HHS Public Access

Author manuscript

Nat Mach Intell. Author manuscript; available in PMC 2021 January 01.

Published in final edited form as:

Nat Mach Intell. 2020 July ; 2(7): 387–395. doi:10.1038/s42256-020-0193-2.

Gaussian Embedding for Large-scale Gene Set Analysis

Sheng Wang[1], **Emily R. Flynn**[2], **Russ B. Altman**[1,2,3,*]

[1]Department of Bioengineering, Stanford University, Stanford, CA 94305, USA.

[2]Biomedical Informatics Training Program, Stanford University, Stanford, CA 94305, USA.

[3]Department of Genetics, Stanford University, Stanford, CA 94305, USA.

Abstract

Gene sets, including protein complexes and signaling pathways, have proliferated greatly, in large part as a result of high-throughput biological data. Leveraging gene sets to gain insight into biological discovery requires computational methods for converting them into a useful form for available machine learning models. Here, we study the problem of embedding gene sets as compact features that are compatible with available machine learning codes. We present Set2Gaussian, a novel network-based gene set embedding approach, which represents each gene set as a multivariate Gaussian distribution rather than a single point in the low-dimensional space, according to the proximity of these genes in a protein-protein interaction network. We demonstrate that Set2Gaussian improves gene set member identification, accurately stratifies tumors, and finds concise gene sets for gene set enrichment analysis. We further show how Set2Gaussian allows us to identify a previously unknown clinical prognostic and predictive subnetwork around NEFM in sarcoma, which we validate in independent cohorts.

Introduction

One of the most basic outcomes of biological data analysis is the discovery of gene sets. Such sets come from many sources[1–4]: Genome-wide association studies (GWAS) produce sets of genes associated with a disease or other phenotype. Gene expression analyses identify gene sets by examining differential expression between conditions, or clustering genes by expression similarity. Proteomics and metabolomics data also produce lists of proteins and metabolites. Biological network analysis associates genes, proteins, or

*Correspondence to Russ B. Altman at russ.altman@stanford.edu.

Contributions

All authors conceived the problem. S.W. conceived the algorithm and performed the computational experiments. R.A. led the research. All authors wrote the manuscript.

Data availability

We also provided pre-trained gene set representations of all gene sets in NCI, Reactome, and MSigDB at https://doi.org/10.6084/m9.figshare.11341181.v1. All results in this paper are based on these representations.

Code availability

A software implementation of Set2Gaussianis is available at http://doi.org/10.5281/zenodo.3827929.

Competing interests

R.B.A. declares the following competing interests: stock or other ownership (Personalis, 23andme, Youscript); consulting or advisory role (United Health, Second Genome, Karius, UK Biobank, Swiss Personalized Health Network).

metabolites that interact with one another into network neighborhoods. In all of these cases, the hypothesis is that genes in the set are likely involved in the same biological process or function. Because of their ability to boost signal-to-noise and increase explanatory power, gene sets have been used in various downstream analyses, including disease signature identification[5,6], drug pathway association prediction[7], survival analysis[8], and drug response prediction[9]. Unfortunately, a bottleneck for gene set-based analysis is a lack of computational tools that convert these gene sets into informative forms which can distinguish completely different gene sets deemed as the same by averaging embedding (Figure 1a). In particular, existing machine learning codes require input features in the form of fixed length vectors, and prefer these vectors to be compact and high-quality in order to avoid overfitting.

Molecular interaction networks provide novel insights into the functional interdependencies of genes and proteins[10]. In particular, high-throughput experimental techniques, such as yeast two-hybrid screens and genetic interaction assays, have enabled researchers to piece together large-scale interaction networks in bulk[11]. Consequently, network-based approaches, including network propagation[12–18], network clustering[19], network integration[20,21], and network regularization[22], have been developed to efficiently analyze these networks. Among them, network embedding has emerged as a powerful network analysis approach because it generates a highly informative and compact vector representation for each node in the network[21,23]. Molecular interaction networks are noisy and incomplete, especially as they increase in size[21,23]. Network embedding typically leverages dimensionality reduction techniques to regularize high-dimensional network data. Prior to the advent of network embedding approaches, researchers manually identified network features for machine learning, which was time-consuming and often required expert knowledge. By contrast, network embedding automates this process by embedding each node in the network as a vector. These node embeddings have shown good performance in machine learning classifiers, and have become the building blocks in a large number of systems biology applications[21,23]. In this paper, we examine networks with genes as nodes; however, it is important to note that these methods can be applied to networks with any type of node, such as networks with disease nodes[10] or drug nodes[24].

