-related genes tends to improve prediction, we compiled a set of cancer-related genes ($n=268$) from Vogelstein et al. 2013 [8], Bailey et al. 2018 [58], and the COSMIC Cancer Gene Census [59]. We compared performance on this curated gene set with performance on an equal number of genes sampled randomly after applying a mutation frequency threshold ($n=268$, see Methods for sampling details) and an equal number of the most mutated genes in TCGA ($n=268$). For all gene sets, we used only the set of TCGA samples for which both gene expression and somatic mutation data exists, resulting in a total of 9,074 samples across all 33 cancer types. This set of samples was further filtered for each target gene to cancer types containing at least 15 mutated samples and at least 5% of samples mutated for that cancer type. As an alternate approach, we tried including/excluding entire genes using similar filters, and the results were consistent across filtering strategies (Additional File 1: Fig. S4). We then evaluated the performance for each target gene in each of the three gene sets.

Overall, genes from the cancer-related gene set were more predictable than randomly chosen genes or those selected by total mutation count (Figure 9A). In total, for a significance threshold of $\alpha = 0.001$, 120/268 genes (44.8%) in the cancer-related gene set are significantly predictable from gene expression data, compared to 14/268 genes (5.22%) in the random gene set and 80/268 genes (29.9%) in the most mutated gene set. Of the 14 significantly predictable genes in the random gene

set, 13 of them are also in the cancer-related gene set (highlighted with 'X' in Figure 9B), and of the 80 significantly predictable genes in the most mutated gene set, 26 of them are also in the cancer-related gene set (highlighted in red in Figure 9C). These results suggest that selecting target genes for mutation prediction based on prior knowledge of their involvement in cancer pathways and processes, rather than randomly or based on mutation frequency alone, can improve predictive signal and identify more highly predictable mutations from gene expression data.

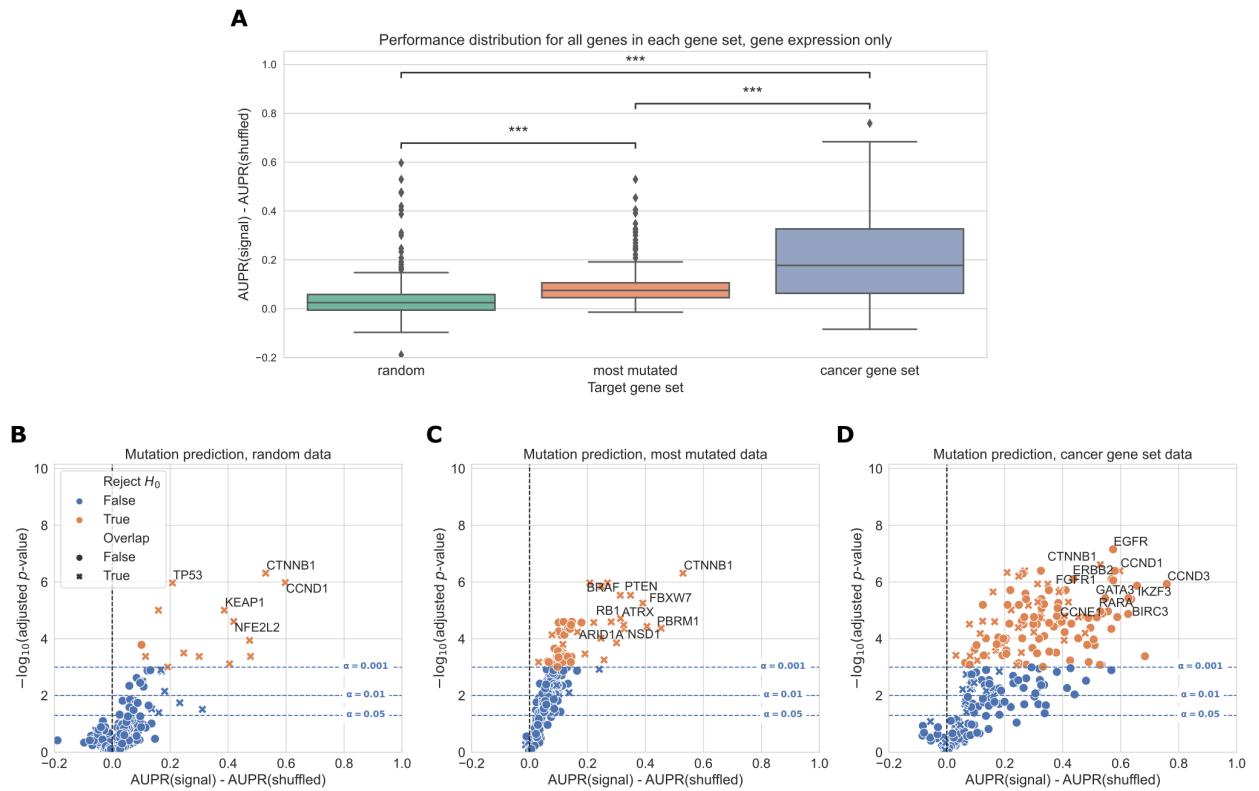


Figure 9: **A.** Overall distribution of performance across three gene sets, using gene expression (RNA-seq) data to predict mutations. Each data point represents the mean cross-validated AUPR difference, compared with a baseline model trained on permuted mutation presence/absence labels, for one gene in the given gene set; notches show bootstrapped 95% confidence intervals. “random” = 268 random genes, “most mutated” = 268 most mutated genes, “cancer gene set” = 268 cancer related genes from curated gene sets. Significance stars indicate results of Bonferroni-corrected pairwise Wilcoxon tests: **: $p < 0.01$, ***: $p < 0.001$, ns: not statistically significant for a cutoff of $p = 0.05$. **B, C, D.** Volcano-like plots showing mutation presence/absence predictive performance for each gene in each of the three gene sets. The x-axis shows the difference in mean AUPR compared with a baseline model trained on permuted labels, and the y-axis shows p -values for a paired t -test comparing cross-validated AUPR values within folds. Points (genes) marked with an “X” are overlapping between the cancer gene set and either the random or most mutated gene set.

Gene expression predicts cancer mutation status more effectively than DNA methylation

We compared gene expression with DNA methylation as downstream readouts of the effects of cancer alterations. In these analyses, we considered both the 27K probe and 450K probe methylation datasets generated for the TCGA Pan-Cancer Atlas. As target genes, we used the same combined cancer-related gene set described in the “Selection of cancer-related genes” section. We used samples that had data for each of the data types being compared, including somatic mutation data to generate mutation labels. This process retained 7,981 samples in the intersection of the expression, 27K methylation, 450K methylation, and mutation datasets, which we used for subsequent analyses. The most frequent missing data types were somatic mutation data (1,114 samples) and 450K methylation data (1,072 samples) (Figure 10A).

For many genes, predictions are better than our baseline model where labels are permuted (values greater than 0 in the box plots), suggesting that there is considerable predictive signal in both

expression and methylation datasets across the cancer-related gene set (Figure 10B). On aggregate across all genes, predictive performance is best overall for gene expression. Both before and after filtering for genes that exceed the significance threshold, gene expression with raw gene features provides a significant performance improvement relative to the 27K methylation and 450K methylation datasets (Figure 10B-C). Results were similar with PCA-compressed gene expression features or raw CpG probes as predictors (Additional File 1: Fig. S5).

Considering each target gene in the cancer-related gene set individually, we observed that 113/272 genes significantly outperformed the permuted baseline using gene expression data, as compared to 62/272 genes for 27K methylation and 77/272 genes for 450K methylation (Figure 10D-F, more information about specific genes in Additional File 1: Fig. S2). Some “well-predicted” genes that outperformed the permuted baseline tended to be similar between data types (Figure 10D-F; genes in the top right of each plot). For example, *C/C* appears in the top right of all 3 plots, and *CCND1* appears in the top right of the gene expression and 450K methylation plots, suggesting that mutations in these genes have strong gene expression and DNA methylation signatures, and these signatures tend to be preserved across cancer types.

In addition to comparing mutation classifiers trained on different data types to the permuted baseline, we also compared classifiers trained on true labels directly to each other, for genes that performed significantly better than the baseline for both of the data types under consideration (Figure 10G-H). We observed that 18/58 genes were significantly more predictable from expression data than 27K methylation data, and 21/69 genes were significantly more predictable from expression data than 450K methylation data. In both cases, no genes were significantly more predictable using the methylation data types. Still, we observed that some points were clustered around the origin, indicating that the data types appear to confer similar information about mutation status. That is, in these cases, matching the gene being studied with the “correct” data modality seems to be unimportant: mutation status has a strong signature which can be extracted from both expression and DNA methylation data roughly equally.

We additionally compared pan-cancer survival prediction performance using principal components derived from each data type; in general, results were comparable across the three data types (Figure 10I). All data types outperformed the covariate-only baseline (see [Methods](#)) for lower numbers of PC features included, although performance was similar to the baseline for higher numbers of PCs. Confidence intervals between the best- and worst-performing data types overlap at most PC counts (with the exception of gene expression at 5,000 PC features), suggesting that similarly to mutation prediction, the three data types tend to have comparable effectiveness for pan-cancer survival prediction.

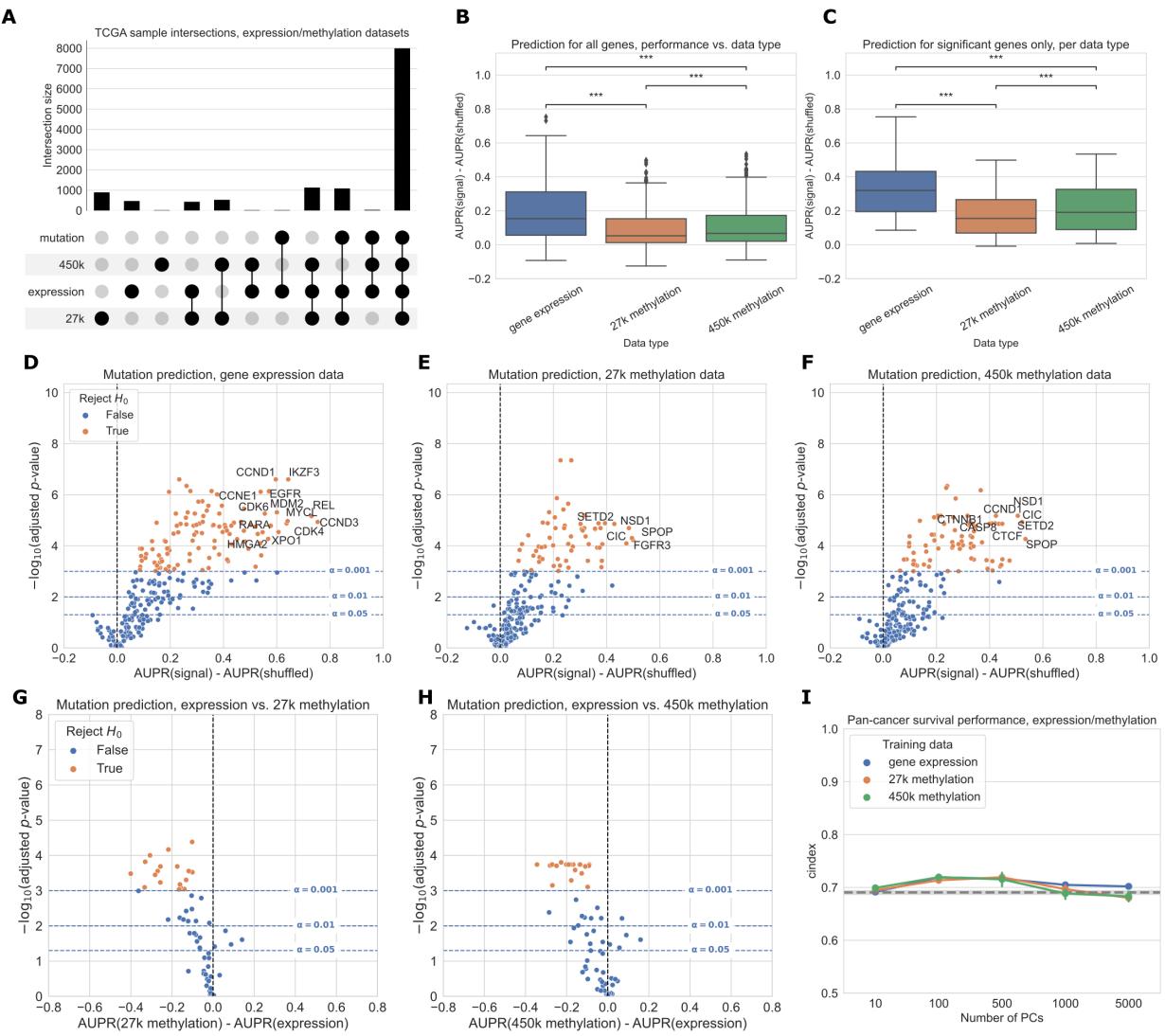


Figure 10: **A.** Count of overlapping samples between gene expression, 27K methylation, 450K methylation, and somatic mutation data used from TCGA. Only non-zero overlap counts are shown. Somatic mutation sample information is included because it is needed to generate the mutation presence/absence labels. **B.** Predictive performance for genes in the cancer-related gene set, using each of the three data types as predictors. The gene expression predictor uses the top 8000 gene features by mean absolute deviation, and the methylation predictors use the top 5000 principal components as predictive features. Significance stars indicate results of Bonferroni-corrected pairwise Wilcoxon tests: **: $p < 0.01$, ***: $p < 0.001$, ns: not statistically significant for a cutoff of $p = 0.05$. **C.** Predictive performance for genes where at least one of the considered data types predicts mutation labels significantly better than the permuted baseline. **D-F.** Predictive performance for each gene in the cancer-related gene set, for each data type, compared with a baseline model trained on permuted labels. **G-H.** Direct comparison of performance using gene expression and each methylation dataset, for genes that perform significantly better than the baseline for both data types. Points (genes) to the left of $y=0$ perform better using gene expression-derived features, and points to the right perform better using methylation-derived features. **I.** Pan-cancer survival prediction performance, quantified using c-index on the y-axis, for gene expression, 27K methylation, and 450K methylation. The x-axis shows results with varying numbers of principal components included for each data type. Models also included covariates for patient age, sample mutation burden, and sample cancer type; grey dotted line indicates mean performance for a covariate-only baseline model.

