BMC Bioinformatics

RESEARCH

Open Access

Predicting chemotherapy response using a variational autoencoder approach

Qi Wei[1*] and Stephen A. Ramsey[2]

*Correspondence:
weiq@oregonstate.edu
[1] School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR, USA
Full list of author information is available at the end of the article

Abstract

Background: Multiple studies have shown the utility of transcriptome-wide RNA-seq profiles as features for machine learning-based prediction of response to chemotherapy in cancer. While tumor transcriptome profiles are publicly available for thousands of tumors for many cancer types, a relatively modest number of tumor profiles are clinically annotated for response to chemotherapy. The paucity of labeled examples and the high dimension of the feature data limit performance for predicting therapeutic response using fully-supervised classification methods. Recently, multiple studies have established the utility of a deep neural network approach, the variational autoencoder (VAE), for generating meaningful latent features from original data. Here, we report the first study of a semi-supervised approach using VAE-encoded tumor transcriptome features and regularized gradient boosted decision trees (XGBoost) to predict chemotherapy drug response for five cancer types: colon, pancreatic, bladder, breast, and sarcoma.

Results: We found: (1) VAE-encoding of the tumor transcriptome preserves the cancer type identity of the tumor, suggesting preservation of biologically relevant information; and (2) as a feature-set for supervised classification to predict response-to-chemotherapy, the unsupervised VAE encoding of the tumor's gene expression profile leads to better area under the receiver operating characteristic curve and area under the precision-recall curve classification performance than the original gene expression profile or the PCA principal components or the ICA components of the gene expression profile, in four out of five cancer types that we tested.

Conclusions: Given high-dimensional "omics" data, the VAE is a powerful tool for obtaining a nonlinear low-dimensional embedding; it yields features that retain biological patterns that distinguish between different types of cancer and that enable more accurate tumor transcriptome-based prediction of response to chemotherapy than would be possible using the original data or their principal components.

Keywords: Variational auto-encoder, Transcriptome, TCGA, Chemotherapy drug response classification, Cancer, Colon adenocarcinomas, Pancreatic adenocarcinoma, Bladder carcinoma, Sarcoma, Breast invasive carcinoma

Introduction

Background

Although chemotherapy is a mainstay of treatment for aggressive cancers, many agents have serious side effects [1]. Whether or not chemotherapy will provide a net benefit to a patient depends in large part on whether the malignancy responds to the treatment. Chemotherapy is often administered in cycles [2], leading to multiple opportunities where treatment appropriateness may be (re-)assessed [3]. Currently, the medical cost-benefit of chemotherapy (versus a non-pharmaceutical approach) is assessed in light of patient health status, expected therapeutic tolerance, and tumor pathological classification [4, 5]. For many cancer types, there is a broad spectrum of cases where the decision of whether or not to undergo chemotherapy is difficult [6–8]. The development of a quantitative model that could predict—based on a specific tumor's molecular profile—whether or not the tumor will respond to chemotherapy would have significant clinical utility. Moreover, an advance in machine-learning methods for the response-to-chemotherapy prediction problem [9, 10] would have potential benefits for other prediction problems in medicine.

Tumorigenesis is driven by alterations in the somatic genome and epigenome in cancer cells [11]; however, the somatic genetic or epigenetic determinants of response to chemotherapy also affect gene expression. Studies of various cancer types have demonstrated that tumor gene expression biomarkers correlate with the probability that a tumor will respond to chemotherapy, for example, a five-protein signature in breast cancer [12], a 13-gene signature in rectal cancer [13, 14], a 63-gene signature in liver cancer [15], and a support vector machine (SVM)-based model to predict survival time in breast cancer based on a 19-gene signature [16]. The findings from such "omics" studies suggest that RNA sequencing (RNA-seq)-based transcriptome measurements of tumor samples labeled for clinical response can be used to train machine-learning classifiers for predicting response to chemotherapy. However, the accuracy of models that can be learned by fully supervised approaches is limited by the small number of available clinically labeled training cases, given that tumor transcriptome data are high-variance and high-dimensional.

For typical cancers, most available tumor transcriptomes are not labeled for chemotherapeutic response; the ratio of such unlabeled to labeled tumor datasets in the Cancer Genome Atlas (TCGA; [17]) is in the range of 10–20, depending on the cancer type. Unlabeled data are a substantial resource that could—in the context of a *semi-supervised* approach—reveal multivariate patterns that could ultimately improve predictive accuracy. Semi-supervised approaches that fuse unsupervised data reduction methods for low-dimensional embedding with supervised methods (such as decision trees) for prediction have proved beneficial in problems where large unlabeled datasets are available; for example, a principal components analysis (PCA)-XGBoost method has been previously used in finance [18], and an independent component analysis (ICA)-based method has been used to classify electroencephalographic signals [19].

Previous applications of VAE in cancer

Multiple studies [20–23] have demonstrated the power of the variational autoencoder (VAE; [24, 25])—an unsupervised nonlinear data embedding model in which two deep neural networks are oppositely connected through a low-dimensional, probabilistic latent space—for finding useful features in high-dimensional data. In the context of cancer, VAEs have been variously used to (1) model gene expression and capture biological features using the TCGA Pan-cancer Project RNA-seq dataset [26, 27]; (2) find encodings that can be used to predict gene inactivation [28]; and (3) obtain an encoding for predicting chemotherapy resistance [29]. Way and Greene [28] explored VAE architectures for predicting gene inactivation in a pan-cancer dataset and reported biological insights obtained from the latent-space embeddings. George and Lio [29] used a VAE-based, unsupervised approach to encode tumor transcriptomes to obtain latent-space features associated with chemotherapy response. Dincer et al. [30] used a semi-supervised, VAE-lasso approach to predict drug sensitivity of cancer cells in vitro. In contrast to previous efforts to model cancer cell line drug sensitivity in vitro [30–33], in this work we focused on predicting therapeutic response in vivo, across five different cancer types (colon adenocarcinoma, pancreatic adenocarcinoma, bladder carcinoma, sarcoma, and breast invasive carcinoma). Specifically, we tested the hypothesis that a tumor transcriptome VAE would be useful for predicting response-to-chemotherapy in vivo, across multiple cancer types.

Research objectives

We first asked to what extent VAE-encoding tumor transcriptomes would preserve characteristics that are associated with distinct cancer types. To that end, we trained a pan-cancer transcriptome VAE and used it to encode over 11k tumor transcriptomes from 33 cancer types. By comparing two-dimensional embeddings of the original tumor transcriptomes with embeddings of the VAE-encoded transcriptomes, we found ("VAE encoding preserves cancer type features" section) that the VAE preserves the clustering of tumors of the same cancer type. Next, we selected five cancer types based on sufficiency of clinical data and trained six VAE models (three architectures and two different loss functions) to encode clinically-unlabeled transcriptomes of the five cancer types. Using TCGA clinical data, we assigned a label "responded" or "progressive" to tumors where the response to chemotherapy information was available ("Obtaining a labeled tumor transcriptome dataset" section). We then used the VAE-encoded transcriptomes for the clinically-labeled tumors as feature data for predicting response to chemotherapy using gradient boosted decision trees (XGBoost; [34]), which we found to be superior to kernel SVM. Using this "semi-supervised VAE-XGBoost" approach, we investigated ("L1 loss is better than L2 loss and cross-entropy loss for this application" section) which loss function type is best for this VAE application.

In the main part of this work, we focused ("Chemotherapy response classification results" section) on the question of whether and to what extent the semi-supervised VAE-XGBoost (our new method, Fig. 1) approach would improve performance for transcriptome-based prediction of response to chemotherapy, versus a fully-supervised approach or versus alternative semi-supervised approaches using PCA or ICA

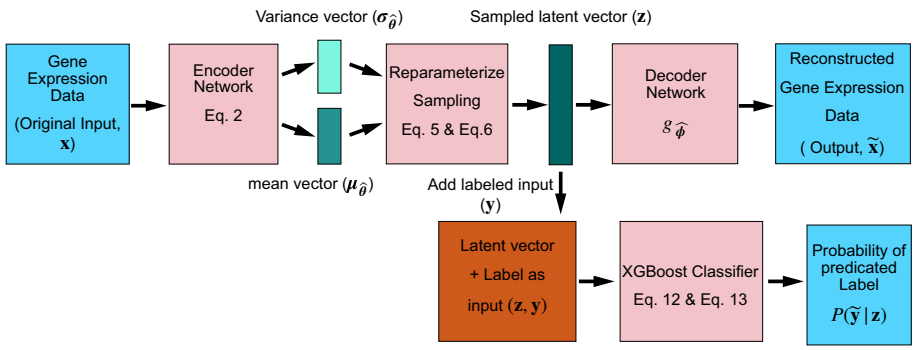

Fig. 1 Overview of the VAE-XGBoost method that we used for predicting tumor response to chemotherapy in vivo for five different cancer types. For each tumor *t*, the encoder's input vector $\boldsymbol{x}_t$ contains the levels of the top 20% of genes by intertumoral gene expression variance. Each network has multiple fully connected dense layers ("VAE model architectures" section). The encoder outputs two vectors of configurable latent variable dimension $h \ll m$: a vector of means $\boldsymbol{\mu}$ and a vector of standard deviations $\boldsymbol{\sigma}$ that parameterize the multivariate normal latent-space vector $\boldsymbol{Z}|\boldsymbol{x}_t$ ("Variational autoencoder (VAE)" section). The sampled encoding $\boldsymbol{Z}|\boldsymbol{x}_t = \boldsymbol{z}_t$ is passed to the decoding neural network (decoder), whose architecture is identical to (with inversion) that of the encoder network. The sampled latent-space vector $\boldsymbol{z}_t$ is passed to XGBoost for supervised classification to predict response to chemotherapy (training label $\boldsymbol{y}$, prediction $\tilde{\boldsymbol{y}}$)

transcriptome encodings. We further investigated the relative importance of these approaches through the lens of XGBoost feature importance ("PCA & VAE feature importance scores, for COAD" section). We carried out these analyses using a comprehensive, five-cancer set of labeled tumor transcriptomes and obtained unbiased classification performance measurements using cross-validation.