Gene embeddings represent each gene as a fixed length vector in a common low-dimensional continuous vector space. The dimension of these gene embeddings is typically much smaller than the original data dimensions (e.g., the number of nodes in the network). While many useful gene embeddings are now available[21,23], learning representations for gene sets remains challenging. One simple approach to represent a gene set is to averaging individual gene representations that comprise the gene set. In natural language processing, such averaging of word vectors has been used to construct sentence embeddings[25]. In computer vision, deep learning architectures rely on pooling which aggregates nearby pixels by taking the average, max, or weighted average of their values[26]. In network modeling, graph neural network models also use pooling to aggregate information from the neighborhood to effectively propagate information throughout the network[27]. However, although simple and intuitive, averaging and other pooling operations may not be sufficiently expressive for representing gene sets and many underfit the data. By contrast with sentences that only have a few words or images where we only average a few nearby

pixels, gene sets can be arbitrarily large. In part due to this size difference, simple aggregation methods are not expressive enough to represent such gene sets. Figure 1a provides an example where the average embedding approach is unable to distinguish two completely different gene sets. A second intuitive approach to represent a gene set is to add new nodes representing the entire gene sets into the existing molecular network and connect these "gene set nodes" to their gene members. One can then run node embedding on this heterogeneous network to obtain the representation of each gene set. However, adding these likely high-degree nodes to the network can substantially change the topological structure and connectivity of the network, leading to inaccuracy in the embedding space. Other methods aim at embedding densely connected subnetworks: ComE performs community embedding and community detection simultaneously using a community-aware high-order proximity[28]; PathEmb models pathways as documents and then applies document embedding models to calculate pathway similarity[29]. However, both ComE and PathEmb require gene sets to be fully connected in the network, which is not the case for most of the biologically meaningful gene sets.

Fundamentally, all of these approaches assume that genes in the same set tend to have similar properties. This is intuitively the case for protein complexes or biological processes; however, this is not true for a large number of other biologically meaningful gene sets. To examine the diversity of functions in gene sets, we calculated the Gene Ontology (GO) enrichment of 150 drug response-related gene sets derived from two pharmacogenomics datasets (Figure 1b)[30]. We found that 86% of these gene sets were significantly enriched with more than one GO function (P-value <0.05 after Bonferroni correction). This indicates that genes in the same set frequently had different functions and were involved in multiple biological processes; however, this diversity is ignored in average embedding and other simple aggregation methods. Recently, Gaussian embedding, which represents each node as a multivariate Gaussian distribution in the low-dimensional space, has been proposed to model the uncertainty of nodes[31–33]. Motivated by prior work on Gaussian embedding of nodes, we propose to represent each gene set as a multivariate Gaussian distribution according to its network topology. This allows us to model the diversity by the uncertainty of each dimension in this vector. To our knowledge, our method is the first approach that learns compact representations for gene sets.

Thus, we present Set2Gaussian, a novel computational method that summarizes genes in the same set closely as a multivariate Gaussian distribution in the low-dimensional space (Figure 1c). Set2Gaussian takes biological networks and a collection of gene sets as input. Each gene is represented as a single point and each gene set as a multivariate Gaussian distribution parameterized by a low-dimensional mean vector and a low-dimensional covariance matrix. The mean vector of each gene set describes the joint contribution of genes in this gene set, and the covariance matrix characterizes the agreement among individual genes in each dimension. Dimensions that have small variance across different genes in the set should have higher weights when they are used to calculate the similarity between two gene sets. In contrast to using a diagonal matrix to represent the covariance matrix in previous work[31], we propose to use a low-rank matrix in order to avoid underfitting. Set2Gaussian is able to differentiate between gene sets that would be considered equivalent by conventional approaches, such as mean pooling. We demonstrate

that Set2Gaussian significantly improves gene set member identification in three large-scale gene set collections. We further show how the embeddings generated by Set2Gaussian allows us to identify a previously unknown clinical prognostic and predictive subnetwork around NEFM in sarcoma, providing insight into the treatment of sarcoma. Finally, we use Set2Gaussian to select concise previously defined gene sets that substantially enhance and accelerate gene set enrichment analysis.