Focusing on several selected genes of interest, we observed that relative classifier performance varies by gene (Figure 11). Past work has indicated that mutations in *TP53* are highly predictable from gene expression data [33], and we observed that the methylation datasets provided similar predictive performance (Figure 11A). Similarly, for *IDH1* both expression and methylation features result in similar performance, consistent with the previously observed role of IDH1 in regulating both DNA methylation and gene expression (Figure 11D) [68]. Mutations in *KRAS* and *ERBB2* (*HER2*) were most predictable from gene expression data, and in both cases the methylation datasets significantly outperformed the baseline as well (Figure 11B and 11E). Gene expression signatures of *ERBB2* alterations are historically well-studied in breast cancer [69], and samples with activating *ERBB2* mutations have recently been shown to share sensitivities to some small-molecule inhibitors across

cancer types [70]. These observations are consistent with the pan-cancer *ERBB2* mutant-associated expression signature that we observed in this study. *NF1* mutations were also most predictable from gene expression data, although the gene expression-based *NF1* mutation classifier did not significantly outperform the baseline with permuted labels at a cutoff of $\alpha = 0.001$ (Figure 11C). *SETD2* is an example of a gene that is more predictable from the methylation datasets than from gene expression, although gene expression with raw gene features significantly outperformed the permuted baseline as well (Figure 11F). *SETD2* is widely mutated across cancer types and affects H3K36 histone methylation most directly, but *SETD2*-mediated changes in H3K36 methylation have been linked to dysregulation of diverse cellular processes including DNA methylation and RNA splicing [45,71].

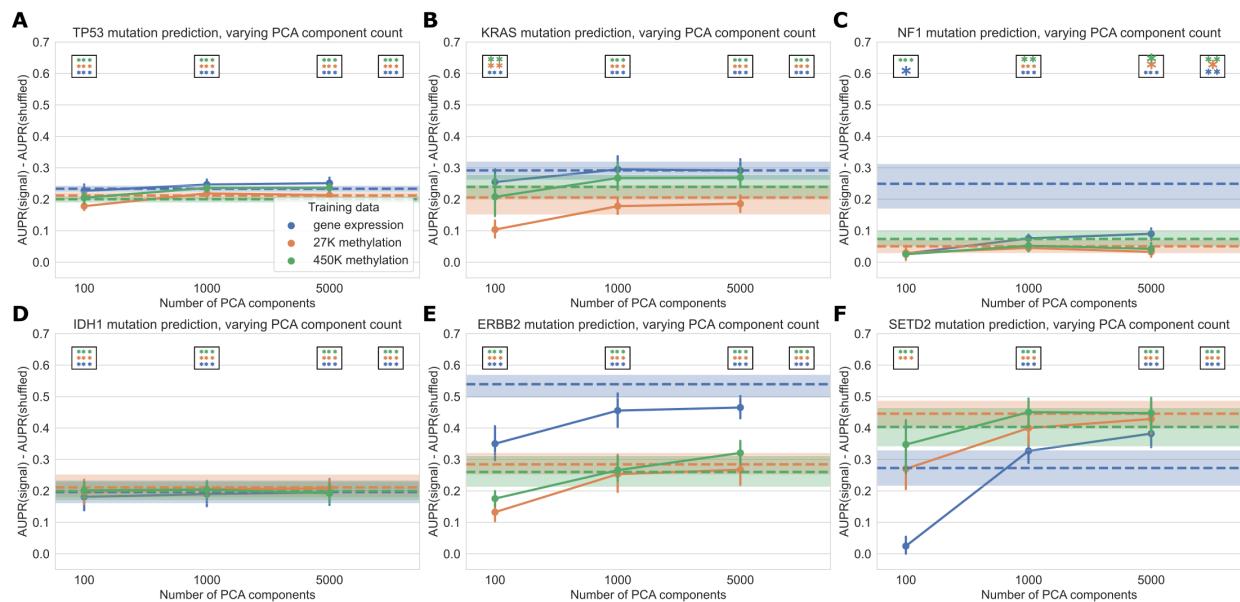


Figure 11: Performance across varying PCA dimensions for specific genes of interest. Dotted lines represent results for “raw” features (8,000 gene features for gene expression data and 8,000 CpG probes for both methylation datasets, selected by largest mean absolute deviation). Error bars and shaded regions show bootstrapped 95% confidence intervals. Stars in boxes show statistical testing results compared with permuted baseline model; each box refers to the model using the number of PCA components it is over (far right box = models with raw features). **: $p < 0.01$, ***: $p < 0.001$, no stars: not statistically significant for a cutoff of $p = 0.05$.

Comparing six different readouts favors expression and DNA methylation

Next, we expanded our comparison to all five functional data modalities (six total readouts, since there are two DNA methylation platforms) available in the TCGA Pan-Cancer Atlas. As with previous experiments, we limited our comparison to the set of samples profiled for each readout, resulting in 5,226 samples with data for all readouts. The data types with the most missing samples were RPPA data (2,215 samples that were missing RPPA data) and 450K methylation (630 samples that were missing 450K methylation data) (Figure 12A). Summarized over all genes in the cancer-related gene set, we observed that gene expression tended to produce better predictions than the other data types (Figure 12B). This remained true when we looked only at the set of genes having at least one significant predictor (i.e. “well-predicted” genes) (Figure 12C).

On the individual gene level, mutations in 33/217 genes were significantly predictable from RPPA data relative to the permuted baseline, compared to 25/217 genes from microRNA data and 2/217 genes from mutational signatures data (Figure 12D-F). For the remaining data types on this smaller set of samples, 79/217 genes outperformed the baseline for gene expression data, 31/217 for 27k methylation, and 42/217 for 450k methylation. Compared to the methylation experiments (Figure 10), we observed fewer “well-predicted” genes for the expression and methylation datasets here (likely due to the considerably smaller sample size) but relative performance was comparable (Additional File

1: Fig. S3). Direct comparisons between each added data type and gene expression data showed that for most “well-predicted” genes, RPPA, microRNA and mutational signatures data generally provide similar or worse performance compared to gene expression (Figure 12G-I).

Performance using RPPA data (Figure 12G) is notable because of its drastically smaller dimensionality than the other data types (190 proteins, compared to thousands of features for the expression and methylation data types). This suggests that each protein abundance measurement provides a high information content, although this is by design as the antibody probes used for the TCGA analysis were selected to cover established cancer-related pathways [72]. Similarly, the scope of the features captured by the mutational signatures data we used is limited to single-base substitution signatures; a broader spectrum of possible signatures is described in previous work [48,73] including doublet-substitution signatures, small indel signatures, and signatures of structural variation, but these were not readily available for the TCGA exome sequencing data. The relatively poor predictive ability of mutational signatures likely stems from a combination of biological and technical factors, as there is no reason to expect that changes in somatic mutation patterns would be directly caused by most cancer driver mutations. Two exceptions are *KMT2C* and *KMT2D* (Figure 12F), which may have a role in mediating DNA damage response [74].

As in the expression/methylation comparison, we also compared pan-cancer survival prediction performance between all six readouts, using the top principal components derived from each data type to ensure comparable information content (Figure 12J). All six readouts performed comparably for this smaller set of samples, with slightly better performance across PC feature dimensions for the 450K methylation array. The covariate-only baseline predictor performed considerably worse than it did in the expression/methylation comparisons, with all -omics data types outperforming the baseline predictor at all PC numbers.

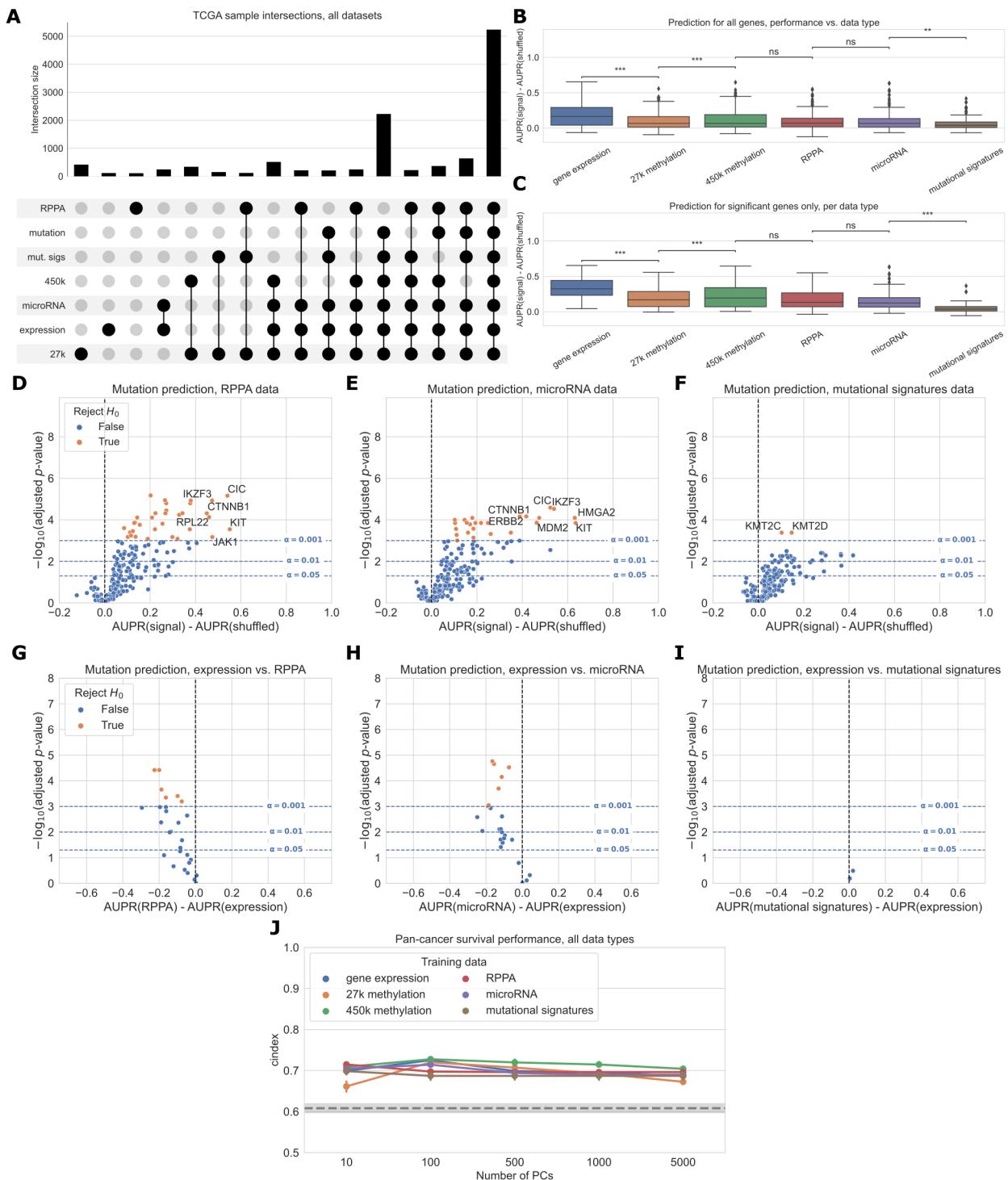


Figure 12: **A.** Overlap of TCGA samples between all data types used in mutation prediction comparisons. Only overlaps with more than 100 samples are shown. Somatic mutation sample information is included because it is needed to generate the mutation presence/absence labels. **B.** Overall distribution of performance per data type across 217 genes from the cancer-related gene set. Each data point represents mean cross-validated AUPR difference, compared with a baseline model trained on permuted labels, for one gene; notches show bootstrapped 95% confidence intervals. Significance stars indicate results of Bonferroni-corrected pairwise Wilcoxon tests: **: $p < 0.01$, ***: $p < 0.001$, ns: not statistically significant for a cutoff of $p = 0.05$. All pairwise tests were run, and corrected for, but only neighboring test results are shown. **C.** Overall performance distribution per data type for genes where the permuted baseline model is significantly outperformed for one or more data types, resulting in a total of 39 genes. **D, E, F.** Volcano-like plots showing predictive performance for each gene in the cancer-related gene set, in each of the added data types (RPPA, microRNA, mutational signatures). The x-axis shows the difference in mean AUPR compared with a baseline model trained on permuted labels, and the y-axis shows p -values for a paired t -test comparing cross-validated AUPR values within folds. **G, H, I.** Direct comparison of performance using gene expression and each added data type, showing only genes that perform significantly better than the baseline model for both data types. Points (genes) to the left of $y=0$ perform better using gene expression-derived features, and points to the right perform better using the added data type (RPPA, microRNA, and mutational signatures respectively). **J.** Pan-cancer survival prediction performance, quantified using c-index on the y-axis, for all data types. The x-axis shows results with varying numbers of principal components included

for each data type. Models also included covariates for patient age, sample mutation burden, and sample cancer type; grey dotted line indicates mean performance for a covariate-only baseline model.

When we constructed a heatmap depicting predictive performance for each gene across data types, we found that many genes tended to be well-predicted by more than one data type (Figure 13). Of the 86 genes that are well-predicted using at least one data type (grey circles in Figure 13), 52/86 (60.5%) are well-predicted by multiple data types, meaning that multiple -omics readouts contain a detectable signature of presence/absence of a mutation in the corresponding gene. Of the remaining 34 genes, 28/34 (82.4%) are well-predicted by gene expression alone. This supports our observation that in a surprising number of cases, choosing the “correct” data modality is unimportant for driver genes with strong functional signatures, although gene expression may be the best “default” choice as it tends to be a strong predictor in the majority of cases. Exceptions included *ERBB4*, *KMT2A*, *PIK3R1*, and *RPL22* (only well-predicted using RPPA data), *FAT4* (only well-predicted using microRNA data), and *KDM6A* (only well-predicted using 450K methylation data).