Results

VAE encoding preserves cancer type features

Given reports [35, 36] that unsupervised embeddings can be used to visualize the grouping of cancer types based on high-dimensional molecular tumor data, using unsupervised methods, we investigated the extent to which VAE encoding of tumor transcriptomes preserves data-space features that determine cancer type-specific groupings. In order to do so, we obtained RNA-seq transcriptome data from the TCGA data portal for 11,057 tumors labeled for 33 different cancer types (Figs. 2, Additional file 1: S2, S3). As a baseline visualization, we generated a two-dimensional embedding of the 11,057 tumor samples by applying *t*-distributed stochastic neighbor embedding (*t*-SNE) to the expression levels of the the top-20% highest-variance genes (threshold selected as described in "Gene expression data" section), yielding 33 clusters (Fig. 2A). Next, we trained a VAE ("Variational autoencoder (VAE), VAE model architectures" sections) with a deep architecture (`VAE-1`) to encode the expression levels of the highest-variance genes in each of 11,057 tumors into an equivalent number of points in a 50-dimensional latent space. An unsupervised *t*-SNE visualization (Fig. 2B) of the VAE-encoded tumor transcriptome data was remarkably similar in structure to the *t*-SNE visualization of the 13,584-dimensional original dataset (Additional file 1: Fig. S1). Additionally, we compared the clustering of the original transcriptome data with VAE-reconstructed transcriptome data by

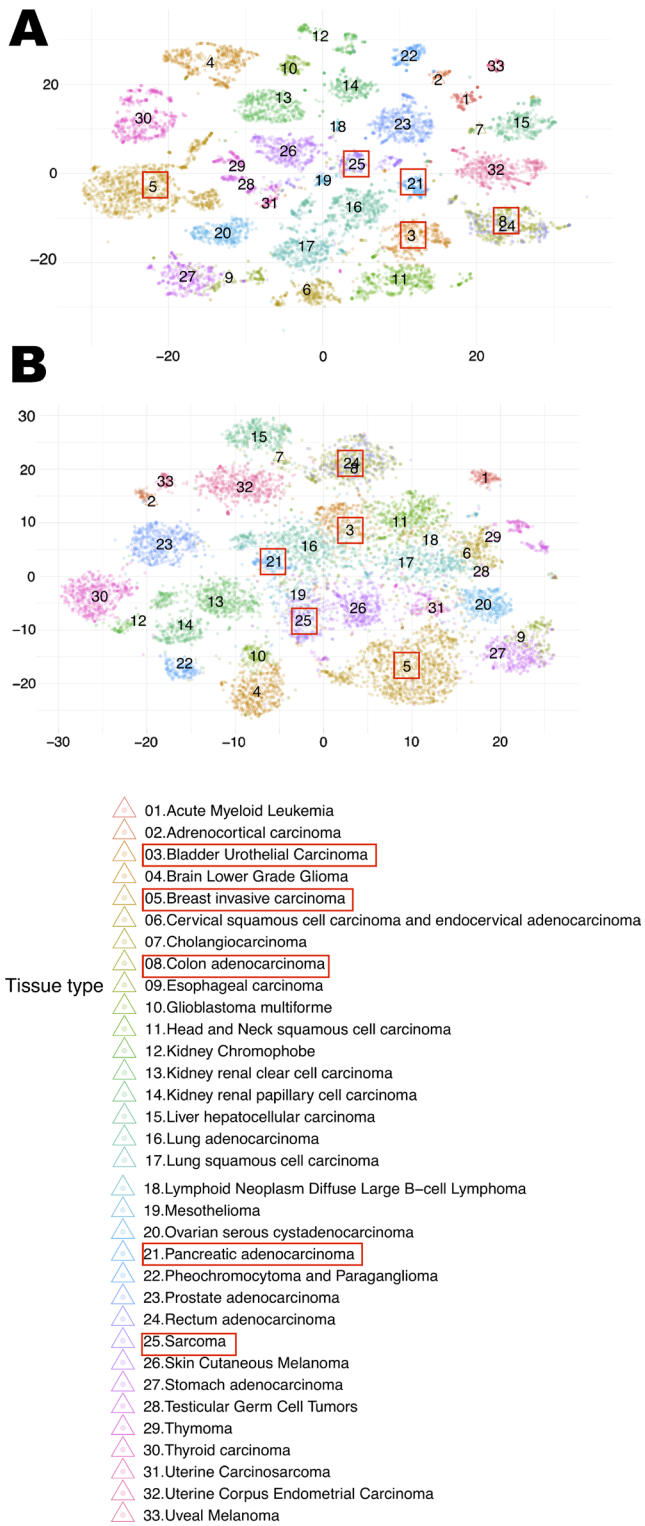

Fig. 2 Two-dimensional embedding of the 11,057 tumor transcriptomes based on *t*-SNE. Each mark represents a transcriptome, with color representing the cancer type. **A** Original gene expression data of the top-20% highest-variance genes. **B** VAE compressed gene expression data. Red rectangles denote the five cancer types selected for chemotherapy response classification

Table 1 Numbers of tumor samples that have clinical information available regarding response-to-chemotherapy, for each cancer type (n.b., the total number of labeled tumor samples exceeds the total number of patients because some patients had multiple tumors)

Cancer type	Total number of samples (labeled and unlabeled)	Number of labeled samples	Proportion of labeled samples	Class balance ratio (responding/progressive)
Breast invasive carcinoma (BRCA)	1217	394	0.324	8.61
Colon adenocarcinomas (COAD)	512	117	0.229	1.72
Bladder carcinoma (BLCA)	430	115	0.267	0.95
Pancreatic adenocarcinoma (PAAD)	182	115	0.632	0.77
Sarcoma (SARC)	265	65	0.245	0.82
Sum	2606	806		

Each cancer type's TCGA abbreviation is shown in parentheses

Uniform manifold approximation and projection (UMAP), and found similar results (Additional file 1: Figs. S2, S3). These analyses indicated that the VAE encoding preserves data-space features that distinguish individual cancer types.

Obtaining a labeled tumor transcriptome dataset

Having demonstrated that the VAE can efficiently encode tumor transcriptomes while preserving features that distinguish different cancer types, and to set the stage for implementing a semi-supervised approach for predicting response to chemotherapy, we obtained a five-cancer-type tumor transcriptome dataset with a significant subset of the tumors labeled as to whether or not the patient responded to chemotherapy, as described below. We obtained transcriptomes of 2,606 tumors across five cancer types [colon adenocarcinoma (COAD), pancreatic adenocarcinoma (PAAD), bladder carcinoma (BLCA), sarcoma (SARC), and breast invasive carcinoma (BRCA); Table 1]. We selected the five cancer types based on availability of a sufficient amount of labeled data in TCGA and for 806 of the tumor transcriptomes, we generated binary labels corresponding to "responded" or "progressive".

The ratio of responding tumors to progressive disease tumors (i.e., the class balance ratio) ranged from a low of 0.77 for pancreatic cancer to a high of 8.61 for breast cancer.

L1 loss is better than L2 loss and cross-entropy loss for this application

Having obtained 2,606 transcriptomes of tumors of five cancer types (with 806 of the tumors labeled by response), we next sought to determine which type of VAE reconstruction loss function—L1, L2, or binary cross entropy—would yield transcriptome encodings that are most amenable to accurate XGBoost-based prediction chemotherapy response. On the 2,606 tumor transcriptomes, we trained three sets of cancer type-specific VAEs ("VAE model architectures" section) using L1 loss, L2 loss, and binary cross-entropy loss respectively. We used the L1, L2, and binary cross-entropy VAEs to encode the 806 labeled tumor transcriptomes (the top 20% most variable genes in each cancer type, merged across the five cancers, for a total of 13,584 genes) spanning the five cancer types, yielding (for each cancer type) three feature matrices: one based on L1 loss,

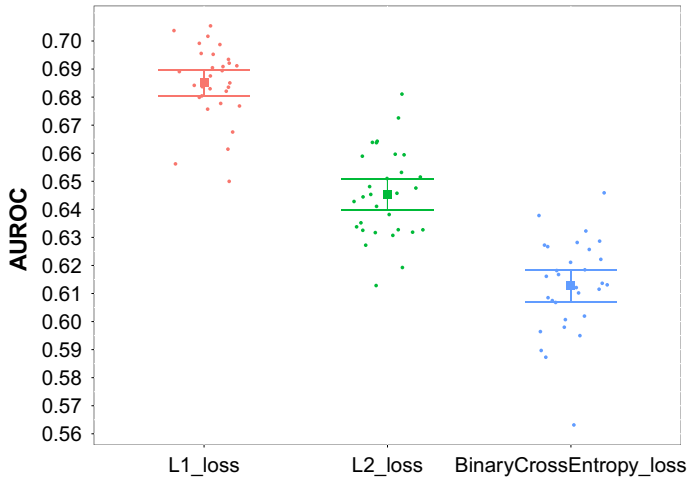

Fig. 3 Predicting chemotherapy response using VAE-XGBoost models with different types of reconstruction loss for the VAE training. Marks represent AUROC values averaged over five different types of cancer, grouped by VAE loss function type. Squares, within-group sample mean; bars, 95% confidence intervals (c.i.)

one based on L2 loss, and a third one based on binary cross-entropy loss. We separately evaluated the three feature matrices for XGBoost prediction of the binary response-to-chemotherapy class label. By test-set area under the receiver operating characteristic (AUROC), averaged across the five cancers, we found (Fig. 3) that the features that were generated by the L1 VAEs led to 6.2% better ($p < 10^{-9}$, Welch's t-test) classification performance than the features generated by the L2 VAEs, 11.7% better ($p < 10^{-9}$, Welch's t-test) classification performance than the features generated by the binary cross-entropy VAEs and thus, for all subsequent analyses, we used VAEs trained with L1 loss.