Results

Overview of Set2Gaussian

A key observation behind our approach is that gene sets can have diverse molecular functions and/or biological processes. Set2Gaussian explicitly models this diversity as a low-dimensional Gaussian distribution which summarizes both location and uncertainty of each dimension, improving upon the expressive power of existing node embedding models. Set2Gaussian takes a network and a collection of gene sets as input (Figure 1c). It then finds a low-dimensional space in which genes and gene sets preserve their distances in the network. Each gene is represented as a single point in the low-dimensional space. Each gene set is represented as a multivariate Gaussian distribution which is parameterized by a mean vector and covariance matrix. In our work, we approximated the covariance matrix through low-rank matrix factorization in order to avoid underfitting, as we demonstrated in our experiments.

Set2Gaussian improves gene set member identification

We first sought to examine whether Set2Gaussian could accurately identify gene set members. Genes in the same gene set are analyzed together to study the underlying biological processes. However, gene sets, especially those generated from high-throughput experiments, may contain a substantial number of false positives and false negatives due to the robustness and quality of the assay. Therefore, reducing the noise by identifying gene set members is an important task for large-scale gene set analysis. Because labeling manually is expensive and tedious, many computational approaches have been proposed to identify gene set members, each utilizing different information, including domain signatures[34], gene expression[35], and sequence data[36].

Set2Gaussian significantly outperformed the proposed baseline gene set representation approaches on identifying gene set members in all three datasets at all size categories (P-value<0.05; Wilcoxon signed-rank test) (Figure 2a, b, c). We first observed clear improvements over three pooling-based approaches in all three datasets. For example, In Reactome, at the medium set [11–30], Set2Gaussian obtained an AUPRC of 0.48, outperforming mean (0.42), weighted mean (0.40), and max (0.22). The above results suggest that representing a gene set through mean or max pooling is unable to model the uncertainty and capture the diverse functions within a gene set, leading to worse performance. We further noticed increasing improvement with larger gene set size. For example, Set2Gaussian obtained 36% improvement on large sets [31–1000] versus 21% improvement on small sets [3–10] in comparison to mean on MSigDB. Larger gene sets may be noisier and contain more diverse functions. Overall, this comparison indicated that using

a single vector to represent a gene set, as in these pooling approaches, is not expressive enough to capture gene set diversity. By adopting a covariance matrix to model uncertainty, Set2Gaussian is able to model this diversity and thus substantially improves gene set identification performance.

Additionally, Set2Gaussian outperformed hypergraph embedding on gene set identification across all gene set sizes in all three datasets. For example, in MSigDB, Set2Gaussian obtained 0.13 AUPRC for small sets [3–10], which was significantly better than hypergraph embedding (AUPRC = 0.032). Interestingly, we found that hypergraph embedding was worse than mean and weighted mean pooling approaches in all three datasets. Network embedding approaches rely on finding similar contexts (e.g., similar neighbors or similar diffusion states) to accurately embed nodes. However, in a hypergraph, a “gene set node” could have a noisy neighborhood structure due to a large number of neighbors, which may make it difficult to find enough nodes with similar contexts to support an accurate embedding. Constructing a hypergraph may also introduce too many high degree nodes, substantially changing the topological structure of the network.

A key feature of Set2Gaussian is that it leverages a low-rank matrix instead of a diagonal matrix to parametrize the covariance in the Gaussian distribution of each gene set. A low-rank covariance matrix is more expressive but could also be more prone to overfitting. To examine whether it is necessary to increase model complexity given the risk of overfitting, we compared Set2Gaussian with Set2Gaussian-diag, which forced the covariance matrix to be a diagonal matrix. We found that Set2Gaussian had improved performance over Set2Gaussian-diag (AUPRC of 0.24 vs. 0.20 respectively) for medium sets [11–30] in Reactome (Figure 2d). Moreover, the improvement of Set2Gaussian against Set2Gaussian-diag was larger on MSigDB than the other two datasets (Supplementary Figure 1). We postulated that because MSigDB had the largest number of gene sets, a diagonal covariance matrix does not accurately project these gene sets into the same low-dimensional space and thus underfits the data. Notably, although worse than Set2Gaussian, Set2Gaussian-diag still performed better than other gene set representation approaches including Mean. This demonstrates the importance of modeling the uncertainty of each dimension by using a Gaussian distribution.