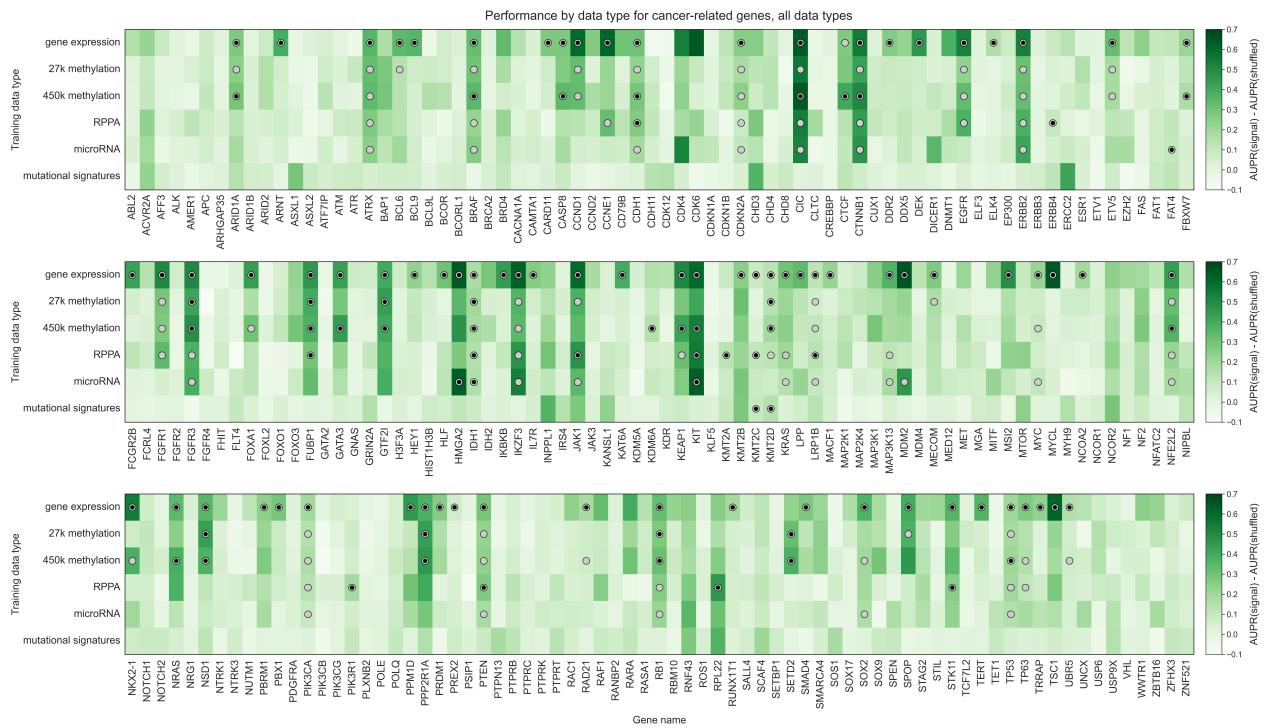


Figure 13: Heatmap displaying predictive performance for mutations in each of the 217 genes from the cancer-related gene set, across all six TCGA data modalities. Each cell quantifies performance for a target gene, using predictive features derived from a particular data type. Grey shaded dots indicate that the given data type provides significantly better predictions than the permuted baseline for the given gene; black inner dots indicate the same and additionally that the given data type provides statistically equivalent performance to the data type with the best average performance (determined by pairwise *t*-tests across data types with FDR correction).

Simple multi-omics integration provides little performance benefit

We also trained “multi-omics” classifiers to predict mutations in six well-studied and widely mutated driver genes across various cancer types: *EGFR*, *IDH1*, *KRAS*, *PIK3CA*, *SETD2*, and *TP53*. Each of these genes is well-predicted from several data types in our earlier experiments (Figure 13), consistent with having strong pan-cancer driver effects. For the multi-omics classifiers, we considered all pairwise combinations of the top three performing individual data types (gene expression, 27K methylation, and 450K methylation), in addition to a model using all three data types. We trained a classifier for multiple data types by concatenating features from the individual data types, then fitting the same elastic net logistic regression model as we used for the single-omics models. Here, we show results using the top 5,000 principal components from each data type as predictive features, to ensure that feature count and scale is comparable among data types; results for raw features are shown in Additional File 1: Fig. S6. We additionally ran the same experiments using a 3-layer fully-connected

neural network for classification, with principal components as input, and results are shown in Additional File 1: Fig. S7. In general, we found predictions using elastic net logistic regression to be more robust across cross-validation folds and hyperparameter choices than predictions using the neural network, although the neural network provided a slight performance improvement using multiple -omics types for some genes.

For each of the six target genes, we observed comparable performance between the best single-omics classifier (blue boxes in Figure 14A) and the best multi-omics classifier (orange boxes in Figure 14A). Across all classifiers and data types, we found varied patterns based on the target gene. For *IDH1* and *TP53* performance is relatively consistent regardless of data type(s), suggesting that baseline performance is high and there is little room for improvement as data is added (Figure 14C, G). The *TP53* classifier using raw features showed a statistically significant improvement when multiple data types were integrated, although the difference in mean performance was relatively small (Additional File 1: Fig. S6, $p=0.0078$). For *EGFR*, *KRAS*, and *PIK3CA*, combining gene expression with methylation data results in statistically equivalent or worse performance to gene expression alone; classifiers trained only on methylation data generally do not perform as well as those that integrate expression data (Figure 14B, D, E). Previously, we saw that the best classifiers for *SETD2* used methylation data alone (Figure 13). When we added multiple data types to our *SETD2* classifier, we did observe an increase in performance (Figure 14F), although this improvement was not statistically significant in a paired-sample t -test for $\alpha=0.05$ ($p=0.078$). Overall, we observed that combining data types in a relatively simple manner, by concatenating features from each individual data type, provided little or no improvement in predictive ability over the best individual data type. This supports our earlier observations of the redundancy of gene expression and methylation data as functional readouts, since our multi-omics classifiers are not in general able to extract gains in predictive performance as more data types are added for this set of cancer drivers.

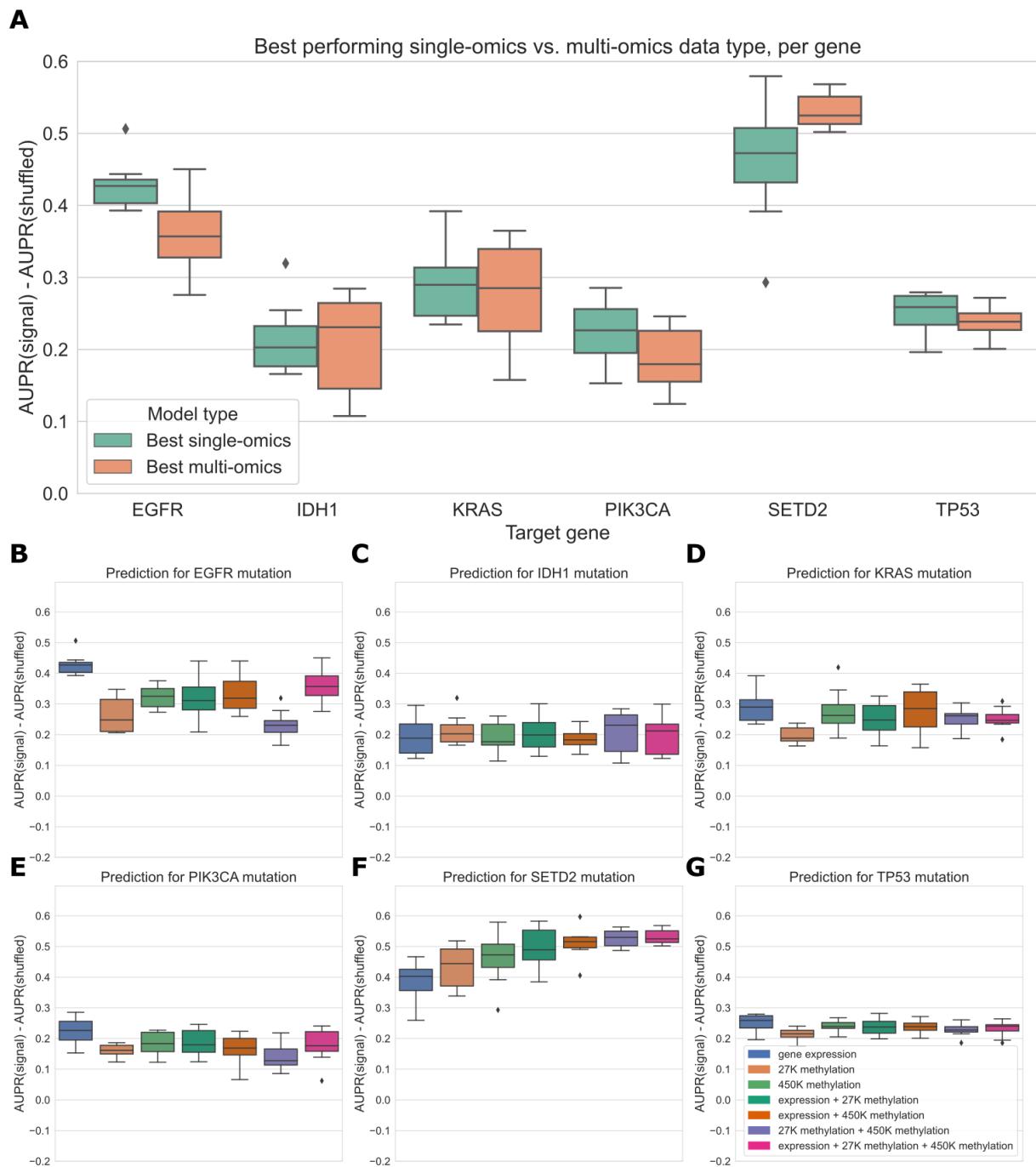


Figure 14: **A.** Comparing the best-performing model (i.e. highest mean AUPR relative to permuted baseline) trained on a single data type against the best “multi-omics” model for each target gene. None of the differences between single-omics and multi-omics models were statistically significant using paired-sample Wilcoxon tests across cross-validation folds, for a threshold of 0.05. **B-G.** Classifier performance, relative to baseline with permuted labels, for mutation prediction models trained on various combinations of data types. Each panel shows performance for one of the six target genes; box plots show performance distribution over 8 evaluation sets (4 cross-validation folds x 2 replicates).

Discussion

We carried out a large-scale comparison of data types in the TCGA Pan-Cancer Atlas as functional readouts of genetic alterations in cancer, integrating results across cancer types and across driver genes. Overall, we found that gene expression captures signatures of mutation state most effectively in general, relative to a baseline model, but we saw that for many genes other data types could be equally effective at predicting mutation presence or absence. For pan-cancer survival prediction, we found that the functional readouts tended to be similarly effective, outperforming a simple baseline using age and sample mutation burden in most cases. Our multi-omics modeling experiment

indicated that the mutation state information captured by gene expression and DNA methylation is highly redundant, as added data types resulted in no gain or modest gains in classifier performance.

Comparing mutation status prediction using raw and PCA compressed expression and DNA methylation data, we observed that feature extraction using PCA provided no benefit compared to using raw gene or CpG probe features. Other studies using DNA methylation array data have found that nonlinear dimension reduction methods, such as variational autoencoders and capsule networks, can be effective for extracting predictive features [75, pubmed:34417465?]. The latter approach is especially interesting because capsule networks and “capsule-like methods” can be constrained to extract features that align with known biology (i.e. that correspond to known disease pathways or CpG site annotations). This can improve model interpretability as well as predictive performance. Similar methods have been applied to extract biologically informed features from gene expression data (see, for instance, [76, 77]). A more comprehensive study of dimension reduction methods in the context of mutation prediction, including the features selected by these methods and their biological relevance and interpretation, would be a beneficial area of future work. In addition to methods for extracting features, another aspect of the study that could be explored further is methods for labeling samples as mutated or not more efficiently. Although the mutation calls we used from MC3 represent the consensus of multiple algorithms aggregated through a standard pipeline, benchmarking other methods for identifying mutated samples could improve the utility of our method, such as calling mutations directly from RNA-seq data to avoid the need for paired samples [78, 79].

In contrast to many other studies demonstrating the benefits of integrating multiple -omics data types for various cancer-related prediction problems [80, 81, 82, 83, 84], we found that combining multiple data types to predict mutation status was generally not effective for this problem. The method we used to integrate different data types by concatenating feature sets is sometimes referred to as “early” data integration (discussed in more detail in [85] and [86]). It is possible that more sophisticated data integration methods, such as “intermediate” integration methods that learn a set of features jointly across datasets, would produce improved predictions. We do not interpret our results as concrete evidence that multi-omics integration is not effective for this problem; rather, we see them as an indication that this is a challenging data integration problem for which further investigation is needed. We also present this problem as a set of benchmark tasks on which multi-omics integration methods can be evaluated. In addition to the methodological questions, the issue of data integration also has implications for the underlying biology: a more nuanced understanding of when different data readouts provide redundant information, and when they can contribute unique information about cancer pathology and development, could have many translational applications.

One limitation of the current study is that, for the mutation prediction problem, we only evaluated classifiers that were trained on pan-cancer data. Considering every possible combination of target gene and TCGA cancer type (85 target genes x 33 cancer types x 6 data types) would have drastically increased the computational load and presented a large multiple testing burden. Alternatively, choosing only a subset of gene/cancer type combinations to study would have biased our results toward known driver gene/cancer type relationships, which we aimed to avoid. In future work it would be interesting to identify classifiers that perform well in a certain cancer type but not in the pan-cancer context and to compare these instances across different cancer types. As a motivating example, other studies have shown that activating mutations in Ras isoforms (*HRAS*, *KRAS*, *NRAS*) tend to have similar effects to one another in thyroid cancer, producing similar gene expression signatures [37]. In multiple myeloma, however, activating *KRAS* and *NRAS* mutations produce distinct expression signatures, necessitating separate classifiers [87]. A high-throughput computational pipeline to identify cases where functional signatures of a particular cancer driver are either concordant or discordant between cancer types could identify opportunities for context-specific protein function prediction, improve biomarker identification, and suggest cases where drugs targeting specific alterations might produce discordant results in different cancer types.

As with any study relying on observational, cross-sectional data such as the TCGA Pan-Cancer Atlas, the conclusions that we can draw are limited by the data. In particular, for any of our “well-predicted” genes (i.e. genes that, when mutated, have strong signatures in one or more data types), we cannot definitively distinguish correlation from causation. To directly assess the effects of particular mutations on various data modalities, some studies use cell line data from sources such as the Cancer Cell Line Encyclopedia (CCLE) [88]. While this approach could help to isolate the causal effect of a given mutation on a given cell line, cell lines are sometimes an imperfect match for the cancers they are derived from [89]. We are also limited in that we cannot assign timing or clonal status to mutations, or fully characterize intratumor heterogeneity, with certainty from the bulk sequencing data generated by TCGA (although some features of tumor mutational processes over time can be estimated from bulk data, e.g. [90]). As methods for generating large longitudinal datasets at single-cell resolution mature and scale, we will need to revise the way we think about cellular function and dysregulation in cancer cells, as dynamic and adaptive processes rather than a single representative snapshot of a tumor.