Chemotherapy response classification results

Having selected L1 reconstruction loss for training VAEs to encode tumor transcriptomes for predicting response-to-chemotherapy, we developed a semi-supervised approach based on VAE encoding of the tumor transcriptome, for predicting chemotherapy response. In brief, our approach consisted of three steps:

1. Training a VAE to encode clinically *unlabeled* tumor transcriptomes (for the top 20% most variable genes) for a single cancer type, into a low-dimensional space ("VAE model architectures" section).
2. Using that VAE to obtain latent-space encodings for the tumor transcriptomes that are labeled for a relevant clinical endpoint (in this work, response to chemotherapy).
3. Training and testing a supervised classifier for predicting chemotherapy response.

Because some cancer types benefited from a deeper VAE network architecture than others for effective encoding, we used three different VAE architectures for learning features for predicting chemotherapy response in the context of three subsets of cancer types (`VAE-1` for breast and pancreatic; `VAE-2` for colon; and `VAE-3` for bladder and sarcoma; Table 5). For each VAE architecture, our approach was to use all of the data

Table 2 Comparison of chemotherapy response prediction performance for XGBoost models trained with VAE-derived features versus autoencoder (AE)-derived features, for three cancer types (BRCA, BLCA, and PAAD)

Cancer type	Mean				*p* (Welch's *t*-test)	
	AUROC		AUPRC		AUROC	AUPRC
	VAE	AE	VAE	AE	VAE versus AE	VAE versus AE
BRCA	**0.674**	0.575	**0.192**	0.137	1.61×10^{-15}	5.38×10^{-10}
PAAD	**0.738**	0.660	**0.764**	0.695	3.46×10^{-10}	6.72×10^{-7}
BLCA	**0.659**	0.573	**0.649**	0.577	7.97×10^{-12}	1.23×10^{-7}
SARC	**0.704**	0.611	**0.736**	0.654	2.78×10^{-7}	1.75×10^{-6}

The *p* values are for row-wise difference of means tests for the two columns under "AUROC" and for the two columns under "AUPRC", respectively. For each cancer type (row), the highest mean AUROC performance is shown in boldface

from the five-cancer set of 2,606 unlabeled tumors for VAE training, but for *predicting* chemotherapy response for a given cancer type, we used encodings from the VAE architecture that corresponds to the cancer type (Table 5).

To select the supervised classification algorithm for step (3) above, we used an empirical approach, comparing the AUROC performance of XGBoost, kernel SVM, and *k*-nearest neighbors for predicting sarcoma response to chemotherapy with features based on VAE-3 encodings (semi-supervised) or expression levels of individual genes (fully-supervised). We found (Additional file 1: Fig. S4) XGBoost to be superior to kernel SVM and *k*-nearest neighbors (KNN), in both semi-supervised and fully supervised analyses, and thus we chose XGBoost as the classification algorithm for subsequent analyses.

To address the primary question of to what extent a VAE-based, semi-supervised (VAE-XGBoost) approach could advance the state-of-the-art for transcriptome-based prediction of chemotherapy response, we sought to compare VAE-XGBoost's performance to that of three alternative approaches: (1) a semi-supervised approach using a regular auto-encoder (AE) with the same architecture as the VAE; (2) a fully supervised approach directly using the transcriptome data; and (3) a semi-supervised approach based on a traditional dimensional reduction technique (either principal component analysis, PCA; or independent component analysis, ICA).

VAE-XGBoost versus AE-XGBoost

To address alternative approach (1) ("traditional auto-encoder"), we compared the performance of VAE-XGBoost to that of a model consisting of a regular auto-encoder combined with XGBoost ("AE-XGBoost"; Table 2 and Additional file 1: Figs. S5–S6). For this four-cancer analysis, we used the `VAE-1` architecture for BRCA and PAAD, which was the same network that we used in the *t*-SNE analysis and the `VAE-3` architecture for BLCA and SARC (Table 5). We measured performance using test-set AUROC and AUPRC using five-fold cross-validation. We found (Table 2) that VAE-XGBoost outperformed AE-XGBoost by an average AUROC increase of 14.5% and an average AUPRC increase of 16.3% over the four-cancer average (breast, pancreatic, bladder, and

Table 3 Comparison of chemotherapy response prediction performance (AUROC) for XGBoost models trained with original transcriptome data ("Raw data") or transcriptome data encoded with PCA, ICA, or VAE. This analysis was carried out across five cancers (BRCA, COAD, BLCA, PAAD, and SARC)

Cancer type	AUROC (mean)				*p* (Welch's *t*-test)		
	VAE	Raw data	PCA	ICA	VAE versus Raw data	VAE versus PCA	VAE versus ICA
BRCA	**0.674**	0.649	0.614	0.609	8.07×10^{-4}	3.80×10^{-12}	3.39×10^{-14}
PAAD	**0.738**	0.694	0.710	0.685	6.99×10^{-6}	5.04×10^{-3}	2.41×10^{-6}
COAD	0.707	0.674	**0.726**	0.689	1.19×10^{-3}	2.81×10^{-2}	5.65×10^{-2}
BLCA	**0.659**	0.626	0.593	0.650	7.81×10^{-5}	4.27×10^{-9}	2.61×10^{-1}
SARC	**0.704**	0.679	0.682	0.701	3.49×10^{-2}	2.91×10^{-2}	8.64×10^{-1}

The *p* values are for row-wise difference of means tests for the indicated pairs of sample groups (columns). For each cancer type (row), the highest mean AUROC performance is shown in boldface

Table 4 Comparison of chemotherapy response prediction performance (AUPRC) for XGBoost models trained with original transcriptome data ("Raw data") or transcriptome data encoded with PCA, ICA, or VAE. This analysis was carried out across five cancers (BRCA, COAD, BLCA, PAAD, and SARC)

	AUPRC (mean)				*p* (Welch's *t*-test)		
Cancer type	VAE	Raw data	PCA	ICA	VAE versus Raw data	VAE versus PCA	VAE versus ICA
BRCA	**0.192**	0.157	0.145	0.150	4.21×10^{-6}	7.42×10^{-10}	5.01×10^{-8}
PAAD	**0.764**	0.729	0.746	0.713	3.38×10^{-4}	9.12×10^{-2}	2.02×10^{-6}
COAD	**0.593**	0.535	0.579	0.545	6.52×10^{-4}	3.91×10^{-1}	1.27×10^{-3}
BLCA	0.649	0.623	0.587	**0.654**	4.30×10^{-2}	1.60×10^{-7}	6.13×10^{-1}
SARC	**0.736**	0.713	0.714	0.729	6.15×10^{-2}	1.61×10^{-1}	5.96×10^{-1}

The *p* values are for row-wise difference of means tests for the indicated pairs of sample groups (columns). For each cancer type (row), the highest mean AUPRC performance is shown in boldface

sarcoma), with ($p < 10^{-9}$, Welch's *t*-test) classification performance. Thus, we used the VAE for neural network-based unsupervised embeddings, for subsequent analyses.

VAE-XGBoost versus fully-supervised XGBoost

To address alternative approach (2) ("fully supervised"), we empirically compared the performance of the VAE-XGBoost method to a fully supervised model in which we applied XGBoost directly to the tumor expression levels of the top 20% most variable genes (13,584 genes) as feature data. For five out of five cancer types (breast, colon, pancreatic, bladder, and sarcoma), in terms of test-set AUROC, the VAE-XGBoost approach outperformed the fully-supervised XGBoost approach (Additional file 1: Fig. S7), by Welch's *t*-test (Table 3). In terms of test-set AUPRC, for four out of five cancer types (breast, colon, pancreatic, and bladder), the VAE-XGBoost approach outperformed the fully-supervised approach of applying XGBoost directly to the expression levels of the tumors' top 20% most variable genes (Additional file 1: Fig. S8), by Welch's *t*-test (Table 4); for SARC, the semi-supervised VAE-XGBoost and fully-supervised models' performances were statistically indistinguishable.

VAE-XGBoost versus PCA-XGBoost and ICA-XGBoost

To address alternative approach (3), we empirically compared VAE-XGBoost to models in which PCA or ICA components were used as XGBoost features (i.e., "PCA-XGBoost" and "ICA-XGBoost"). We aimed to empirically study prediction performance of these models for each of the five cancer types separately, using the set of cancer type-specific labeled tumors (806 labeled tumors in all). For four out of five cancer types (bladder, breast, pancreatic, and sarcoma), in terms of AUROC the semi-supervised VAE-XGBoost method significantly outperformed the semi-supervised PCA-XGBoost method (Table 3 and Additional file 1: Fig. S7). Additionally, for three out of five cancer types (breast, colon, and pancreatic), the semi-supervised VAE-XGBoost method significantly outperformed the semi-supervised ICA-XGBoost method (Table 3 and Additional file 1: Fig. S7). The five-cancer average AUROC for VAE-XGBoost was 0.688, a performance gain of 6.3% over the five-cancer average AUROC for PCA-XGBoost (0.646), a gain of 6.5% over the ICA-XGBoost (0.645) and a gain of 4.5% over the fully-supervised model's average (0.658). Notably, a single deep VAE architecture (`VAE-1`, which had a 50-dimensional latent space and six layers in the encoder) yielded latent-space encodings that outperformed semi-supervised PCA-XGBoost for two cancer types (breast and pancreatic); a single shallow VAE architecture (`VAE-3`, which had a 500-dimensional latent space and two layers in the encoder) yielded latent-space encodings that outperformed semi-supervised PCA-XGBoost for two cancer types (bladder and sarcoma).