In addition to accurately recovering existing gene set collections, the top predictions made by Set2Gaussian also revealed novel gene set members that could supplement existing pathway databases. For example, the NLRP1 inflammasome pathway in Reactome is known to have three genes, BCL2, BCL2L1, and NLRP1. All these three genes were included in the top ten predictions of Set2Gaussian. In addition, Set2Gaussian predicted that BAX, CASP1, MCL1, BCL2A1, CASP9, CASP3, and NLRP1 were within the multidimensional Gaussian distribution described by the original set and thus might be the members of this pathway. These predictions were supported by the literature; NLRP1 activates a downstream protease Caspase-1 (CASP1)[37]. BAX, BCL2A1, and MCL1 belong to the BLC2 family, which inhibits NLRP1 induced CASP1 activation in a concentration-dependent manner[38].

Set2Gaussian identifies novel subtypes and subnetworks in Sarcoma

We next applied Set2Gaussian to tumor stratification and subnetwork identification in Sarcoma. Tumor stratification aims at dividing a heterogeneous population of tumors into clinically and biologically meaningful subtypes. We modeled each tumor as a gene set denoted by the tumor's set of mutations and then clustered these gene sets based on the gene set representations derived from Set2Gaussian. We found that Set2Gaussian's subtypes had significantly different survival in Sarcoma across groups; while subtypes from the other three approaches did not (Figure 3a). For example, at k=2 subtypes, the survival times were significantly different between subtypes (P-value <0.03, log-rank test); and five-year survival rates varied greatly (41% and 71% respectively) (Figure 3b). We attributed Set2Gaussian's superior performance against comparison approaches to its projection of mutation profiles as low-dimensional Gaussian distributions, thus obtaining more accurate similarity between tumors.

By finding the top differentially mutated genes between two subtypes that have the highest and lowest survival rates, we further identified a connected subnetwork of five genes in protein-protein interaction networks from the STRING database, including NEFM, FREM2, KIF1A, NRXN1, and DSCAM (Figure 3c). We named this subnetwork NEFM-subnetwork since it was central around NEFM. The subtype with a larger NEFM-subnetwork mutation rate tends to have worse survival in comparison to other subtypes. Since this subnetwork seemed to be associated with Sarcoma survival, we divided the 235 Sarcoma tumors into two subtypes based on whether at least one of NEFM-subnetwork genes was mutated in a given tumor. We found that dividing tumors using only these five genes obtained subtypes that had significantly different overall survival (P-value < 0.03, log-rank test) (Figure 3d), and subtypes had very different five-year survival rates (49% for NEFM-subnetwork mutated vs. 87% for NEFM-subnetwork not mutated).

After demonstrating that the NEFM-subnetwork stratifies sarcoma tumors by survival, we investigated if this subnetwork could be used as a biomarker for chemosensitivity prediction. To this end, we first collected gene expression profiles of 16 soft tissue cell lines from GDSC[39]. Among genes within the NEFM subnetwork, we found that FREM2 expression was significantly correlated with Paclitaxel sensitivity (Figure 3e). Paclitaxel is a microtubule-stabilizing drug that has been used to treat Kaposi's sarcoma, lung, ovarian, and breast cancer[40]. We clustered these 16 cell lines into two groups according to the gene expression values of the NEFM-subnetwork. We observed that these two clusters had significantly different Paclitaxel sensitivities (indicated by the area under the dose-response curve (P-value < 0.0019, t-test)) (Figure 3f). Notably, clustering on any of these five genes alone did not yield clusters with significantly different Paclitaxel sensitivities. When using this subnetwork to cluster a larger collection of 990 cell lines across 25 tissue lineages, we also observed significantly different Paclitaxel sensitivities (P-value < 0.004). Additionally, gene expression of Neurofilament Medium (NEFM) itself was significantly overexpressed in the Paclitaxel sensitive group relative to the Paclitaxel resistant group (log2 fold change 7.93 vs 4.00 respectively, P-value < 3.12e-8, rank-sum test). Neurofilaments are known to modulate microtubule stability[41], and higher neurofilament levels have been observed to destabilize microtubule network[42], thus may increase sensitivity to Paclitaxel. The

significant differences in overall survival and chemosensitivity indicate that this previously unrecognized NEFM-subnetwork may be a biomarker for sarcoma treatment.