Conclusions

Based on our results, for studies focused on the functional consequences of cancer mutations, we recommend that researchers cancers prioritize downstream readouts based on the gene or genes of interest (Figure 13). On balance, prediction of mutation status is best in general using gene expression data, and prediction of patient survival is similar for all data types in the study. However, the finding that for many genes, multiple functional profiles contain much of the same information will be useful for some study designs, given varying cost and stability of different readouts. In addition to gene expression, results using DNA methylation and RPPA measurements as predictive features were promising, especially considering the substantially lower dimensionality of the RPPA dataset compared to other data types. It is important to note that the specific technologies chosen by TCGA, and the tradeoffs inherent in such a high-throughput study, could influence the comparison: it is possible that, for instance, another technology for measuring DNA methylation (such as bisulfite sequencing) or another technique for measuring protein abundance (such as mass spectrometry-based proteomics) could change performance for those data types. Future technology advances, in both quality and quantity of data, are likely to improve our understanding of the full picture of functional consequences of mutations in cancer cells.

Declarations

Availability of data and materials

The datasets analyzed during this study were previously published as part of the TCGA Pan-Cancer Atlas project, and are publicly available from the NIH NCI Genomic Data Commons (GDC) [91]. The mutational signatures dataset was downloaded from the ICGC Data Portal [92]. Scripts used to download and preprocess the datasets for this study are available at

https://github.com/greenelab/mpmp/tree/master/00_download_data.

All analyses were implemented in the Python programming language and are available at Zenodo [93] and in the following GitHub repository: <https://github.com/greenelab/mpmp> [94] under the open-source BSD 3-clause license. Scripts to download large data files from GDC and other sources are located in the `00_download_data` directory. Scripts to run experiments comparing data modalities used individually are located in the `02_classify_mutations` directory, scripts to run multi-omics experiments are located in the `05_classify_mutations_multimodal` directory, and scripts to run survival prediction experiments are located in the `06_predict_survival` directory. The Python environment was managed using `conda`, and directions for setting up the environment can be found in the `README.md` file. Most analyses were run on the HTC CPU cluster at the University

of Pittsburgh, except the neural networks which were trained and evaluated on the PMACS LPC GPU cluster at the University of Pennsylvania; scripts for training classifiers both locally for a single gene and on a Slurm cluster to reproduce the analysis of many genes in parallel are provided in the linked GitHub repo. This manuscript was written using Manubot [24] and is available on GitHub at <https://github.com/greenelab/mpmp-manuscript> under the CC0-1.0 license [95] and at Zenodo [96].

As a data resource, coefficients and hyperparameter choices for final models fit using all data types are available on Figshare: coefficients are available at <https://doi.org/10.6084/m9.figshare.19576012> [97] and hyperparameters are at <https://doi.org/10.6084/m9.figshare.19576048> [98]. File format/entries are described in the supplementary material in Additional File 1.

Acknowledgements

We would like to thank Alexandra Lee, Ariel Hippen, Ben Heil, Milton Pividori, and Natalie Davidson for reviewing the software associated with this work and providing insightful feedback. This research was supported in part by the University of Pittsburgh Center for Research Computing through the resources provided. Figure 1 (the schematic of the background and evaluation pipeline) was created using BioRender.com.

Supplementary material for “Widespread redundancy in -omics profiles of cancer mutation states”

A version of the main paper figures using the area under the receiver-operator curve (AUROC) metric rather than AUPR is available at <https://doi.org/10.6084/m9.figshare.14919729>.

In a previous version of this paper, we ran our analysis only for the genes in the Vogelstein et al. 2013 gene set. While there were some gene-to-gene differences in this set, we did not observe large differences between methylation and gene expression performances overall. Scaling up the gene set by combining cancer gene sets from the literature as described in the methods/results sections affected the study results somewhat, as mutations in the added genes tend to be better predicted using gene expression than other data types. During the revision, we explored the difference between the genes in this gene set and the genes in the “merged” cancer-related gene set but not in the Vogelstein genes. GO analysis results for the Vogelstein genes are available at <https://doi.org/10.6084/m9.figshare.19565890>, and results for the non-Vogelstein genes are available at <https://doi.org/10.6084/m9.figshare.19565887>. We noticed that the non-Vogelstein genes tend to be enriched for terms relating to transcription factors and transcriptional regulation.

As a data resource, coefficients and hyperparameter choices for final models fit using all data types are available on Figshare: coefficients are available at <https://doi.org/10.6084/m9.figshare.19576012> and hyperparameters are at <https://doi.org/10.6084/m9.figshare.19576048>. Columns in the coefficients dataset correspond to target genes (gene symbols), and rows correspond either to PCA components (for 27K and 450K methylation), -omics features (for all other data types), or covariates (cancer type indicator variables or log(mutation burden)). An ‘NA’ value in a cell indicates that feature was not used in the model for the corresponding gene (for an -omics feature this could mean it was not in the top 8000 features by MAD, for a cancer type feature this means that cancer type was not included in the training set based on our mutation filters). A 0 value in a cell indicates that feature was included in model training, but it was not selected by the elastic net feature selection algorithm. Columns in the

hyperparameters dataset correspond to hyperparameters (alpha and l1_ratio for elastic net logistic regression) and rows correspond to target genes. For the methylation data types, PCA results (score and loading matrices) corresponding to the coefficients data are also available at <https://doi.org/10.6084/m9.figshare.19908034>. These contain the top 5,000 principal components for each data type, which were used in the classifiers evaluated in the main paper.

Regarding the hyperparameters for the final models, recall that for the main figures in the paper, we evaluate each of our models using 2 replicates of 4-fold cross-validation. For each of these folds (train/test splits), we further split the training set into train and validation sets to select hyperparameters, independently for each fold, and evaluate the models on the test set to get the results in the paper. Because we are evaluating performance over multiple folds, it is not perfectly straightforward to get a single set of regression coefficients, since we have a (potentially different) set of coefficients for each cross-validation fold. In order to synthesize these results into a single model for each gene in each data type, we selected one of the 8 sets of hyperparameters (from the 8 best models, 1 per CV fold) at random, with probability proportional to performance (AUPR) on the validation set used to select the hyperparameters, described above (so test set performance is not used here). We then used the selected hyperparameters to train a single model on the entire dataset.

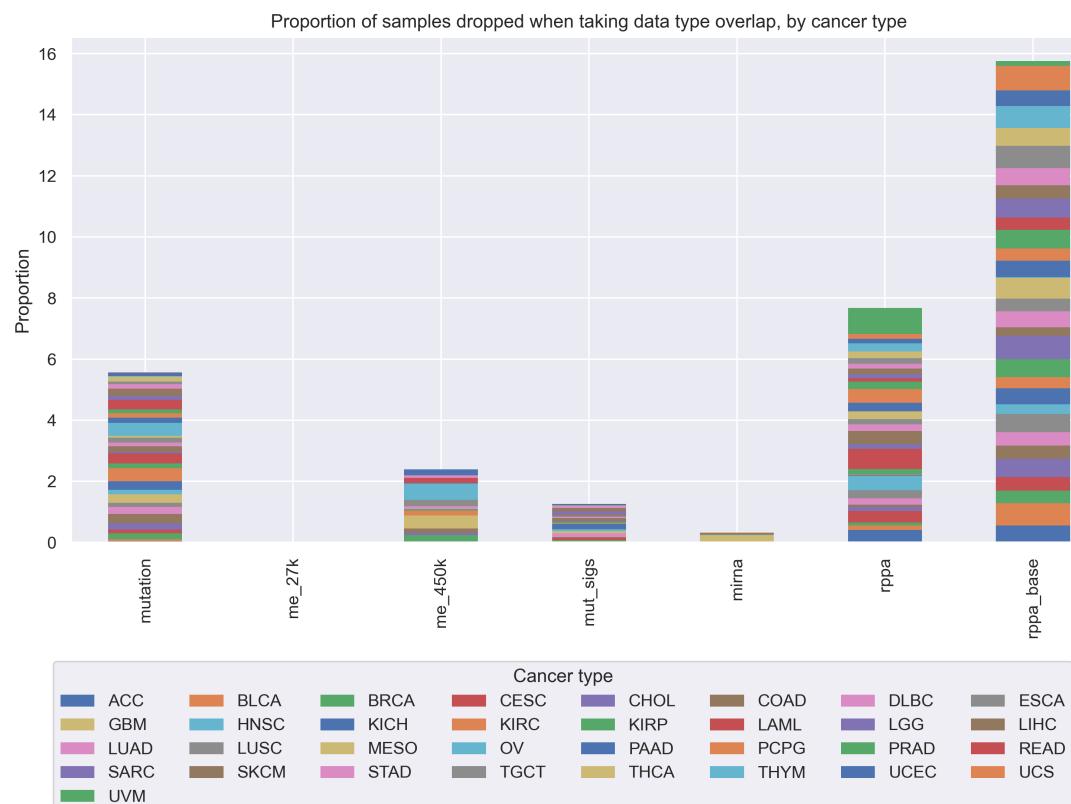


Figure 15: Proportion of samples from each TCGA cancer type that are “dropped” as more data types are added to our analyses. We started with gene expression data, and for each added data type, we took the intersection of samples that were profiled for that data type and the previous data types, dropping all samples that were missing 1 or more data types. Overall, at each step, the proportions of “dropped” samples appear to be fairly evenly spread between cancer types, showing that in general we are not disproportionately losing one or several cancer types as more data modalities are added to our analyses.

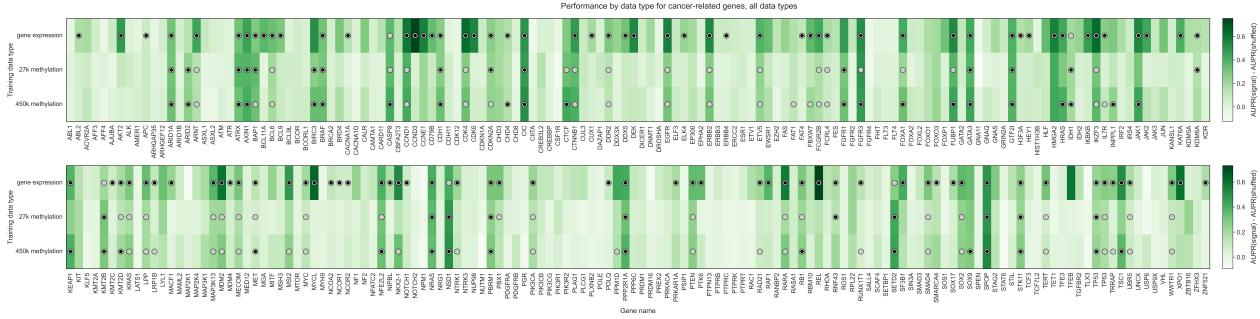


Figure 16: Heatmap displaying predictive performance for mutations in each of the 272 genes from the cancer-related gene set, across gene expression and the two DNA methylation arrays. Each cell quantifies performance for a target gene, using predictive features derived from a particular data type. Grey shaded dots indicate that the given data type provides significantly better predictions than the permuted baseline for the given gene; black inner dots indicate the same and additionally that the given data type provides statistically equivalent performance to the data type with the best average performance (determined by pairwise t -tests across data types with FDR correction).

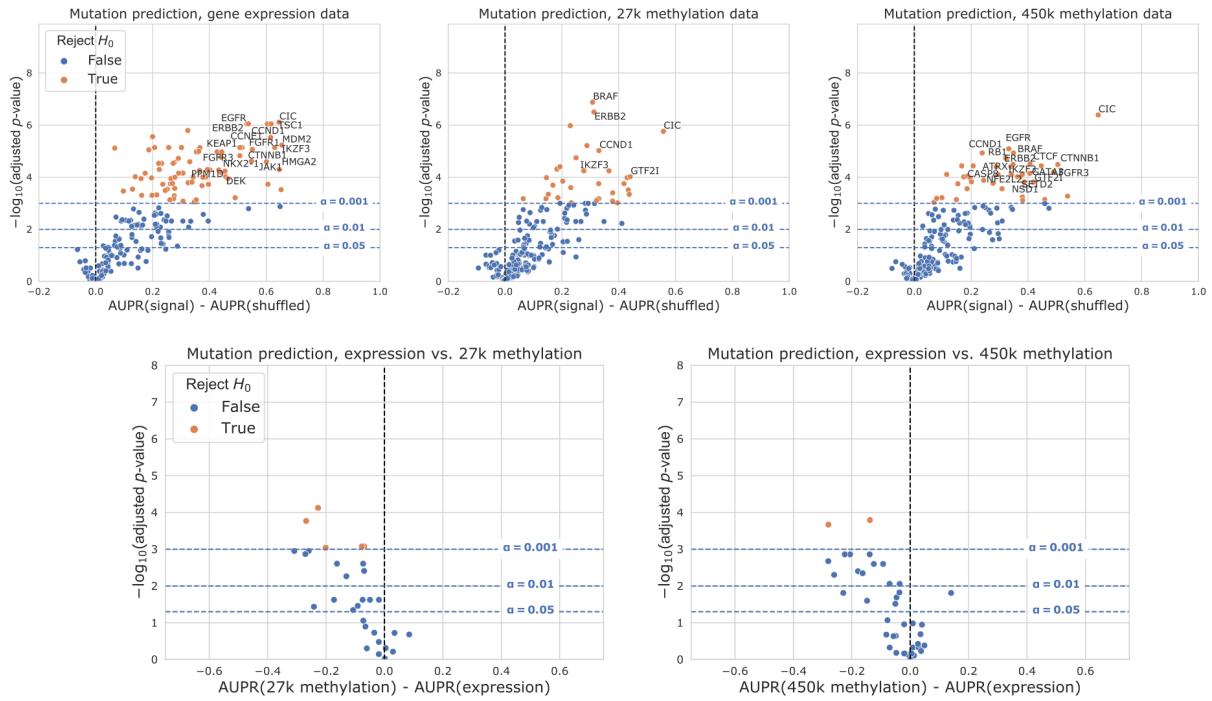


Figure 17: Volcano-like plots showing predictive performance for each gene in the cancer-related gene set for expression and DNA methylation, on the sample set used for the “all data types” experiments. The first row shows performance relative to the permuted baseline, and the second row shows direct comparisons between data types for genes that outperformed the permuted baseline only for both data types. The x-axis shows the difference in mean AUPR compared with a baseline model trained on permuted labels, and the y-axis shows p -values for a paired t -test comparing cross-validated AUPR values within folds.