For three out of five cancer types (breast, bladder, and pancreatic), in terms of AUPRC the semi-supervised VAE-XGBoost method significantly outperformed the semi-supervised PCA-XGBoost method (Additional file 1: Fig. S8 and Table 4). Additionally, for three out of five cancer types (breast, colon, and pancreatic), the semi-supervised VAE-XGBoost method significantly outperformed the semi-supervised ICA-XGBoost method (Additional file 1: Fig. S8 and Table 4). The five-cancer average AUPRC for VAE-XGBoost was 0.441, a performance gain of 9.1% over the five-cancer average AUPRC for PCA-XGBoost (0.403), a gain of 8.2% over the ICA-XGBoost (0.406), and a gain of 8.5% over the fully-supervised model's average (0.405).

PCA & VAE feature importance scores, for COAD

Having established that the semi-supervised VAE-XGBoost outperforms the semi-supervised PCA-XGBoost approach for tumor transcriptome-based prediction of chemotherapy response for four out of five cancer types, we sought to understand the basis for the higher performance of PCA-XGBoost over VAE-XGBoost on the fifth cancer type, colon adenocarcinoma (COAD). Specifically, we investigated whether the strong performance of PCA-XGBoost on COAD is attributable to differences in the distributions of XGBoost feature importance scores of the PCA features versus VAE latent-space features. We found that the distribution of feature importance scores (as a function of rank) was more sharply peaked at lowest-ranked features in the VAE than in the PCA (Fig. 4), suggesting that the performance gain with PCA reflects a broader spectrum of informative features for that particular cancer type.

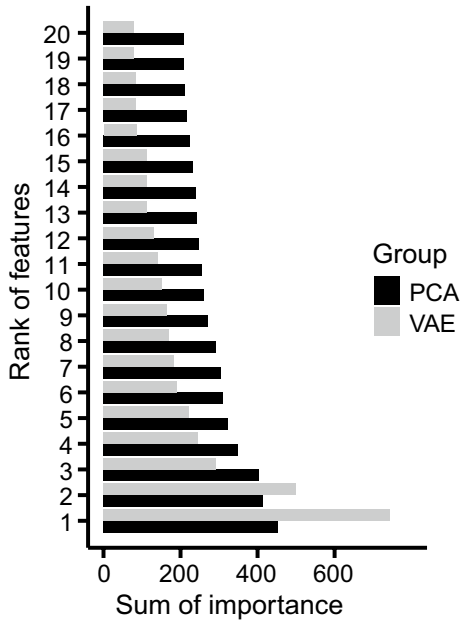

Fig. 4 Comparing the distributions of feature importance scores for the most informative features for PCA-XGBoost and VAE-XGBoost, for predicting response to chemotherapy in colon cancer (COAD). Bars indicate the sum (over 30 replications) of XGBoost feature importance scores. "Group" indicates the low-dimensional embedding method used (VAE or PCA). Bars separately ordered from highest to lowest (only top 20 most important features shown)

Table 5 VAE architectures used for predicting chemotherapy response (*h*, latent space dimension; "layers", # of layers used in the encoder/decoder)

Name	Cancer types	*h*	Layers
`VAE-1`	BRCA, PAAD	50	Six
`VAE-2`	COAD	400	Two
`VAE-3`	BLCA, SARC	500	Two

Discussion

As far as we are aware, this work is the first report of a multi-cancer investigation of the potential for a VAE-based, semi-supervised approach for predicting in vivo chemotherapy response from the tumor transcriptome. Across the five cancer types that we studied, the ratio of responding tumors to progressive disease tumors ranged from a low of 0.77 for pancreatic cancer to a high of 8.61 for breast cancer, reflecting a broad range of resistances to standard-of-care chemotherapy. Our results clearly demonstrate the utility of the VAE for compressing high-dimensional data to a continuous, low-dimensional latent space while retaining features that are essential for distinguishing different cancer types and for predicting response to chemotherapy. Nevertheless, three limitations of this work bear noting.

The first limitation concerns the type(s) of tumor "omics" data from which features are derived for the predictive model. While in this work we focused on tumor

transcriptome data which can be measured with high precision over a wide dynamic range of transcript abundances by RNA-seq, we note that TCGA datasets of tumor somatic mutations and copy number alteration events are also available [17]. Given the voluminous literature on the use of tumor somatic genomic data for precision cancer diagnosis [37–39], tumor DNA datasets are fertile ground for developing a semi-supervised, multi-omics model for predicting response to chemotherapy.

Second, for decision tree-based response-to-chemotherapy prediction, the performance of VAE-encoded transcriptome features is somewhat sensitive to the type of normalization used for the gene expression levels (data not shown). We explored various published normalization methods for the RNA-seq data including standardization of log counts and using FPKM; we ultimately chose min-max-normalized $\log_2$ total-count-normalized counts for the gene expression levels to be used to derive features. However, there are additional transcript quantification methods [40] that could be explored in the context of finding optimal tumor transcriptome VAE encodings for precision oncology. A similar comment applies to the specific form of the reconstruction loss function: in our analysis, features from the VAE trained with L1 loss clearly (across five cancers) outperformed those from the VAE trained with L2 or cross-entropy loss, and thus, consistent with Way and Greene [28], we used L1 loss for the VAE that we used to address the main question of this work as well as the pan-cancer *t*-SNE analysis.

The third limitation relates to the VAE architecture. While it is promising that a single deep VAE architecture (`VAE-1`, with a 50-dimensional latent space and six fully-connected layers) yielded features that outperformed PCA and the original RNA-seq feature data for two different cancer types (breast and pancreatic), for the other three cancer types, it was necessary to use shallower (two-layer) VAE architectures with bigger latent space dimensions (400 and 500, respectively). Of the five cancer types that we studied, colon cancer and sarcoma had the lowest proportions of labeled samples (0.229 and 0.245 respectively; see Table 1). Our findings suggest that for some cancers, a deep, low-latent-dimension VAE architecture yields optimal features for predicting response, while for other cancers, a shallow, medium-sized-latent-dimension VAE architecture is more effective. Hu and Greene [41], based on a study employing single-cell transcriptome profiling, noted substantial performance differences with hyperparameter tuning on VAE architectures; they further noted that in terms of the robustness of performance with respect to hyperparameter variation, a base VAE with two layers was better than a deeper VAE architecture. Lakhmiri et al. [42] reported VAE architecture hyperparameter tuning as well as the training phase have a great impact on the overall precision of the network and its ability to generalize, and proposed Δ-MADS, a hybrid derivative-free optimization algorithm for VAE fitting. More study with larger datasets will be required in order to determine whether a single VAE architecture could be successfully used for general-purpose tumor transcriptome feature extraction for precision oncology.

While our results show promise for the VAE in the context of a semi-supervised approach for response-to-chemotherapy prediction, for colon cancer, the VAE-XGBoost method did not outperform PCA-XGBoost (though it did outperform the fully supervised approach of XGBoost trained on the unencoded gene expression data). One possible explanation for the colon cancer-specific superior performance of PCA features over VAE features for predicting response to chemotherapy may be due to the fact that

while (for COAD) feature importance for the VAE features is sharply peaked for the first few features and falls off fairly rapidly with feature rank, the PCA features have a significantly flatter distribution of relative feature importance (Fig. 4). Follow-on studies with larger datasets will be required to delineate under what circumstances transcriptome VAE encodings will prove superior to linear principal components. Multiple groups have argued [43–45] that to improve current precision oncology models, significantly expanded training datasets are needed to overcome the challenges posed by tumor heterogeneity, and that models must more broadly leverage somatic genetic and epigenetic information. We anticipate that the performance of VAE-XGBoost could improve significantly with more unlabeled and labeled tumor transcriptome data. Finally, we note a possible future extension of this work that will become feasible when larger training datasets are available: because response to chemotherapy is drug-dependent, the XGBoost classifier can easily include and use the chemotherapy drug type used for the patient (Additional file 1: Table S1) as a categorical feature.

Conclusions

For four of the five cancer types that we studied, the semi-supervised VAE-XGBoost approach significantly outperformed a semi-supervised PCA-XGBoost approach for tumor transcriptome-based prediction of response to chemotherapy, reaching a top AUROC of 0.738 for pancreatic adenocarcinoma. For three of the five cancer types that we studied, the semi-supervised VAE-XGBoost approach significantly outperformed a semi-supervised ICA-XGBoost approach for tumor transcriptome-based prediction of response to chemotherapy. For BLCA and SARC, the semi-supervised VAE-XGBoost and ICA-XGBoost models' performances were statistically indistinguishable. For five out of five cancer types, the semi-supervised VAE-XGBoost approach significantly outperformed a fully-supervised approach consisting of XGBoost applied to the expression levels of the top 20% most variably expressed genes. Given high-dimensional "omics" data, the VAE is a powerful tool for obtaining a nonlinear low-dimensional embedding; it yields features that retain biological patterns that distinguish between different types of cancer and that enable more accurate tumor transcriptome-based prediction of response to chemotherapy than would be possible using the original data or their principal components.

Methods

We carried out all data processing and machine-learning tasks on a Dell XPS 8700 workstation equipped with Nvidia Titan RTX GPU and running the Ubuntu GNU/Linux operating system version 16.04. All of the analysis code that we implemented was executed in Python version 3.5.5 except that we used R version 3.3.3 for statistical analysis of AUROC and AUPRC values ("Area Under ROC Curve (AUROC), Area Under the precision-recall Curve (AUPRC)" sections), gene-level MAD calculations ("Gene expression data" section) and plotting ("Lower-dimensional embedding" section). We carried out all statistical tests using the R computing environment (version 3.3.3) and using the R software package `stats` version 3.4.4.