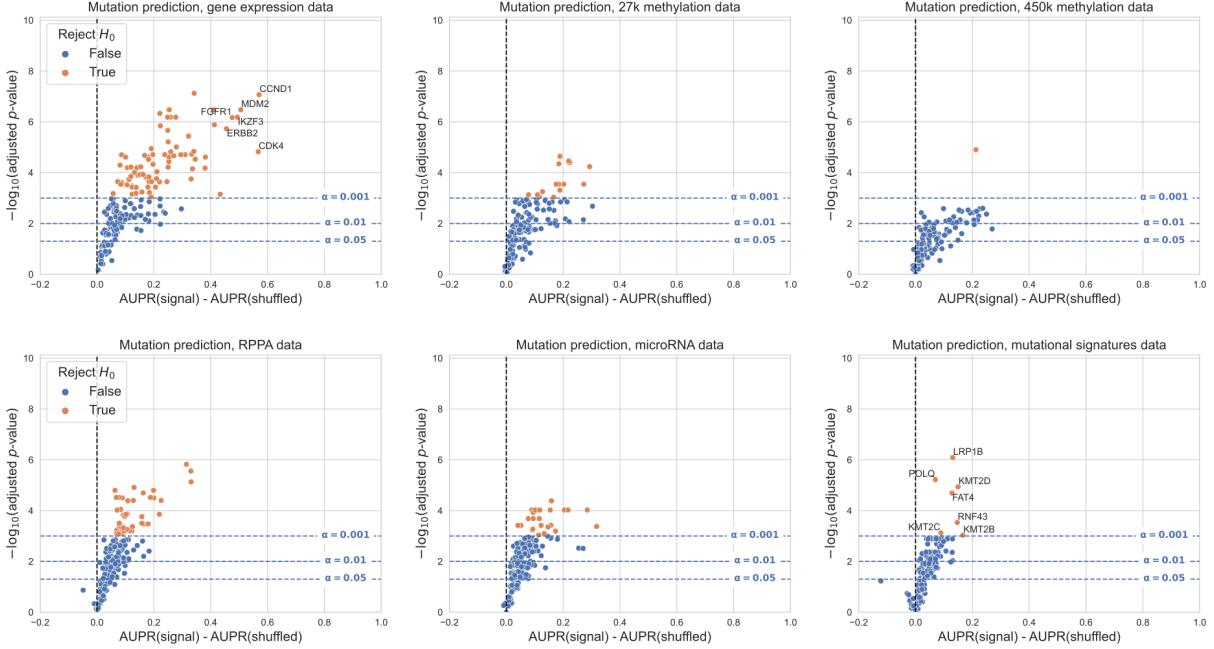


Figure 18: Volcano-like plots showing predictive performance for each gene in the cancer-related gene set for all data types, relative to the permuted baseline model, when genes are filtered based on the entire dataset rather than by cancer type. For this filtering approach, we included/excluded entire genes rather than individual cancer types: specifically, we trained a classifier for each gene where all cancer types combined had at least 5% mutated samples and at least 100 total mutated samples, resulting in 182 total classifiers. The x-axis shows the difference in mean AUPR compared with a baseline model trained on permuted labels, and the y-axis shows p -values for a paired t -test comparing cross-validated AUPR values within folds. Counts of genes making the significance threshold of 0.001: gene expression 81/182 (44.5%), 27K methylation 16/182 (8.8%), 450K methylation 1/182 (0.6%), RPPA 41/182 (22.5%), microRNA 25/182 (13.7%), mutational signatures 7/182 (3.9%).

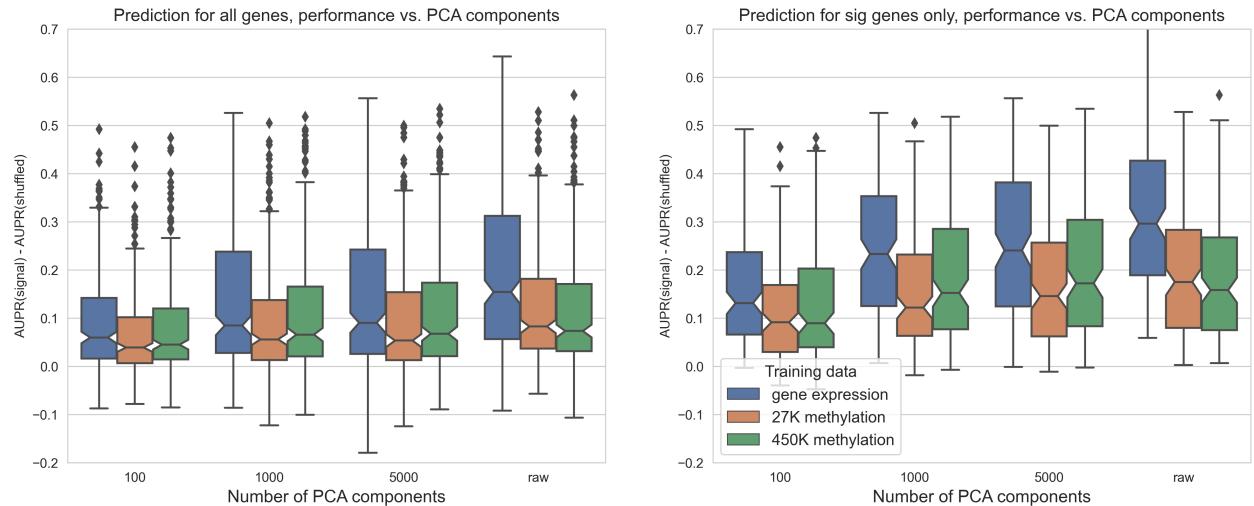


Figure 19: Predictive performance for genes in the cancer-related gene set, using each of the three data types as predictors. The x-axis shows the number of PCA components used as features, "raw" = no PCA compression.

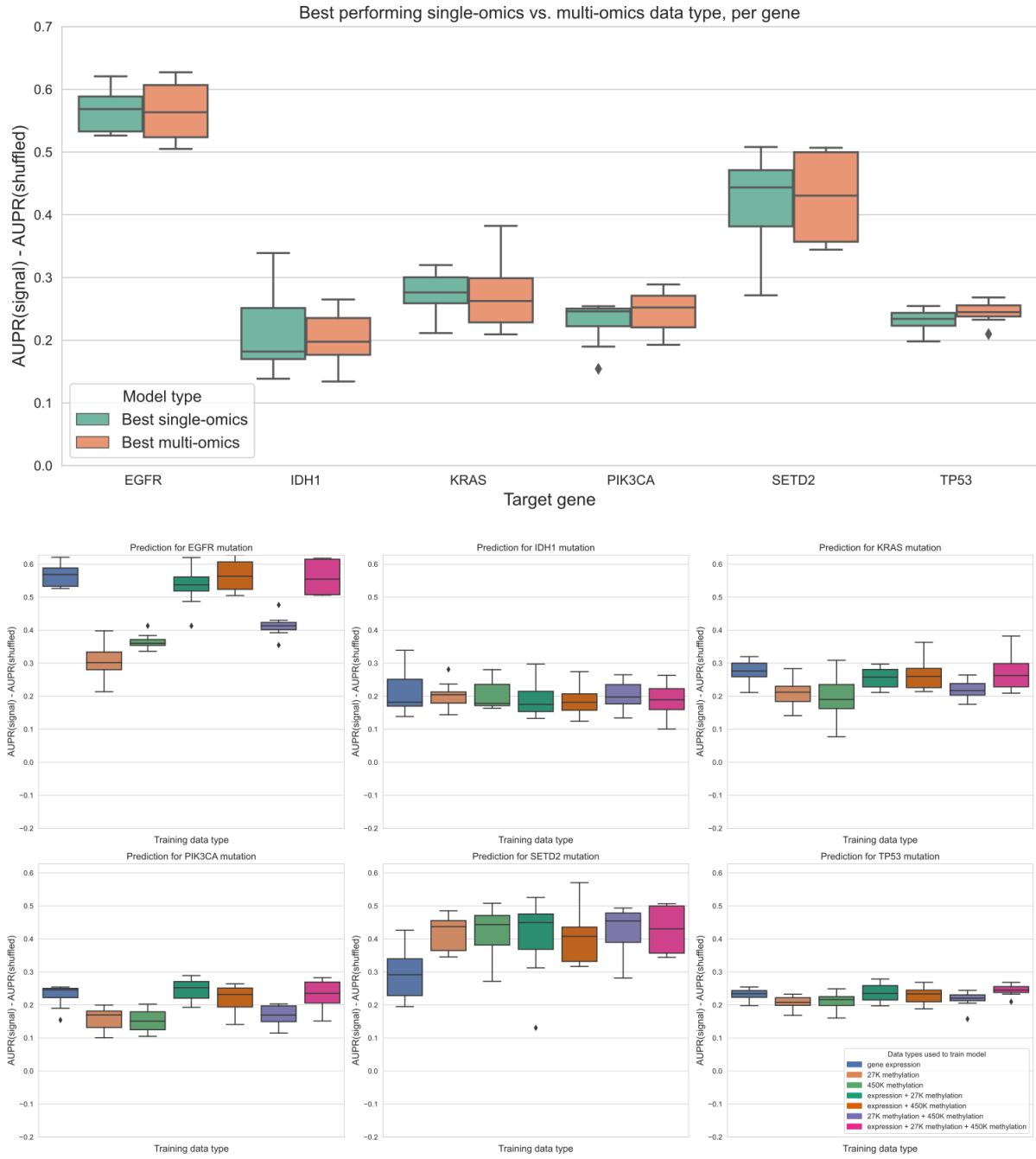


Figure 20: Top plot: comparing the best-performing model (i.e. highest mean AUPR relative to permuted baseline) trained on a single data type against the best “multi-omics” model for each target gene, using raw (not PCA compressed) features. For feature parity between data types, the top 20,000 features by mean absolute deviation were used for each data type. The difference between single-omics and multi-omics performance for *TP53* was statistically significant ($p=0.0078$), but other differences between single-omics and multi-omics models were not statistically significant using paired-sample Wilcoxon tests across cross-validation folds, for a threshold of 0.05. Bottom plots: classifier performance, relative to baseline with permuted labels, for individual genes. Each panel shows performance for one of the six target genes; box plots show performance distribution over 8 evaluation sets (4 cross-validation folds x 2 replicates).

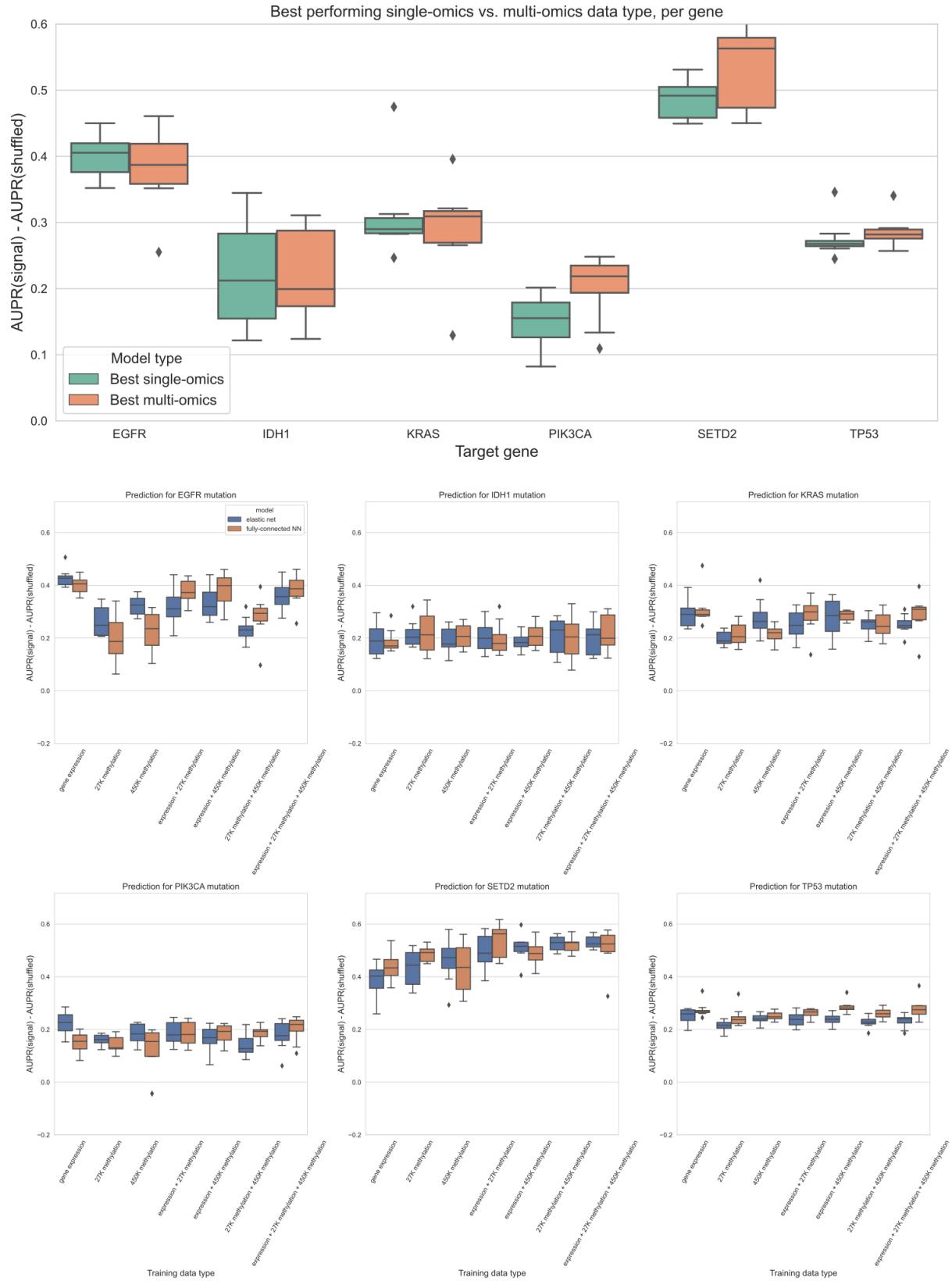


Figure 21: Top plot: comparing the best-performing model (i.e. highest mean AUPR relative to permuted baseline) trained on a single data type against the best “multi-omics” model for each target gene, using a 3-layer fully-connected neural network. The top 5,000 principal components were used as predictive features for each data type. The difference between single-omics and multi-omics performance for *PIK3CA* ($p = 0.0156$, in favor of multi-omics) and *TP53* ($p = 0.0391$, in favor of single-omics) were statistically significant, but other differences between single-omics and multi-omics models were not statistically significant using paired-sample Wilcoxon tests across cross-validation folds, for a threshold of 0.05. Bottom plots: comparison of classifier performance between elastic net and fully-connected NN, relative to baseline with permuted labels, for individual genes. Each panel shows performance for one of the six target genes; box plots show performance distribution over 8 evaluation sets (4 cross-validation folds x 2 replicates).