Gene expression data

From the Xena data portal [46], we obtained TCGA Level 3 tumor RNA-seq transcriptome data of 33 cancer types (totaling 11, 057 tumors) and, for the response-to-chemotherapy prediction problem, five cancer types [colon adenocarcinomas (COAD), pancreatic adenocarcinoma (PAAD), bladder carcinoma (BLCA), sarcoma (SARC), and breast invasive carcinoma (BRCA)] totaling 2,606 tumors. We selected the five cancer types based on two criteria: (1) a sufficient number (at least 65) of paired tumor-transcriptome and clinical data samples available for the cancer type; and (2) a sufficient number (at least 180) of tumor transcriptome samples available (regardless of the clinical data availability) for the cancer type. We obtained both the RNA-seq (gene-level) total-read-count-normalized $\log_2(1+C)$ read counts and normalized (fragments per kilobase of transcript per million mapped reads, FPKM [47]) expression data for for 60,483 human genes. To focus the machine-learning on the portion of the tumor transcriptome that had the most variation from tumor to tumor, we identified the top 20% most variable genes as measured by the median absolute deviation (MAD) across tumors, of gene expression in terms of FPKM (we used FPKM for this purpose in order to mitigate bias due to read length and tumor-specific depth of sequencing) based on our preliminary results for prediction of response-to-chemotherapy for SARC, for different quantile thresholds of genes by variability of expression (Additional file 1: Fig. S4). For deriving feature-sets for XGBoost prediction directly based on transcript abundances or based on VAE- or PCA encoding, the 20% criterion applied to each of the five cancer types yielded a set of 13,584 genes. We computed MAD using the R package `stats` version 3.4.4 [48] with default parameters. After the variance-filtering step, we used the $\log_2(1+C)$ of total-count-normalized count values for the top-20% highest-variance genes (that were selected as described above) to obtain (or encode) feature values. We compared the performance—in terms of minimizing the VAE reconstruction loss (see "Variational autoencoder (VAE)" section)—of different feature scaling methods (no scaling, min-max normalization, and standardization [49]) and selected min-max normalization as the method that we used to rescale gene-level count data for input into the VAE.

Lower-dimensional embedding

We computed *t*-SNE embedding components of the tumors using the function `sklearn.manifold.TSNE` from the python software package `scikit-learn` version 0.19.1 with parameters `init` = "`pca`", `perplexity` = `20`, `learning_rate` = `300`, and `n_iter` = `400`. We computed UMAP embedding components using the function `sklearn.manifold.umap.UMAP` from the python software package `scikit-learn` version 0.19.1 with parameters `n_neighbors` = `50`, `min_dist` = `0.3`, and `metric` = "`euclidean`". For plotting the embeddings, we used the R software package `ggplot2` version 3.1.1.

Variational autoencoder (VAE)

An autoencoder is a type of model that combines "encoder" and "decoder" neural networks to learn a low-dimensional continuous data encoding from which the input signal can be approximately reconstructed [50]. A key advantage of an autoencoder is that it is

unsupervised, i.e., it can be trained without labeled examples. Unlike classical autoencoders (e.g., sparse or denoising autoencoders), the variational autoencoder (VAE) is a generative probabilistic model which maps an input vector to a latent-space *random variable* (r.v.). Below, we mathematically define the VAE.

Let $\mathbb{T}$ denote the set of tumors for which the VAE is to be fit to the tumor transcriptomes (with $n \equiv |\mathbb{T}|$) and let m denote the number of genes for which transcript abundances are used to represent the tumor transcriptome. After min-max transformation of the tumor transcriptome measurements ("Gene expression data" section), each tumor's transcriptome is represented as a vector $\boldsymbol{x} \in [0,1]^m$. Let $\boldsymbol{X}$ denote the random variable representing the population distribution from which tumor transcriptomes are sampled, and let $\mathbf{X} \in [0,1]^{m \times n}$ represent the composite matrix of all sampled tumor transcriptomes). We aim to learn a VAE that will comprise an encoder and decoder, with the encoder consisting of mean and variance functions $\boldsymbol{\mu} : [0,1]^m \rightarrow \mathbb{R}^h$ and $\boldsymbol{\sigma} : [0,1]^m \rightarrow \mathbb{R}^h_+$, respectively. Together, $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ map the tumor transcriptome vector $\boldsymbol{x}_t$ to a h-dimensional r.v. $\boldsymbol{Z}|\boldsymbol{x}_t$,

$$\boldsymbol{Z}|\boldsymbol{x}_t \sim \mathcal{N}(\boldsymbol{\mu}(\boldsymbol{x}_t), \mathrm{diag}(\boldsymbol{\sigma}(\boldsymbol{x}_t))), \tag{1}$$

where $\mathrm{diag}(\boldsymbol{m})$ is a matrix whose diagonal elements are the elements of the vector $\boldsymbol{m}$. This equation is the same as Eq. 1 in Zemouri et al. [51]. The decoder is a function $\boldsymbol{g} : \mathbb{R}^h \rightarrow [0,1]^m$ that, for an outcome $\boldsymbol{Z}|\boldsymbol{x}_t = \boldsymbol{z}_t \in \mathbb{R}^h$, maps

$$\boldsymbol{g} : \boldsymbol{z}_t \mapsto \boldsymbol{g}(\boldsymbol{z}_t) \equiv \widetilde{\boldsymbol{x}}_t; \tag{2}$$

the tilde on $\widetilde{\boldsymbol{x}}_t$ denotes that it is the decoded data for the tumor transcriptome $\boldsymbol{x}_t$. A good autoencoder should have low reconstruction error L, which is convenient to define in terms of the p-norm of the difference between the tumor transcriptome data $\boldsymbol{x}_t$ and the reconstructed data $\widetilde{\boldsymbol{x}}_t$, i.e., $||\boldsymbol{x}_t - \widetilde{\boldsymbol{x}}_t||_p^p$, where $||\ \ ||_p$ denotes the p-norm. However, this definition of the reconstruction error is only deterministic in the context of a specific outcome $\boldsymbol{Z}|\boldsymbol{x}_t = \boldsymbol{z}_t$. Thus, it is conventional to define the reconstruction error as an *expectation value* over outcomes of $\boldsymbol{Z}|\boldsymbol{x}_t$,

$$L|(\boldsymbol{X} = \boldsymbol{x}_t) \equiv \mathbb{E}_{\boldsymbol{Z}|\boldsymbol{x}_t = \boldsymbol{z}_t}(||\boldsymbol{x}_t - \boldsymbol{g}(\boldsymbol{z}_t)||_p^p), \tag{3}$$

where $\mathbb{E}_\Omega$ represents an expectation value over a space of outcomes Ω. It should be noted the above representation of the reconstruction error is in terms of the outcome, $\boldsymbol{z}_t$, of a r.v. $(\boldsymbol{Z}|\boldsymbol{x}_t)$ whose distributional parameter functions $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ have hyperparameters (neural network coefficients) that will be fitted. This equation is similar to Eq. 3 in Zemouri et al. [51]. Compared to the binary cross-entropy loss used in Eq. 3 in Zemouri et al. [51], our Eq. 3 used L1 loss instead (see findings from an empirical study in "L1 loss is better than L2 loss and cross-entropy loss for this application" section demonstrating the superiority of L1 over L2 or binary cross-entropy for the VAE reconstruction loss function). Because Eq. 3 is ill-suited to backpropagation, it is helpful to recast it in terms of a new random variable $\boldsymbol{\mathcal{E}}_t$ that depends on $\boldsymbol{Z}|\boldsymbol{x}_t$ by

$$\boldsymbol{\mathcal{E}}_t \equiv (\mathrm{diag}(\boldsymbol{\sigma}(\boldsymbol{x}_\mathrm{t})))^{-\frac{1}{2}}(\boldsymbol{Z}_\mathrm{t}|\boldsymbol{x}_\mathrm{t} - \boldsymbol{\mu}(\boldsymbol{x}_\mathrm{t})). \tag{4}$$

It follows from Eqs. 4 and 1 that $\boldsymbol{\mathcal{E}}_t$ is standard multivariate normal,

$$\boldsymbol{\mathcal{E}}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{5}$$

where $\mathbf{I}$ is the $h \times h$ identity matrix, and thus, $\boldsymbol{\mathcal{E}}_t$ does not depend on $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$, or t. We therefore drop the subscript t and simply denote the rescaled latent-space random variable as $\boldsymbol{\mathcal{E}}$. Solving Eq. 4 for $\boldsymbol{Z}|\boldsymbol{x}_t$ and applying it to Eq. 3, the reconstruction error $L|(\boldsymbol{X}=\boldsymbol{x}_t)$ can be represented by

$$L|(\boldsymbol{X}=\boldsymbol{x}_t) \;=\; \mathbb{E}_{\boldsymbol{\mathcal{E}}}\left(\left|\left|\boldsymbol{x}_t - \boldsymbol{g}\left(\boldsymbol{\mu}(\boldsymbol{x}_t) + \sqrt{\text{diag}(\boldsymbol{\sigma}(\boldsymbol{x}_t))}\,\boldsymbol{\mathcal{E}}\right)\right|\right|_p^p\right), \tag{6}$$

which is amenable to backpropagation because the only r.v. in it is $\boldsymbol{\mathcal{E}}$, whose distributional parameters do not depend on the neural network coefficients that we will be varying. In practice, rather than computing the multivariate integral over outcomes of $\boldsymbol{\mathcal{E}}$, $L|(\boldsymbol{X}=\boldsymbol{x}_t)$ is typically approximated by averaging over a limited number J of samples from $\boldsymbol{\mathcal{E}}$,

$$L|(\boldsymbol{X}=\boldsymbol{x}_t) \simeq \left\langle\left(\left|\left|\boldsymbol{x}_t - \boldsymbol{g}\left(\boldsymbol{\mu}(\boldsymbol{x}_t) + \sqrt{\text{diag}(\boldsymbol{\sigma}(\boldsymbol{x}_t))}\,\boldsymbol{\epsilon}_j\right)\right|\right|_p^p\right)\right\rangle_j, \tag{7}$$

where $\langle\rangle_j$ denotes average over $j \in \{1, \ldots, J\}$ and $\boldsymbol{\epsilon}_j$ is sample j from $\boldsymbol{\mathcal{E}}$. Following Way and Greene [28], we used a number of samples that is equivalent to the dimension of the transcriptome, i.e., $J = m$. For the case of $p = 2$ (i.e., L2 norm), minimizing $L|(\boldsymbol{X}=\boldsymbol{x}_t)$ as defined above is equivalent to maximizing the expectation value of the log-likelihood $\log(P(\boldsymbol{g}(\boldsymbol{Z}) = \boldsymbol{x}_t \mid \boldsymbol{X} = \boldsymbol{x}_t))$. However, following Way and Greene [28] and consistent with empirical evidence ("L1 loss is better than L2 loss and cross-entropy loss for this application" section), for our five-cancer study of the utility of a VAE-based approach for response-to-chemotherapy prediction, as well as for the pan-cancer t-SNE analysis ("VAE encoding preserves cancer type features" section), we chose to use L1 reconstruction loss, i.e., $p = 1$ in Eq. 3.

The reconstruction loss measures bias error, whose minimization must be balanced against the simultaneous goal of controlling variance error through regularization. In the VAE, regularization requires incentivizing (in the learning of $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$, and $\boldsymbol{g}$) the latent space distributions of $\boldsymbol{Z}|\boldsymbol{x}$ to be close to standard multivariate normal. This is accomplished by assigning a penalty based on the Kullback-Leibler divergence between the distribution of $\boldsymbol{Z}|\boldsymbol{x}_t$ and the target distribution $\boldsymbol{\mathcal{E}}$, represented by $D_{\text{KL}}(P(\boldsymbol{Z}|\boldsymbol{x}_t)\,||\,P(\boldsymbol{\mathcal{E}}))$. This regularization is analytically tractable [52], and for a given tumor t yields (Supplementary Equation, Eq. S2) the following regularization function:

$$D_{\text{KL}}\left(P(\boldsymbol{Z}_t|\boldsymbol{x}_t)\;\middle|\middle|\;P(\boldsymbol{\mathcal{E}})\right) = ||\boldsymbol{\mu}(\boldsymbol{x}_t)||_2^2 + ||\boldsymbol{\sigma}(\boldsymbol{x}_t)||_2^2 - ||\log(\boldsymbol{\sigma}(\boldsymbol{x}_t))||_1 - 1, \tag{8}$$

where $\log(\boldsymbol{\sigma}_t)$ denotes an element-wise log and $||\;\;||_1$ is the L1 norm.

Fitting the VAE to $\mathbf{X}$ requires selecting $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$, and $\boldsymbol{g}$ from their respective function spaces; in practice, we search over functions that can be represented using a neural network for $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ (parameterized by the vector $\boldsymbol{\theta}$)[1] and a neural network for the function

[1] Note, functions $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ are just two different outputs of the encoding neural network, differing only at the final layer, and thus for simplicity of notation we represent them as having a common parameter vector $\boldsymbol{\theta}$.

$\boldsymbol{g}$ (parameterized by the vector $\boldsymbol{\phi}$). Exploring the space of functions $\boldsymbol{\mu}_{\boldsymbol{\theta}}$, $\boldsymbol{\sigma}_{\boldsymbol{\theta}}$, and $\boldsymbol{g}_{\boldsymbol{\phi}}$ corresponds to computationally searching for the vector pair $(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\phi}})$ that together minimize the joint (over all tumors) sum of the tumor-specific reconstruction loss and the regularization penalty,

$$(\widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\phi}}) = \underset{(\boldsymbol{\theta},\boldsymbol{\phi})}{\operatorname{argmin}} \sum_{t\in\mathbb{T}} \left[L|(\boldsymbol{X} = \boldsymbol{x}_t) + D_{\mathrm{KL}}\left(P(\boldsymbol{Z}|\boldsymbol{x}_t) \,\middle|\middle|\, P(\boldsymbol{\mathcal{E}})\right)\right]. \tag{9}$$

Applying Eqs. 6, 7, and 8, and setting $p = 1$ as discussed above, we obtain the explicit formula for fitting a VAE to $\mathbf{X}$,

$$\begin{aligned}(\widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\phi}}) = \underset{(\boldsymbol{\theta},\boldsymbol{\phi})}{\operatorname{argmin}} \sum_{t\in\mathbb{T}} \Bigg[& \frac{1}{J}\sum_{j=1}^{J}\left(\left\|\boldsymbol{x}_t - \boldsymbol{g}_{\boldsymbol{\phi}}\left(\boldsymbol{\mu}_{\boldsymbol{\theta}}(\boldsymbol{x}_t) + \sqrt{\operatorname{diag}(\boldsymbol{\sigma}_{\boldsymbol{\theta}}(\boldsymbol{x}_t))}\,\boldsymbol{\epsilon}_j\right)\right\|_1\right) \\ & + ||\boldsymbol{\mu}_{\boldsymbol{\theta}}(\boldsymbol{x}_t)||_2^2 + ||\boldsymbol{\sigma}_{\boldsymbol{\theta}}(\boldsymbol{x}_t)||_2^2 - ||\log(\boldsymbol{\sigma}_{\boldsymbol{\theta}}(\boldsymbol{x}_t))||_1 - 1\Bigg].\end{aligned} \tag{10}$$

We implemented Eq. 10 in Tensorflow version 1.4.1 with Keras version 2.1.3 as the model-level library. We solved Eq. 10 using the Adam optimization algorithm [53] (with batch normalization) from the python package `keras-gpu` version 2.1.3 with parameters `learning_rate` $= 2 \times 10^{-3}$, `beta_1` $= 0.9$, and `beta_2` $= 0.999$, to obtain $(\widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\phi}})$. Then, for each tumor t, we used a single sample $\boldsymbol{Z}|\boldsymbol{x}_t = \boldsymbol{z}_t$ from the distribution $\mathcal{N}(\boldsymbol{\mu}_{\widehat{\boldsymbol{\theta}}}(\boldsymbol{x}_t), \operatorname{diag}(\boldsymbol{\sigma}_{\widehat{\boldsymbol{\theta}}}(\boldsymbol{x}_t)))$ as the final latent-space encoding of the tumor to be used for supervised learning ("Regularized gradient boosted decision trees (XGBoost)" section).

VAE model architectures

We trained six transcriptome-encoding VAEs based on three VAE architectures, the pan-cancer VAE architecture (for the 33-cancer unsupervised analysis, "VAE encoding preserves cancer type features" section) and three cancer type-specific VAE architectures for response-to-chemotherapy prediction ("Chemotherapy response classification results" section) (`VAE-1` was used for two different cancer types, BRCA and PAAD, `VAE-2` was used for COAD, and `VAE-3` was used for two different cancer types, BLCA and SARC). For the pan-cancer, we used the `VAE-1` model with a latent space dimension $h = 50$. For the cancer type-specific VAE architectures, we again used the same number of fully-connected layers in the encoder as in the decoder (Table 5).

Labeling tumors based on response to chemotherapy

From Xena and cBioPortal [54, 55], we obtained and combined TCGA clinical data (where available) for the patients whose tumor transcriptomes we acquired ("Gene expression data" section). From Xena, we extracted the variables `submitter_id.samples`, `therapy_type`, and `measure_of_response`; from cBioPortal, we extracted the variables `Sample_ID`, `Disease.Free.Status`, and `Pharmaceutical.Therapy.Indicator`. We co-analyzed the Xena- and cBioPortal-obtained clinical data to label tumors "responded" ($y = 0$) or "progressive" ($y = 1$), by assigning $y = 0$ when the clinical record had `Complete response` or `partial response` in the `measure_of_response` column of the clinical data

Table 6 XGBoost classification algorithm hyperparameters and hyperparameter ranges used in grid-search tuning

Hyperparameter name	Hyperparameter description	Hyperparameter range
`n_estimators`	Number of trees to fit	(1, 2, 3, . . ., 40)
`max_depth`	Maximum tree depth	(1, 2, 3, . . ., 10)
`learning_rate`	Boosting learning rate	(0.05, 0.1, 0.2, 0.4, 0.6, 0.8)
`min_child_weight`	Minimum sum of instance weight needed in a child	(1, 2, 3, . . ., 10)
`subsample`	Sub-sample ratio of the training instance	(0.1, 0.2, 0.3, . . ., 1.0)
`colsample_bytree`	Sub-sample ratio of columns when constructing each tree	(0.1, 0.2, 0.3, . . ., 1.0)
`reg_alpha`	Coefficient of L1 regularization for the node weights	(0, 1, 2, 3)
`reg_lambda`	Coefficient of L2 regularization for the node weights	(1, 2, . . ., 100)

Table 7 SVM classification algorithm hyperparameters and hyperparameter ranges used in grid-search tuning

Hyperparameter name	Hyperparameter description	Hyperparameter range
`kernel`	Kernel type to be used	('linear', 'poly', 'rbf', 'sigmoid')
`C`	Regularization parameter	(5, 6, 7, . . ., 50)
`degree`	Degree of the polynomial kernel function ('poly')	(1, 2, 3, . . ., 20)

from Xena, or with value `DiseaseFree` in the `Disease.Free.Status` column of the clinical data from cBioPortal while therapy type is recorded as `Chemotherapy` in both. We assigned $y = 1$ to tumors whose clinical records had values `Radiographic progressive disease`, `Clinical progressive disease`, or `stable disease` in the Xena clinical data column `measure_of_response`, or had value `Recurred/progressed` in the cBioPortal data column `Disease.Free.Status` while the `therapy_type` is recorded as `Chemotherapy` in both files. This yielded 806 labeled tumors out of 2,606 total. A total of 39 different drugs were used to treat the 794 patients (Additional file 1: Table S1).

Regularized gradient boosted decision trees (XGBoost)

For predicting whether or not (based on its transcriptome-derived feature-set: raw, PCA, ICA, or VAE) a tumor would respond to chemotherapy, we used XGBoost [34], an efficient implementation of regularized gradient boosted decision trees. We used the classifier function `XGBClassifier` from the python software package `xgboost` version 0.80, with `gamma=0`. We tuned eight hyper-parameters (Table 6) by exhaustive grid-search with five-fold cross-validation, using `model_selection.GridSearchCV` from `scikit-learn` version 0.19.1. To obtain feature importance scores, we used `get_score` with `importance_type = cover`.