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](https://doi.org/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](https://doi.org/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: [10.1200/jco.2008.18.1370](https://doi.org/10.1200/jco.2008.18.1370) · PMID: [19204204](#) · PMCID: [PMC2667820](#)
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: [10.1016/s0960-9776\(09\)70290-5](https://doi.org/10.1016/s0960-9776(09)70290-5)
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](https://doi.org/10.1126/science.1235122) · PMID: [23539594](#) · PMCID: [PMC3749880](#)
9. **Scalable Open Science Approach for Mutation Calling of Tumor Exomes Using Multiple Genomic Pipelines**
Kyle Ellrott, Matthew H Bailey, Gordon Saksena, Kyle R Covington, Cyriac Kandoth, Chip Stewart, Julian Hess, Singer Ma, Kami E Chiotti, Michael McLellan, ... Armaz Mariamidze
Cell Systems (2018-03) <https://doi.org/gf9twn>
DOI: [10.1016/j.cels.2018.03.002](https://doi.org/10.1016/j.cels.2018.03.002) · PMID: [29596782](#) · PMCID: [PMC6075717](#)
10. **GISTIC2.0 facilitates sensitive and confident localization of the targets of focal somatic copy-number alteration in human cancers**

- Craig H Mermel, Steven E Schumacher, Barbara Hill, Matthew L Meyerson, Rameen Beroukhim, Gad Getz
Genome Biology (2011-04) <https://doi.org/dzhjqh>
DOI: [10.1186/gb-2011-12-4-r41](https://doi.org/10.1186/gb-2011-12-4-r41) · PMID: [21527027](#) · PMCID: [PMC3218867](#)
11. **Widespread redundancy in -omics profiles of cancer mutation states**
Jake Crawford, Brock C Christensen, Maria Chikina, Casey S Greene
Genome Biology (2022-06-27) <https://doi.org/gqfqnm>
DOI: [10.1186/s13059-022-02705-y](https://doi.org/10.1186/s13059-022-02705-y) · PMID: [35761387](#) · PMCID: [PMC9238138](#)
12. **Regression Shrinkage and Selection Via the Lasso**
Robert Tibshirani
Journal of the Royal Statistical Society: Series B (Methodological) (1996-01)
<https://doi.org/gfn45m>
DOI: [10.1111/j.2517-6161.1996.tb02080.x](https://doi.org/10.1111/j.2517-6161.1996.tb02080.x)
13. **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](https://doi.org/10.1016/j.celrep.2018.03.046) · PMID: [29617658](#) · PMCID: [PMC5918694](#)
14. **The Benefits of Implicit Regularization from SGD in Least Squares Problems**
Difan Zou, Jingfeng Wu, Vladimir Braverman, Quanquan Gu, Dean P Foster, Sham M Kakade
arXiv (2022-07-12) <https://arxiv.org/abs/2108.04552>
15. **Can Implicit Bias Explain Generalization? Stochastic Convex Optimization as a Case Study**
Assaf Dauber, Meir Feder, Tomer Koren, Roi Livni
arXiv (2020-12-23) <https://arxiv.org/abs/2003.06152>
16. **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>
17. **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](https://doi.org/10.1145/3446776)
18. **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](https://doi.org/10.1073/pnas.1907378117) · PMID: [32332161](#) · PMCID: [PMC7720150](#)
19. **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>
20. **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](https://doi.org/10.18637/jss.v033.i01)

21. **Evaluating cancer cell line and patient-derived xenograft recapitulation of tumor and non-diseased tissue gene expression profiles<i>in silico</i>**
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](https://doi.org/10.1101/2023.04.11.536431) · PMID: [37090499](https://pubmed.ncbi.nlm.nih.gov/37090499/) · PMCID: [PMC10120639](https://pubmed.ncbi.nlm.nih.gov/PMC10120639/)
22. **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: [10.1101/2020.02.21.959627](https://doi.org/10.1101/2020.02.21.959627)
23. **The Cancer Genome Atlas Pan-Cancer analysis project**
John N Weinstein, Eric A Collisson, Gordon B Mills, Kenna RMills Shaw, Brad A Ozenberger, Kyle Ellrott, Ilya Shmulevich, Chris Sander, Joshua M Stuart
Nature Genetics (2013-09-26) <https://doi.org/f3nt5c>
DOI: [10.1038/ng.2764](https://doi.org/10.1038/ng.2764) · PMID: [24071849](https://pubmed.ncbi.nlm.nih.gov/24071849/) · PMCID: [PMC3919969](https://pubmed.ncbi.nlm.nih.gov/PMC3919969/)
24. **Open collaborative writing with Manubot**
Daniel S Himmelstein, Vincent Rubinetti, David R Slochower, Dongbo Hu, Venkat S Malladi, Casey S Greene, Anthony Gitter
PLOS Computational Biology (2019-06-24) <https://doi.org/c7np>
DOI: [10.1371/journal.pcbi.1007128](https://doi.org/10.1371/journal.pcbi.1007128) · PMID: [31233491](https://pubmed.ncbi.nlm.nih.gov/31233491/) · PMCID: [PMC6611653](https://pubmed.ncbi.nlm.nih.gov/PMC6611653/)
25. **Oncogenic Signaling Pathways in The Cancer Genome Atlas**
Francisco Sanchez-Vega, Marco Mina, Joshua Armenia, Walid K Chatila, Augustin Luna, Konnor C La, Sofia Dimitriadiy, David L Liu, Havish S Kantheti, Sadegh Saghafinia, ... Armaz Mariamidze
Cell (2018-04) <https://doi.org/gc7r9b>
DOI: [10.1016/j.cell.2018.03.035](https://doi.org/10.1016/j.cell.2018.03.035) · PMID: [29625050](https://pubmed.ncbi.nlm.nih.gov/29625050/) · PMCID: [PMC6070353](https://pubmed.ncbi.nlm.nih.gov/PMC6070353/)
26. **Systematic identification of mutations and copy number alterations associated with cancer patient prognosis**
Joan C Smith, Jason M Sheltzer
eLife (2018-12-11) <https://doi.org/gf4zgg>
DOI: [10.7554/elife.39217](https://doi.org/10.7554/elife.39217) · PMID: [30526857](https://pubmed.ncbi.nlm.nih.gov/30526857/) · PMCID: [PMC6289580](https://pubmed.ncbi.nlm.nih.gov/PMC6289580/)
27. **Challenges in identifying cancer genes by analysis of exome sequencing data**
Matan Hofree, Hannah Carter, Jason F Kreisberg, Sourav Bandyopadhyay, Paul S Mischel, Stephen Friend, Trey Ideker
Nature Communications (2016-07-15) <https://doi.org/f8x7t3>
DOI: [10.1038/ncomms12096](https://doi.org/10.1038/ncomms12096) · PMID: [27417679](https://pubmed.ncbi.nlm.nih.gov/27417679/) · PMCID: [PMC4947162](https://pubmed.ncbi.nlm.nih.gov/PMC4947162/)
28. **Evaluating the evaluation of cancer driver genes**
Collin J Tokheim, Nickolas Papadopoulos, Kenneth W Kinzler, Bert Vogelstein, Rachel Karchin
Proceedings of the National Academy of Sciences (2016-11-22) <https://doi.org/f9d77w>
DOI: [10.1073/pnas.1616440113](https://doi.org/10.1073/pnas.1616440113) · PMID: [27911828](https://pubmed.ncbi.nlm.nih.gov/27911828/) · PMCID: [PMC5167163](https://pubmed.ncbi.nlm.nih.gov/PMC5167163/)
29. **Detailed modeling of positive selection improves detection of cancer driver genes**
Siming Zhao, Jun Liu, Pranav Nanga, Yuwen Liu, AErcument Cicek, Nicholas Knoblauch, Chuan He, Matthew Stephens, Xin He
Nature Communications (2019-07-30) <https://doi.org/gjmhnn>
DOI: [10.1038/s41467-019-11284-9](https://doi.org/10.1038/s41467-019-11284-9) · PMID: [31363082](https://pubmed.ncbi.nlm.nih.gov/31363082/) · PMCID: [PMC6667447](https://pubmed.ncbi.nlm.nih.gov/PMC6667447/)

30. **Review: Precision medicine and driver mutations: Computational methods, functional assays and conformational principles for interpreting cancer drivers**
Ruth Nussinov, Hyunbum Jang, Chung-Jung Tsai, Feixiong Cheng
PLOS Computational Biology (2019-03-28) <https://doi.org/gg8jhm>
DOI: [10.1371/journal.pcbi.1006658](https://doi.org/10.1371/journal.pcbi.1006658) · PMID: [30921324](#) · PMCID: [PMC6438456](#)
31. **Modeling RAS Phenotype in Colorectal Cancer Uncovers Novel Molecular Traits of RAS Dependency and Improves Prediction of Response to Targeted Agents in Patients**
Justin Guinney, Charles Ferté, Jonathan Dry, Robert McEwen, Gilles Manceau, KJ Kao, Kai-Ming Chang, Claus Bendtsen, Kevin Hudson, Erich Huang, ... Pierre Laurent-Puig
Clinical Cancer Research (2014-01-01) <https://doi.org/f5njhn>
DOI: [10.1158/1078-0432.ccr-13-1943](https://doi.org/10.1158/1078-0432.ccr-13-1943) · PMID: [24170544](#) · PMCID: [PMC4141655](#)
32. **Identification of pan-cancer Ras pathway activation with deep learning**
Xiangtao Li, Shaochuan Li, Yunhe Wang, Shixiong Zhang, Ka-Chun Wong
Briefings in Bioinformatics (2020-10-30) <https://doi.org/gjmd3p>
DOI: [10.1093/bib/bbaa258](https://doi.org/10.1093/bib/bbaa258)
33. **Genomic and Molecular Landscape of DNA Damage Repair Deficiency across The Cancer Genome Atlas**
Theo A Knijnenburg, Linghua Wang, Michael T Zimmermann, Nyasha Chambwe, Galen F Gao, Andrew D Cherniack, Huihui Fan, Hui Shen, Gregory P Way, Casey S Greene, ... Armaz Mariamidze
Cell Reports (2018-04) <https://doi.org/gfspsc>
DOI: [10.1016/j.celrep.2018.03.076](https://doi.org/10.1016/j.celrep.2018.03.076) · PMID: [29617664](#) · PMCID: [PMC5961503](#)
34. **Prediction of PIK3CA mutations from cancer gene expression data**
Jun Kang, Ahwon Lee, Youn Soo Lee
PLOS ONE (2020-11-09) <https://doi.org/gjmd3s>
DOI: [10.1371/journal.pone.0241514](https://doi.org/10.1371/journal.pone.0241514) · PMID: [33166334](#) · PMCID: [PMC7652327](#)
35. **Compressing gene expression data using multiple latent space dimensionalities learns complementary biological representations**
Gregory P Way, Michael Zietz, Vincent Rubinetti, Daniel S Himmelstein, Casey S Greene
Genome Biology (2020-05-11) <https://doi.org/gg2mjh>
DOI: [10.1186/s13059-020-02021-3](https://doi.org/10.1186/s13059-020-02021-3) · PMID: [32393369](#) · PMCID: [PMC7212571](#)
36. **Systematic interrogation of mutation groupings reveals divergent downstream expression programs within key cancer genes**
Michał R Grzadkowski, Hannah Manning, Julia Somers, Emek Demir
Cold Spring Harbor Laboratory (2020-06-03) <https://doi.org/gjmd7t>
DOI: [10.1101/2020.06.02.128850](https://doi.org/10.1101/2020.06.02.128850)
37. **Using Transcriptional Signatures to Find Cancer Drivers with LURE**
David Haan, Ruikang Tao, Verena Friedl, Ioannis N Anastopoulos, Christopher K Wong, Alana S Weinstein, Joshua M Stuart
Biocomputing 2020 (2019-11-27) <https://doi.org/gjmd4t>
DOI: [10.1142/9789811215636_0031](https://doi.org/10.1142/9789811215636_0031)
38. **Reverse regression increases power for detecting trans-eQTLs**
Saikat Banerjee, Franco L Simonetti, Kira E Detrois, Anubhav Kaphele, Raktim Mitra, Rahul Nagial, Johannes Söding
Cold Spring Harbor Laboratory (2020-05-09) <https://doi.org/gjmhd8>
DOI: [10.1101/2020.05.07.083386](https://doi.org/10.1101/2020.05.07.083386)

39. **Cancer transcriptome profiling at the juncture of clinical translation**
Marcin Cieślik, Arul M Chinnaiyan
Nature Reviews Genetics (2017-12-27) <https://doi.org/gcsmnr>
DOI: [10.1038/nrg.2017.96](https://doi.org/10.1038/nrg.2017.96)
40. **Identification of a CpG Island Methylator Phenotype that Defines a Distinct Subgroup of Glioma**
Houtan Noushmehr, Daniel J Weisenberger, Kristin Diefes, Heidi S Phillips, Kanan Pujara, Benjamin P Berman, Fei Pan, Christopher E Pelloski, Erik P Sulman, Krishna P Bhat, ... Kenneth Aldape
Cancer Cell (2010-05) <https://doi.org/dbtmsd>
DOI: [10.1016/j.ccr.2010.03.017](https://doi.org/10.1016/j.ccr.2010.03.017) · PMID: [20399149](#) · PMCID: [PMC2872684](#)
41. **DNA Methylation, Isocitrate Dehydrogenase Mutation, and Survival in Glioma**
Brock C Christensen, Ashley A Smith, Shichun Zheng, Devin C Koestler, EAndres Houseman, Carmen J Marsit, Joseph L Wiemels, Heather H Nelson, Margaret R Karagas, Margaret R Wrensch, ... John K Wiencke
JNCI: Journal of the National Cancer Institute (2011-01-19) <https://doi.org/bbjqf9>
DOI: [10.1093/jnci/djq497](https://doi.org/10.1093/jnci/djq497) · PMID: [21163902](#) · PMCID: [PMC3022619](#)
42. **< i>IDH1</i>and< i>IDH2</i>Mutations in Gliomas**
Hai Yan, DWilliams Parsons, Genglin Jin, Roger McLendon, BAhmed Rasheed, Weishi Yuan, Ivan Kos, Ines Batinic-Haberle, Siân Jones, Gregory J Riggins, ... Darell D Bigner
New England Journal of Medicine (2009-02-19) <https://doi.org/btz6db>
DOI: [10.1056/nejmoa0808710](https://doi.org/10.1056/nejmoa0808710) · PMID: [19228619](#) · PMCID: [PMC2820383](#)
43. **IDH mutation impairs histone demethylation and results in a block to cell differentiation**
Chao Lu, Patrick S Ward, Gurpreet S Kapoor, Dan Rohle, Sevin Turcan, Omar Abdel-Wahab, Christopher R Edwards, Raya Khanin, Maria E Figueroa, Ari Melnick, ... Craig B Thompson
Nature (2012-02-15) <https://doi.org/f4msnt>
DOI: [10.1038/nature10860](https://doi.org/10.1038/nature10860) · PMID: [22343901](#) · PMCID: [PMC3478770](#)
44. **Connections between TET proteins and aberrant DNA modification in cancer**
Yun Huang, Anjana Rao
Trends in Genetics (2014-10) <https://doi.org/f6jm7v>
DOI: [10.1016/j.tig.2014.07.005](https://doi.org/10.1016/j.tig.2014.07.005) · PMID: [25132561](#) · PMCID: [PMC4337960](#)
45. **SETting the Stage for Cancer Development: SETD2 and the Consequences of Lost Methylation**
Catherine C Fahey, Ian J Davis
Cold Spring Harbor Perspectives in Medicine (2017-02-03) <https://doi.org/gjmfvq>
DOI: [10.1101/cshperspect.a026468](https://doi.org/10.1101/cshperspect.a026468) · PMID: [28159833](#) · PMCID: [PMC5411680](#)
46. **Mechanisms underlying mutational signatures in human cancers**
Thomas Helleday, Saeed Eshtad, Serena Nik-Zainal
Nature Reviews Genetics (2014-07-01) <https://doi.org/f25gnp>
DOI: [10.1038/nrg3729](https://doi.org/10.1038/nrg3729) · PMID: [24981601](#) · PMCID: [PMC6044419](#)
47. **Quantitative Proteomics of the Cancer Cell Line Encyclopedia**
David P Nusinow, John Szpyt, Mahmoud Ghandi, Christopher M Rose, ERobert McDonald III, Marian Kalocsay, Judit Jané-Valbuena, Ellen Gelfand, Devin K Schweppe, Mark Jedrychowski, ... Steven P Gygi
Cell (2020-01) <https://doi.org/ggxbh5>
DOI: [10.1016/j.cell.2019.12.023](https://doi.org/10.1016/j.cell.2019.12.023) · PMID: [31978347](#) · PMCID: [PMC7339254](#)

48. **The repertoire of mutational signatures in human cancer**
Ludmil B Alexandrov, Jaegil Kim, Nicholas J Haradhvala, Mi Ni Huang, Alvin Wei Tian Ng, Yang Wu, Arnoud Boot, Kyle R Covington, Dmitry A Gordenin, Erik N Bergstrom, ...
Nature (2020-02-05) <https://doi.org/ggkfnv>
DOI: [10.1038/s41586-020-1943-3](https://doi.org/10.1038/s41586-020-1943-3) · PMID: [32025018](https://pubmed.ncbi.nlm.nih.gov/32025018/) · PMCID: [PMC7054213](https://pubmed.ncbi.nlm.nih.gov/PMC7054213/)
49. **Significant associations between driver gene mutations and DNA methylation alterations across many cancer types**
Yun-Ching Chen, Valer Gotea, Gennady Margolin, Laura Elnitski
PLOS Computational Biology (2017-11-10) <https://doi.org/gchz8h>
DOI: [10.1371/journal.pcbi.1005840](https://doi.org/10.1371/journal.pcbi.1005840) · PMID: [29125844](https://pubmed.ncbi.nlm.nih.gov/29125844/) · PMCID: [PMC5709060](https://pubmed.ncbi.nlm.nih.gov/PMC5709060/)
50. **A pan-cancer analysis of driver gene mutations, DNA methylation and gene expressions reveals that chromatin remodeling is a major mechanism inducing global changes in cancer epigenomes**
Ahrim Youn, Kyung In Kim, Raul Rabadan, Benjamin Tycko, Yufeng Shen, Shuang Wang
BMC Medical Genomics (2018-11-06) <https://doi.org/gjmhfb>
DOI: [10.1186/s12920-018-0425-z](https://doi.org/10.1186/s12920-018-0425-z) · PMID: [30400878](https://pubmed.ncbi.nlm.nih.gov/30400878/) · PMCID: [PMC6218985](https://pubmed.ncbi.nlm.nih.gov/PMC6218985/)
51. **Computational analysis reveals histotype-dependent molecular profile and actionable mutation effects across cancers**
Daniel Heim, Grégoire Montavon, Peter Hufnagl, Klaus-Robert Müller, Frederick Klauschen
Genome Medicine (2018-11-15) <https://doi.org/gjmhfc>
DOI: [10.1186/s13073-018-0591-9](https://doi.org/10.1186/s13073-018-0591-9) · PMID: [30442178](https://pubmed.ncbi.nlm.nih.gov/30442178/) · PMCID: [PMC6238410](https://pubmed.ncbi.nlm.nih.gov/PMC6238410/)
52. **CNAmet: an R package for integrating copy number, methylation and expression data**
Riku Louhimo, Sampsa Hautaniemi
Bioinformatics (2011-01-12) <https://doi.org/fbq4p2>
DOI: [10.1093/bioinformatics/btr019](https://doi.org/10.1093/bioinformatics/btr019)
53. **Impacts of somatic mutations on gene expression: an association perspective**
Peilin Jia, Zhongming Zhao
Briefings in Bioinformatics (2016-04-28) <https://doi.org/gjnd5b>
DOI: [10.1093/bib/bbw037](https://doi.org/10.1093/bib/bbw037) · PMID: [27127206](https://pubmed.ncbi.nlm.nih.gov/27127206/) · PMCID: [PMC5862283](https://pubmed.ncbi.nlm.nih.gov/PMC5862283/)
54. **Inference of patient-specific pathway activities from multi-dimensional cancer genomics data using PARADIGM**
Charles J Vaske, Stephen C Benz, JZachary Sanborn, Dent Earl, Christopher Szeto, Jingchun Zhu, David Haussler, Joshua M Stuart
Bioinformatics (2010-06-01) <https://doi.org/bcvgf>
DOI: [10.1093/bioinformatics/btq182](https://doi.org/10.1093/bioinformatics/btq182) · PMID: [20529912](https://pubmed.ncbi.nlm.nih.gov/20529912/) · PMCID: [PMC2881367](https://pubmed.ncbi.nlm.nih.gov/PMC2881367/)
55. **Systematic analysis of somatic mutations impacting gene expression in 12 tumour types**
Jiarui Ding, Melissa K McConechy, Hugo M Horlings, Gavin Ha, Fong Chun Chan, Tyler Funnell, Sarah C Mullaly, Jüri Reimand, Ali Bashashati, Gary D Bader, ... Sohrab P Shah
Nature Communications (2015-10-05) <https://doi.org/f7z86p>
DOI: [10.1038/ncomms9554](https://doi.org/10.1038/ncomms9554) · PMID: [26436532](https://pubmed.ncbi.nlm.nih.gov/26436532/) · PMCID: [PMC4600750](https://pubmed.ncbi.nlm.nih.gov/PMC4600750/)
56. **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>

57. **A beta-mixture quantile normalization method for correcting probe design bias in Illumina Infinium 450 k DNA methylation data**
Andrew E Teschendorff, Francesco Marabita, Matthias Lechner, Thomas Bartlett, Jesper Tegner, David Gomez-Cabrero, Stephan Beck
Bioinformatics (2012-11-21) <https://doi.org/f25mvt>
DOI: [10.1093/bioinformatics/bts680](https://doi.org/10.1093/bioinformatics/bts680) · PMID: [23175756](https://pubmed.ncbi.nlm.nih.gov/23175756/) · PMCID: [PMC3546795](https://pubmed.ncbi.nlm.nih.gov/PMC3546795/)
58. **Comprehensive Characterization of Cancer Driver Genes and Mutations**
Matthew H Bailey, Collin Tokheim, Eduard Porta-Pardo, Sohini Sengupta, Denis Bertrand, Amila Weerasinghe, Antonio Colaprico, Michael C Wendl, Jaegil Kim, Brendan Reardon, ... Armaz Mariamidze
Cell (2018-04) <https://doi.org/gc7r88>
DOI: [10.1016/j.cell.2018.02.060](https://doi.org/10.1016/j.cell.2018.02.060) · PMID: [29625053](https://pubmed.ncbi.nlm.nih.gov/29625053/) · PMCID: [PMC6029450](https://pubmed.ncbi.nlm.nih.gov/PMC6029450/)
59. **The COSMIC Cancer Gene Census: describing genetic dysfunction across all human cancers**
Zbyslaw Sondka, Sally Bamford, Charlotte G Cole, Sari A Ward, Ian Dunham, Simon A Forbes
Nature Reviews Cancer (2018-10-06) <https://doi.org/gfb3z4>
DOI: [10.1038/s41568-018-0060-1](https://doi.org/10.1038/s41568-018-0060-1) · PMID: [30293088](https://pubmed.ncbi.nlm.nih.gov/30293088/) · PMCID: [PMC6450507](https://pubmed.ncbi.nlm.nih.gov/PMC6450507/)
60. **Regularization and Variable Selection Via the Elastic Net**
Hui Zou, Trevor Hastie
Journal of the Royal Statistical Society Series B: Statistical Methodology (2005-03-09) <https://doi.org/b8cwrr>
DOI: [10.1111/j.1467-9868.2005.00503.x](https://doi.org/10.1111/j.1467-9868.2005.00503.x)
61. **An introduction to ROC analysis**
Tom Fawcett
Pattern Recognition Letters (2006-06) <https://doi.org/bpsghb>
DOI: [10.1016/j.patrec.2005.10.010](https://doi.org/10.1016/j.patrec.2005.10.010)
62. **A critical investigation of recall and precision as measures of retrieval system performance**
Vijay Raghavan, Peter Bollmann, Gwang S Jung
ACM Transactions on Information Systems (1989-07) <https://doi.org/bg4tps>
DOI: [10.1145/65943.65945](https://doi.org/10.1145/65943.65945)
63. **The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets**
Takaya Saito, Marc Rehmsmeier
PLOS ONE (2015-03-04) <https://doi.org/f69237>
DOI: [10.1371/journal.pone.0118432](https://doi.org/10.1371/journal.pone.0118432) · PMID: [25738806](https://pubmed.ncbi.nlm.nih.gov/25738806/) · PMCID: [PMC4349800](https://pubmed.ncbi.nlm.nih.gov/PMC4349800/)
64. **The MCC-F1 curve: a performance evaluation technique for binary classification**
Chang Cao, Davide Chicco, Michael M Hoffman
arXiv (2020-06-23) <https://arxiv.org/abs/2006.11278>
65. **Genome-wide identification and analysis of prognostic features in human cancers**
Joan C Smith, Jason M Sheltzer
Cold Spring Harbor Laboratory (2021-06-01) <https://doi.org/gmqfqt>
DOI: [10.1101/2021.06.01.446243](https://doi.org/10.1101/2021.06.01.446243)
66. **scikit-survival: A Library for Time-to-Event Analysis Built on Top of scikit-learn**
Sebastian Pölsterl
Journal of Machine Learning Research (2020) <http://jmlr.org/papers/v21/20-729.html>