Principal component analysis (PCA) and independent component analysis (ICA)

For PCA, we used the function `decomposition.PCA` (with parameters `svd_solver="full"`) and `n_components=0.9` (90% variance, yielding 387 components) from the python package `scikit-learn` version 0.19.1. For plotting, we used `matplotlib` version 2.1.2. For ICA, we used the function `decomposition.`

Table 8 KNN classification algorithm hyperparameters and hyperparameter ranges used in grid-search tuning

Hyperparameter name	Hyperparameter description	Hyperparameter range
`n_neighbors`	Number of neighbors to use	(1, 2, 3, . . ., 20)
`weights`	Weight function used in prediction	('uniform', 'distance')
`algorithm`	Algorithm used to compute the nearest neighbors	('ball_tree', 'kd_tree', 'brute', 'auto')
`leaf_size`	Leaf sized passed to BallTree or KDTree	(1, 2, 3, . . ., 20)
`p`	Power parameter for the Minkowski metric	(1, 2, 3, 4)

`FastICA` (with parameters `n_components`=387 (i.e., the same number of components as used in the PCA method) from the python package `scikit-learn` version 0.19.1. For plotting, we used `matplotlib` version 2.1.2.

Support vector machine (SVM)

For predicting whether or not (based on its transcriptome-drived feature-set: raw or VAE) a tumor would respond to chemotherapy, we used SVM [56]. We used the classifier function `SVC` from the python software package `sklearn.svm`, with `gamma = "auto"`. We tuned three hyper-parameters (Table 7) by exhaustive grid-search with five-fold cross-validation, using `model_selection.GridSearchCV` from `scikit-learn` version 0.19.1.

K-nearest neighbors vote (KNN)

For predicting whether or not (based on its transcriptome-drived feature-set: raw or VAE) a tumor would respond to chemotherapy, we used KNN [57], an implementation based on the ***k*** nearest neighbors of each query point. We used the classifier function `neighbors.KNeighborsClassifier` from the python software package `scikit-learn`. We tuned five hyper-parameters (Table 8) by exhaustive grid-search with five-fold cross-validation, using `model_selection.GridSearchCV` from `scikit-learn` version 0.19.1.

Area under ROC curve (AUROC)

For computing the AUROC (i.e., sensitivity versus false positive error rate curve), we used the function `metrics.roc_auc_score` from the python software package `scikit-learn` version 0.19.1 with parameter `average="weighted"`. We logit-transformed AUROC values before testing (using two-tailed Welch's *t*-test). For the L1 versus L2 analysis ("L1 loss is better than L2 loss and cross-entropy loss for this application" section), we carried out 30 replications of five-fold cross-validation; within each replication, across the five folds, we obtained prediction scores for each tumor from the fold in which the tumor was in the test set, enabling us to compute an overall AUROC within each replication. For each training data set, we carried out 30 replications of five-fold cross-validation by altering the random seed used for assigning data to folds, during the cross-validation. We used the same procedure for five different cancer types (BLCA, BRCA, COAD, PAAD, SARC) as shown in the panel names of Additional file 1: Figure S7.

Area under the precision-recall curve (AUPRC)

For computing the AUPRC, we used the function `metrics.precision_recall_curve` and `metrics.auc` from the python software package `scikit-learn` version 0.19.1. We logit-transformed AUPRC values before testing (using two-tailed Welch's *t*-test). We carried out 30 replications of five-fold cross-validation; within each replication, across the five folds, we obtained prediction scores for each tumor from the fold in which the tumor was in the test set, enabling us to compute an overall AUPRC within each replication. For each training data set, we have done 30 replications of five-fold cross-validation by altering the random seed used for assign split of data during cross-validation. We have conducted the same procedure for five different cancer types (BLCA, BRCA, COAD, PAAD, SARC) as shown in the panel names of Additional file 1: Figure S8.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s12859-021-04339-6.

Additional file 1. This supplementary file contains Supplementary Figures S1, S2, S3, S4, S5, S6, S7, and S8, as well as Table S1 and Supplementary Note C.

Acknowledgements
Not applicable.

Authors' contributions
Designed the study: SAR and QW; wrote the software: QW; carried out the computational analyses: QW; processed and analyzed the data: QW and SAR; wrote the article: SAR and QW. Both authors read and approved the final manuscript.

Funding
SAR acknowledges support from the Animal Cancer Foundation.

Availability of data and materials
Software code written for this project "VAE for chemotherapy drug response prediction" are freely available under an open-source license, platform independent, written in Python and R with CUDA and tensorflow installed, at the URL: https://github.com/ATHED/VAE_for_chemotherapy_drug_response_prediction. Supplementary data are available at *BMC Bioinformatics* online.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Author details
[1]School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR, USA. [2]Department of Biomedical Sciences, Oregon State University, Corvallis, OR, USA.

Received: 30 April 2021 Accepted: 17 August 2021
Published online: 22 September 2021

References

1. Airley R. Cancer chemotherapy. New York: Wiley-Blackwell; 2009.
2. Skeel RT. Handbook of cancer chemotherapy. 6th ed. Philadelphia: Lippincott Williams & Wilkins; 2003.
3. Chabner BA, Longo DL. Cancer chemotherapy and biotherapy: principles and practice. 4th ed. Philadelphia: Lippincott Willians & Wilkins; 2005.
4. Kaestner SA, Sewell GJ. Chemotherapy dosing part I: scientific basis for current practice and use of body surface area. Clin Oncol. 2007;19:23–37. https://doi.org/10.1016/j.clon.2006.10.010.

5. Gurney H. How to calculate the dose of chemotherapy. Br J Cancer. 2002;86:1297–302. https://doi.org/10.1038/sj.bjc.6600139.
6. Corrie PG. Cytotoxic chemotherapy: clinical aspects. Medicine. 2008;36(1):24–8. https://doi.org/10.1016/j.mpmed.2007.10.012.
7. Whelan T, Sawka C, Levine M, Gafni A, Reyno L, Willan A, Julian J, Dent S, Abu-Zahra H, Chouinard E, Tozer R, Pritchard K, Bodendorfer I. Helping patients make informed choices: a randomized trial of a decision aid for adjuvant chemotherapy in lymph node-negative breast cancer. JNCI: J Natl Cancer Inst. 2003;95(8):581–7. https://doi.org/10.1093/jnci/95.8.581.
8. Malfuson J-V, Etienne A, Turlure P, de Revel T, Thomas X, Contentin N, Terré C, Rigaudeau S, Bordessoule D, Vey N, Gardin C, Dombret H. for the Acute Leukemia French Association (ALFA): risk factors and decision criteria for intensive chemotherapy in older patients with acute myeloid leukemia. Haematologica. 2008;93(12):1806–13. https://doi.org/10.3324/haematol.13309.
9. Chiu Y-C, Chen H-IH, Zhang T, Zhang S, Gorthi A, Wang L-J, Huang Y, Chen Y. Predicting drug response of tumors from integrated genomic profiles by deep neural networks. BMC Med Genom. 2019;12(1):18. https://doi.org/10.1186/s12920-018-0460-9.
10. Geeleher P, Cox NJ, Huang RS. Clinical drug response can be predicted using baseline gene expression levels and in vitrodrug sensitivity in cell lines. Genome Biol. 2014;15(3):47. https://doi.org/10.1186/gb-2014-15-3-r47.
11. Weir B, Zhao X, Meyerson M. Somatic alterations in the human cancer genome. Cancer Cell. 2004;6(5):433–8. https://doi.org/10.1016/j.ccr.2004.11.004.
12. Gámez-Pozo A, Trilla-Fuertes L, Prado-Vázquez G, Chiva C, López-Vacas R, Nanni P, Berges-Soria J, Grossmann J, Díaz-Almirón M, Ciruelos E, Sabidó E, Espinosa E, Fresno VJ. Prediction of adjuvant chemotherapy response in triple negative breast cancer with discovery and targeted proteomics. PLoS ONE. 2017;12:6. https://doi.org/10.1371/journal.pone.0178296.
13. Casado E, García VM, Sánchez JJ, Blanco M, Maurel J, Feliu J, Fernández-Martos C, de Castro J, Castelo B, Belda-Iniesta C, Sereno M, Sánchez-Llamas B, Burgos E, Ángel García-Cabezas M, Manceñido N, Miquel R, García-Olmo D, González-Barón M, Cejas P. A combined strategy of SAGE and quantitative PCR provides a 13-gene signature that predicts preoperative chemoradiotherapy response and outcome in rectal cancer. PLoS ONE. 2011;17:4145–54. https://doi.org/10.1158/1078-0432.CCR-10-2257.
14. Del Rio M, Molina F, Bascoul-Mollevi C, et al. Gene expression signature in advanced colorectal cancer patients select drugs and response for the use of leucovorin, fluorouracil, and irinotecan. J Clin Oncol. 2007;25(7):773–8. https://doi.org/10.1200/JCO.2006.07.4187.
15. Kurokawa Y, Matoba R, Nagano H, Sakon M, Takemasa I, Nakamori S, Dono K, Umeshita K, Ueno N, Ishii S, Kato K, Monden M. Molecular prediction of response to 5-fluorouracil and interferon-*a* combination chemotherapy in advanced hepatocellular carcinoma. AACR. 2004;10(18):6029–38. https://doi.org/10.1158/1078-0432.CCR-04-0243.
16. Rezaeian I, Eliseos JM, Katherina B, Huy QP, Iman R, Dimo A, Alioune N, Luis R, Peter KR. Predicting outcomes of hormone and chemotherapy in the molecular taxonomy of breast cancer international consortium (METABRIC) study by biochemically-inspired machine learning. F1000Research. 2017;5:2124. https://doi.org/10.12688/f1000research.9417.3.
17. Hutter C, Zenklusen JC. The cancer genome atlas: creating lasting value beyond its data. Cell. 2018;173(2):283–5.
18. Wen H, Huang F. Personal loan fraud detection based on hybrid supervised and unsupervised learning. In: 2020 5th IEEE international conference on big data analytics (ICBDA); 2020. p. 339–343 https://doi.org/10.1109/ICBDA49040.2020.9101277
19. Qin J, Li Y, Liu Q. ICA based semi-supervised learning algorithm for BCI systems. In: Rosca J, Erdogmus D, Príncipe JC, Haykin S, editors. Independent component analysis and blind signal separation. Berlin: Springer; 2006. p. 214–21.
20. An J, Cho S. Variational autoencoder based anomaly detection using reconstruction probability. Technical Report SNUDM-TR-2015-03, Seoul National University. 2015. http://dm.snu.ac.kr/static/docs/TR/SNUDM-TR-2015-03.pdf.
21. Li X, She J. Collaborative variational autoencoder for recommender systems. In: Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining. ACM, New York, NY; 2017. p. 305–314. https://doi.org/10.1145/3097983.3098077.
22. Bouchacourt D, Tomioka R, Nowozin S. Multi-level variational autoencoder: learning disentangled representations from grouped observations. arXiv:1705.08841 2017.
23. Kipf TN, Welling M. Variational graph auto-encoders. arXiv:1611.07308 2016.
24. Kingma DP, Welling M. Auto-encoding variational bayes. arxiv:1312.6114 2013.
25. Jimenez Rezende D, Mohamed S, Wierstra D. Stochastic backpropagation and approximate inference in deep generative models. arXiv:1401.4082 2014.
26. Way GP, Greene CS. Extracting a biologically relevant latent space from cancer transcriptomes with variational autoencoders. Pac Symp Biocomput. 2018;23:80–91. https://doi.org/10.1142/9789813235533_0008.
27. Titus AJ, Wilkins OM, Bobak CA, Christensen BC. Unsupervised deep learning with variational autoencoders applied to breast tumor genome-wide DNA methylation data with biologic feature extraction. Cold Spring Harbor Laboratory. bioRxiv. 2018. https://doi.org/10.1101/433763.
28. Way GP, Greene CS. Evaluating deep variational autoencoders trained on pan-cancer gene expression. arXiv:1711.04828 2017.
29. George TM, Lio P. Unsupervised machine learning for data encoding applied to ovarian cancer transcriptomes. Cold Spring Harbor Laboratory. bioRxiv. 2019. https://doi.org/10.1101/855593.
30. Dincer AB, Celik S, Hiranuma N, Lee S-I. Deepprofile: Deep learning of cancer molecular profiles for precision medicine. bioRxiv. 2018. https://doi.org/10.1101/278739.
31. Theodore S, Konstantinos V, Sonali N, Filippos K, Athanassios K, Alexander P, Tyler JM, et al. A deep learning framework for predicting response to therapy in cancer. Cell Rep. 2019;29(11):3367–33734. https://doi.org/10.1016/j.celrep.2019.11.017.
32. Liu P, Li H, Li S, Leung K-S. Improving prediction of phenotypic drug response on cancer cell lines using deep convolutional network. BMC Bioinform. 2019;20(1):408. https://doi.org/10.1186/s12859-019-2910-6.

33. Ladislav R, Daniel H, Petr S, Benjamin H-K, Anna G. Dr.VAE: improving drug response prediction via modeling of drug perturbation effects. Bioinformatics. 2019;35(19):3743–51. https://doi.org/10.1093/bioinformatics/btz158.
34. Chen T, Guestrin C. XGBoost: A scalable tree boosting system. arXiv:1603.02754 2016.
35. Dolezal JM, Dash AP, Prochownik EV. Diagnostic and prognostic implications of ribosomal protein transcript expression patterns in human cancers. BMC Cancer. 2018;18(1):275. https://doi.org/10.1186/s12885-018-4178-z.
36. Esteva A, Kuprel B, Novoa RA, Ko J, Swetter SM, Blau HM, Thrun S. Dermatologist-level classification of skin cancer with deep neural networks. Nature. 2017;542(7639):115–8. https://doi.org/10.1038/nature21056.
37. Mitchel J, Chatlin K, Tong L, Wang, MD. A translational pipeline for overall survival prediction of breast cancer patients by decision-level integration of multi-omics data. In: 2019 IEEE international conference on bioinformatics and biomedicine (BIBM); 2019. p. 1573–1580. https://doi.org/10.1109/BIBM47256.2019.8983243
38. Zhang Y, Feng T, Wang S, Dong R, Yang J, Su J, Wang B. A novel xgboost method to identify cancer tissue-of-origin based on copy number variations. Front Genet. 2020;11:1319. https://doi.org/10.3389/fgene.2020.585029.
39. Lee K, Jeong H-O, Lee S, Jeong W-K. CPEM: Accurate cancer type classification based on somatic alterations using an ensemble of a random forest and a deep neural network. Sci Rep. 2019;9(1):16927. https://doi.org/10.1038/s41598-019-53034-3.
40. Evans C, Hardin J, Stoebel DM. Selecting between-sample RNA-Seq normalization methods from the perspective of their assumptions. Brief Bioinform. 2017;19(5):776–92. https://doi.org/10.1093/bib/bbx008.
41. Hu Q, Greene CS. Parameter tuning is a key part of dimensionality reduction via deep variational autoencoders for single cell rna transcriptomics. bioRxiv. 2018. https://doi.org/10.1101/385534.
42. Lakhmiri D, Alimo R, Le Digabel S. Tuning a variational autoencoder for data accountability problem in the Mars Science Laboratory ground data system. arxiv:2006.03962 2020.
43. Senft D, Leiserson MDM, Ruppin E, Ronai ZA. Precision oncology: the road ahead. Trends Mol Med. 2017;23(10):874–98. https://doi.org/10.1016/j.molmed.2017.08.003.
44. Marchiano EJ, Birkeland AC, Swiecicki PL, Spector-Bagdady K, Shuman AG. Revisiting expectations in an era of precision oncology. Oncologist. 2018;23(3):386–8. https://doi.org/10.1634/theoncologist.2017-0269.
45. Massard C, Michiels S, Ferté C, Le Deley M-C, Lacroix L, Hollebecque A, Verlingue L, Ileana E, Rosellini S, Ammari S, Ngo-Camus M, Bahleda R, Gazzah A, Varga A, Postel-Vinay S, Loriot Y, Even C, Breuskin I, Auger N, Job B, De Baere T, Deschamps F, Vielh P, Scoazec J-Y, Lazar V, Richon C, Ribrag V, Deutsch E, Angevin E, Vassal G, Eggermont A, André F, Soria J-C. High-throughput genomics and clinical outcome in hard-to-treat advanced cancers: results of the moscato 01 trial. Cancer Discov. 2017;7(6):586–95. https://doi.org/10.1158/2159-8290.CD-16-1396.
46. Goldman M, Craft B, Hastie M, Repečka K, McDade F, Kamath A, Banerjee A, Luo Y, Rogers D, Brooks AN, Zhu J, Haussler D. The UCSC Xena platform for public and private cancer genomics data visualization and interpretation. Cold Spring Harbor Laboratory. bioRxiv. 2019. https://doi.org/10.1101/326470.
47. Dillies M-A, Rau A, Aubert J, Hennequet-Antier C, Jeanmougin M, Servant N, Keime C, Marot G, Castel D, Estelle J, et al. A comprehensive evaluation of normalization methods for Illumina high-throughput RNA sequencing data analysis. Brief Bioinform. 2013;14(6):671–83. https://doi.org/10.1093/bib/bbs046.
48. R Core Team. R: A language and environment for statistical computing. Vienna: R Foundation; 2013. (**ISBN 3-900051-07-0**).
49. Kreyszig E, Kreyszig H, Norminton EJ. Advanced engineering mathematics. 10th ed. Hoboken: Wiley; 2011.
50. Kramer MA. Nonlinear principal component analysis using autoassociative neural networks. AIChE J. 1991;37(2):233–43. https://doi.org/10.1002/aic.690370209.
51. Zemouri R, Lévesque M, Amyot N, Hudon C, Kokoko O, Tahan SA. Deep convolutional variational autoencoder as a 2d-visualization tool for partial discharge source classification in hydrogenerators. IEEE Access. 2020;8:5438–54. https://doi.org/10.1109/ACCESS.2019.2962775.
52. Duchi J. Derivations for linear algebra and optimization. Technical report, Standford University. 2007. http://web.stanford.edu/~jduchi/projects/general_notes.pdf.
53. Kingma DP, Ba J. Adam: a method for stochastic optimization. arXiv:1412.6980 2014.
54. Cerami E, Gao J, Dogrusoz U, Gross BE, Sumer SO, Aksoy BA, Jacobsen A, Byrne CJ, Heuer ML, Larsson E, Antipin Y, Reva B, Goldberg AP, Sander C, Schultz N. The cBio cancer genomics portal: an open platform for exploring multidimensional cancer genomics data. Cancer Discov. 2012;2:401. https://doi.org/10.1158/2159-8290.CD-12-0095.
55. Gao J, Aksoy BA, Dogrusoz U, Dresdner G, Gross B, Sumer SO, Sun Y, Jacobsen A, Sinha R, Larsson E, Cerami E, Sander C, Schultz N. Integrative analysis of complex cancer genomics and clinical profiles using the cBioPortal. Sci Signal. 2013;6:11. https://doi.org/10.1126/scisignal.2004088.
56. Cortes C, Vapnik V. Support-vector networks. Mach Learn. 1995;20(3):273–97. https://doi.org/10.1023/A:1022627411411.
57. Goldberger J, Roweis S, Hinton G, Salakhutdinov R. Neighbourhood components analysis. In: Proceedings of the 17th international conference on neural information processing systems. NIPS'04. MIT Press, Cambridge, MA, USA; 2004. p. 513–520. https://doi.org/10.5555/2976040.2976105

Publisher's Note