67. **Adam: A Method for Stochastic Optimization**
Diederik P Kingma, Jimmy Ba
arXiv(2017-01-31) <https://arxiv.org/abs/1412.6980>
68. **Integrated genomic characterization of IDH1-mutant glioma malignant progression**
Hanwen Bai, Akdes Serin Harmancı, EZeynep Erson-Omay, Jie Li, Süleyman Coşkun, Matthias Simon, Boris Krischek, Koray Özduuman, SBülent Omay, Eric A Sorensen, ... Murat Günel
Nature Genetics(2015-11-30) <https://doi.org/f895hx>
DOI: [10.1038/ng.3457](https://doi.org/10.1038/ng.3457) · PMID: [26618343](https://pubmed.ncbi.nlm.nih.gov/26618343/) · PMCID: [PMC4829945](https://pubmed.ncbi.nlm.nih.gov/PMC4829945/)
69. **Identification and validation of an ERBB2 gene expression signature in breast cancers**
François Bertucci, Nathalie Borie, Christophe Ginestier, Agnès Groulet, Emmanuelle Charafe-Jauffret, José Adélaïde, Jeannine Geneix, Loïc Bachelart, Pascal Finetti, Alane Koki, ... Daniel Birnbaum
Oncogene(2004-01-26) <https://doi.org/dq9kvk>
DOI: [10.1038/sj.onc.1207361](https://doi.org/10.1038/sj.onc.1207361)
70. **Pan-Cancer Landscape and Analysis of ERBB2 Mutations Identifies Poziotinib as a Clinically Active Inhibitor and Enhancer of T-DM1 Activity**
Jacqueline P Robichaux, Yasir Y Elamin, RSK Vijayan, Monique B Nilsson, Lemei Hu, Junqin He, Fahao Zhang, Marlese Pisegna, Alissa Poteete, Huiying Sun, ... John V Heymach
Cancer Cell(2019-10) <https://doi.org/ggcv27>
DOI: [10.1016/j.ccr.2019.09.001](https://doi.org/10.1016/j.ccr.2019.09.001) · PMID: [31588020](https://pubmed.ncbi.nlm.nih.gov/31588020/) · PMCID: [PMC6944069](https://pubmed.ncbi.nlm.nih.gov/PMC6944069/)
71. **Shaping the cellular landscape with Set2/SETD2 methylation**
Stephen L McDaniel, Brian D Strahl
Cellular and Molecular Life Sciences(2017-04-06) <https://doi.org/gbrd9b>
DOI: [10.1007/s00018-017-2517-x](https://doi.org/10.1007/s00018-017-2517-x) · PMID: [28386724](https://pubmed.ncbi.nlm.nih.gov/28386724/) · PMCID: [PMC5545052](https://pubmed.ncbi.nlm.nih.gov/PMC5545052/)
72. **TCPA: a resource for cancer functional proteomics data**
Jun Li, Yiling Lu, Rehan Akbani, Zhenlin Ju, Paul L Roebuck, Wenbin Liu, Ji-Yeon Yang, Bradley M Broom, Roeland GW Verhaak, David W Kane, ... Han Liang
Nature Methods(2013-09-15) <https://doi.org/gffkjm>
DOI: [10.1038/nmeth.2650](https://doi.org/10.1038/nmeth.2650) · PMID: [24037243](https://pubmed.ncbi.nlm.nih.gov/24037243/) · PMCID: [PMC4076789](https://pubmed.ncbi.nlm.nih.gov/PMC4076789/)
73. **Patterns of somatic structural variation in human cancer genomes**
Yilong Li, Nicola D Roberts, Jeremiah A Wala, Ofer Shapira, Steven E Schumacher, Kiran Kumar, Ekta Khurana, Sebastian Waszak, Jan O Korbel, James E Haber, ...
Nature(2020-02-05) <https://doi.org/ggkfnw>
DOI: [10.1038/s41586-019-1913-9](https://doi.org/10.1038/s41586-019-1913-9) · PMID: [32025012](https://pubmed.ncbi.nlm.nih.gov/32025012/) · PMCID: [PMC7025897](https://pubmed.ncbi.nlm.nih.gov/PMC7025897/)
74. **Recruitment of KMT2C/MLL3 to DNA Damage Sites Mediates DNA Damage Responses and Regulates PARP Inhibitor Sensitivity in Cancer**
Antao Chang, Liang Liu, Justin M Ashby, Dan Wu, Yanan Chen, Stacey S O'Neill, Shan Huang, Juan Wang, Guanwen Wang, Dongmei Cheng, ... Peiqing Sun
Cancer Research(2021-04-14) <https://doi.org/gpt2z4>
DOI: [10.1158/0008-5472.can-21-0688](https://doi.org/10.1158/0008-5472.can-21-0688) · PMID: [33853832](https://pubmed.ncbi.nlm.nih.gov/33853832/) · PMCID: [PMC8260460](https://pubmed.ncbi.nlm.nih.gov/PMC8260460/)
75. **MethylNet: an automated and modular deep learning approach for DNA methylation analysis**
Joshua J Levy, Alexander J Titus, Curtis L Petersen, Youdinghuan Chen, Lucas A Salas, Brock C Christensen
BMC Bioinformatics(2020-03-17) <https://doi.org/ggxvrz>
DOI: [10.1186/s12859-020-3443-8](https://doi.org/10.1186/s12859-020-3443-8) · PMID: [32183722](https://pubmed.ncbi.nlm.nih.gov/32183722/) · PMCID: [PMC7076991](https://pubmed.ncbi.nlm.nih.gov/PMC7076991/)

76. **Predicting Drug Response and Synergy Using a Deep Learning Model of Human Cancer Cells**
Brent M Kuenzi, Jisoo Park, Samson H Fong, Kyle S Sanchez, John Lee, Jason F Kreisberg, Jianzhu Ma, Trey Ideker
Cancer Cell (2020-11) <https://doi.org/gh7z2n>
DOI: [10.1101/j.ccell.2020.09.014](https://doi.org/10.1101/j.ccell.2020.09.014) · PMID: [33096023](https://pubmed.ncbi.nlm.nih.gov/33096023/) · PMCID: [PMC7737474](https://pubmed.ncbi.nlm.nih.gov/PMC7737474/)
77. **Using Biological Constraints to Improve Prediction in Precision Oncology**
Mohamed Omar, Wikum Dinalankara, Lotte Mulder, Tendai Coady, Claudio Zanettini, Eddie Luidy Imada, Laurent Younes, Donald Geman, Luigi Marchionni
Cold Spring Harbor Laboratory (2021-05-27) <https://doi.org/gkcc43>
DOI: [10.1101/2021.05.25.445604](https://doi.org/10.1101/2021.05.25.445604)
78. **Reliable Identification of Genomic Variants from RNA-Seq Data**
Robert Piskol, Gokul Ramaswami, Jin Billy Li
The American Journal of Human Genetics (2013-10) <https://doi.org/f5fk6v>
DOI: [10.1101/j.ajhg.2013.08.008](https://doi.org/10.1101/j.ajhg.2013.08.008) · PMID: [24075185](https://pubmed.ncbi.nlm.nih.gov/24075185/) · PMCID: [PMC3791257](https://pubmed.ncbi.nlm.nih.gov/PMC3791257/)
79. **Detection and benchmarking of somatic mutations in cancer genomes using RNA-seq data**
Alexandre Coudray, Anna M Battenhouse, Philipp Bucher, Vishwanath R Iyer
PeerJ (2018-07-31) <https://doi.org/gd2f3q>
DOI: [10.7717/peerj.5362](https://doi.org/10.7717/peerj.5362) · PMID: [30083469](https://pubmed.ncbi.nlm.nih.gov/30083469/) · PMCID: [PMC6074801](https://pubmed.ncbi.nlm.nih.gov/PMC6074801/)
80. **Principles and methods of integrative genomic analyses in cancer**
Vessela N Kristensen, Ole Christian Lingjærde, Hege G Russnes, Hans Kristian M Vollan, Arnoldo Frigessi, Anne-Lise Børresen-Dale
Nature Reviews Cancer (2014-04-24) <https://doi.org/gf66jv>
DOI: [10.1038/nrc3721](https://doi.org/10.1038/nrc3721)
81. **Predicting censored survival data based on the interactions between meta-dimensional omics data in breast cancer**
Dokyoon Kim, Ruowang Li, Scott M Dudek, Marylyn D Ritchie
Journal of Biomedical Informatics (2015-08) <https://doi.org/f7n49h>
DOI: [10.1101/j.jbi.2015.05.019](https://doi.org/10.1101/j.jbi.2015.05.019) · PMID: [26048077](https://pubmed.ncbi.nlm.nih.gov/26048077/) · PMCID: [PMC4550096](https://pubmed.ncbi.nlm.nih.gov/PMC4550096/)
82. **Combining DNA methylation and RNA sequencing data of cancer for supervised knowledge extraction**
Eleonora Cappelli, Giovanni Felici, Emanuel Weitschek
BioData Mining (2018-10-25) <https://doi.org/gkcdbm>
DOI: [10.1186/s13040-018-0184-6](https://doi.org/10.1186/s13040-018-0184-6) · PMID: [30386434](https://pubmed.ncbi.nlm.nih.gov/30386434/) · PMCID: [PMC6203208](https://pubmed.ncbi.nlm.nih.gov/PMC6203208/)
83. **MOLI: multi-omics late integration with deep neural networks for drug response prediction**
Hossein Sharifi-Noghabi, Olga Zolotareva, Colin C Collins, Martin Ester
Bioinformatics (2019-07) <https://doi.org/fxbz>
DOI: [10.1093/bioinformatics/btz318](https://doi.org/10.1093/bioinformatics/btz318) · PMID: [31510700](https://pubmed.ncbi.nlm.nih.gov/31510700/) · PMCID: [PMC6612815](https://pubmed.ncbi.nlm.nih.gov/PMC6612815/)
84. **Accurate cancer phenotype prediction with AKLIMATE, a stacked kernel learner integrating multimodal genomic data and pathway knowledge**
Vladislav Uzunangelov, Christopher K Wong, Joshua M Stuart
PLOS Computational Biology (2021-04-16) <https://doi.org/gkcdbn>
DOI: [10.1371/journal.pcbi.1008878](https://doi.org/10.1371/journal.pcbi.1008878) · PMID: [33861732](https://pubmed.ncbi.nlm.nih.gov/33861732/) · PMCID: [PMC8081343](https://pubmed.ncbi.nlm.nih.gov/PMC8081343/)
85. **Methods for biological data integration: perspectives and challenges**

Vladimir Gligorijević, Nataša Pržulj
Journal of The Royal Society Interface (2015-11) <https://doi.org/bdzp>
DOI: [10.1098/rsif.2015.0571](https://doi.org/10.1098/rsif.2015.0571) · PMID: [26490630](https://pubmed.ncbi.nlm.nih.gov/26490630/) · PMCID: [PMC4685837](https://pubmed.ncbi.nlm.nih.gov/PMC4685837/)

86. **Machine learning for integrating data in biology and medicine: Principles, practice, and opportunities**
Marinka Zitnik, Francis Nguyen, Bo Wang, Jure Leskovec, Anna Goldenberg, Michael M Hoffman
Information Fusion (2019-10) <https://doi.org/gf7rj8>
DOI: [10.1016/j.inffus.2018.09.012](https://doi.org/10.1016/j.inffus.2018.09.012) · PMID: [30467459](https://pubmed.ncbi.nlm.nih.gov/30467459/) · PMCID: [PMC6242341](https://pubmed.ncbi.nlm.nih.gov/PMC6242341/)
87. **Integrated phosphoproteomics and transcriptional classifiers reveal hidden RAS signaling dynamics in multiple myeloma**
Yu-Hsiu T Lin, Gregory P Way, Benjamin G Barwick, Margarette C Mariano, Makeba Marcoulis, Ian D Ferguson, Christoph Driessens, Lawrence H Boise, Casey S Greene, Arun P Wiita
Blood Advances (2019-10-29) <https://doi.org/gg7m56>
DOI: [10.1182/bloodadvances.2019000303](https://doi.org/10.1182/bloodadvances.2019000303) · PMID: [31698452](https://pubmed.ncbi.nlm.nih.gov/31698452/) · PMCID: [PMC6855123](https://pubmed.ncbi.nlm.nih.gov/PMC6855123/)
88. **Next-generation characterization of the Cancer Cell Line Encyclopedia**
Mahmoud Ghandi, Franklin W Huang, Judit Jané-Valbuena, Gregory V Kryukov, Christopher C Lo, ERobert McDonald III, Jordi Barretina, Ellen T Gelfand, Craig M Bielski, Haoxin Li, ... William R Sellers
Nature (2019-05) <https://doi.org/gf2m3h>
DOI: [10.1038/s41586-019-1186-3](https://doi.org/10.1038/s41586-019-1186-3) · PMID: [31068700](https://pubmed.ncbi.nlm.nih.gov/31068700/) · PMCID: [PMC6697103](https://pubmed.ncbi.nlm.nih.gov/PMC6697103/)
89. **Comprehensive transcriptomic analysis of cell lines as models of primary tumors across 22 tumor types**
K Yu, B Chen, D Aran, J Charalel, C Yau, DM Wolf, LJ van 't Veer, AJ Butte, T Goldstein, M Sirota
Nature Communications (2019-08-08) <https://doi.org/ggh7t7>
DOI: [10.1038/s41467-019-11415-2](https://doi.org/10.1038/s41467-019-11415-2) · PMID: [31395879](https://pubmed.ncbi.nlm.nih.gov/31395879/) · PMCID: [PMC6687785](https://pubmed.ncbi.nlm.nih.gov/PMC6687785/)
90. **Clonal status of actionable driver events and the timing of mutational processes in cancer evolution**
Nicholas McGranahan, Francesco Favero, Elza C de Bruin, Nicolai Juul Birkbak, Zoltan Szallasi, Charles Swanton
Science Translational Medicine (2015-04-15) <https://doi.org/f7f83d>
DOI: [10.1126/scitranslmed.aaa1408](https://doi.org/10.1126/scitranslmed.aaa1408) · PMID: [25877892](https://pubmed.ncbi.nlm.nih.gov/25877892/) · PMCID: [PMC4636056](https://pubmed.ncbi.nlm.nih.gov/PMC4636056/)
91. **TCGA Pan-Cancer Atlas**
The Cancer Genome Atlas (TCGA) Research Network
Webpage (2022) <https://gdc.cancer.gov/about-data/publications/pancanatlas>
92. **ICGC/TCGA Mutational Signatures**
ICGC-TCGA Pan-Cancer Analysis Mutational Signatures Working Group
Webpage (2022) https://dcc.icgc.org/releases/PCAWG/mutational_signatures
93. **Code for 'Widespread redundancy in -omics profiles of cancer mutation states'**
Jake Crawford, Brock C. Christensen, Maria Chikina, Casey S. Greene
Zenodo (2022) <https://doi.org/10.5281/zenodo.6590133>
94. **Code for 'Widespread redundancy in -omics profiles of cancer mutation states'**
Jake Crawford, Brock C. Christensen, Maria Chikina, Casey S. Greene
GitHub (2022) <https://github.com/greenelab/mpmp>
95. **Widespread redundancy in -omics profiles of cancer mutation states**
Jake Crawford, Brock C. Christensen, Maria Chikina, Casey S. Greene
GitHub (2022) <https://github.com/greenelab/mpmp-manuscript>

96. **Widespread redundancy in -omics profiles of cancer mutation states**
Jake Crawford, Brock C. Christensen, Maria Chikina, Casey S. Greene
Zenodo (2022) <https://doi.org/10.5281/zenodo.6638831>
97. **Widespread redundancy in -omics profiles of cancer mutation states**
Jake Crawford, Brock C. Christensen, Maria Chikina, Casey S. Greene
figshare (2022) <https://doi.org/10.6084/m9.figshare.19576012>
98. **Widespread redundancy in -omics profiles of cancer mutation states**
Jake Crawford, Brock C. Christensen, Maria Chikina, Casey S. Greene
figshare (2022) <https://doi.org/10.6084/m9.figshare.19576048>