Set2Gaussian finds concise previously defined gene sets for GSEA

With the rapid development of sequencing technology, high-throughput experiments have been used to generate large numbers of gene sets or pathways. Given a newly generated gene set, the immediate question is its function. Gene set enrichment analysis (GSEA) is a method for finding gene sets that are significantly different between two biological states[43]. Additionally, GSEA is extensively used to find significantly enriched gene sets for a novel gene set, from a large collection of previously defined gene sets[44] which are obtained from gene set databases such as MSigDB[3]. While GSEA provides a list of possible enriched gene sets, it does not resolve or de-duplicate these gene sets. However, gene set databases are continually growing, inevitably leading to a substantial amount of redundant information in previously defined gene sets. The redundancy not only slows down the enrichment analysis but also masks the true signal. Hence, there would be great potential value to decrease the number of previously defined gene sets and keep only the most informative and representative sets. Notably, data-reduction techniques have been applied to other computational biology tasks and obtain promising results[45,46]. Thus, we investigated whether representations generated by Set2Gaussian can be used to effectively downsize existing previously defined gene sets. To filter existing gene sets, we first used Set2Gaussian to project all gene sets of NCI, MSigDB, and Reactome into the same low-dimensional space and then selected the ones that are close to a large number of genes (see Methods).

Our results are summarized in Figure 4 a,b,c. We found that using previously defined gene sets created by Set2Gaussian achieved the best performance for GSEA. Specifically, we found significant enrichment for 72.1% of queried gene sets, which was significantly larger than 68.5% of *all*, 50.5% of *Standard*, 16.4% of *Proportional*, and 15.1% of *Hitting*. A detailed comparison of Set2Gaussian with baselines is shown in Figure 4b,c, which demonstrates the number of significantly enriched gene sets for each cell line. For all cell lines, Set2Gaussian enriched for more gene sets in comparison to Standard. And for 60 among 69 cell lines, Set2Gaussian enriched for more gene sets in comparison to All. We observed significant improvement across a large range of K from 100 to 1000 with a step size of 10 (all P-values < 0.05) (Figure 4d). At $K > 700$ gene sets, the improvement remained significant but started to decrease; we expected that this was because a K that was too large might include redundant gene sets in this collection. The highest percent improvement came from K=200 on 69 cell lines (P-value < 7e-10, Wilcoxon signed-rank test). In addition to increasing the number of enriched gene sets, Set2Gaussian-filtered previously defined gene sets also substantially enhanced the enrichment analysis computational speed in comparison to all previously defined gene sets (almost 65-fold). We provide Set2Gaussian-filtered previously defined gene sets in Supplementary Table 1, which covers a variety of biological processes and molecular functions, such as cytokine receptor interaction, cell cycle, and olfactory signaling pathway.

Discussion

Tens of thousands of genes sets are discovered from high-throughput experimental screening assays, providing novel insights into the underlying biological processes and molecular functions. Set2Gaussian is developed to facilitate the usage of these gene sets by creating high-quality and compact gene set features that can be used as input for machine learning codes. Our analysis of 11,892 gene sets indicates that Set2Gaussian is able to accurately recover gene set members, thus substantially reducing noise in experimentally derived gene sets. An important output of Set2Gaussian is the Set2Gaussian-derived feature representation of these 11,892 gene sets, which is made available by us to facilitate future gene set-based analysis.

We further showed how Set2Gaussian can be applied to high-dimensional somatic mutation data. Set2Gaussian allows us to stratify tumors into biologically meaningful subtypes and obtain an unrecognized NEFM-subnetwork, whereas none of the compared methods is able to achieve. We validated the NEFM-subnetwork in an independent cohort and found that it could be used as the biomarker for Paclitaxel sensitivity. Finally, we used Set2Gaussian to enhance and accelerate the widely used GSEA, by reducing the redundancy in previously identified gene sets. Set2Gaussian reduces the number of previously identified gene sets from 11,892 to 200. These 200 most informative gene sets obtained better GSEA hits when we used them to analyze large-scale cell line perturbation experiments.

One limitation of Set2Gaussian is that it requires an input network of genes to embed gene sets. In this work, we used the protein-protein interaction networks from the STRING database. Although it limits the applications of Set2Gaussian, the network provides additional information to the functional interdependencies between genes and has been extensively used to embed genes. In fact, biological networks have been massively generated these days from high-throughput experimental techniques[11] and scientific papers[47]. Even when there is no existing network available, one can still create a network based on the co-occurrence or mutual exclusivity in the gene set collection[8]. In addition, the quality of embeddings obtained by Set2Gaussian might be affected by noise in hub genes. Hub genes, which are genes that have a large number of neighbors in the network, could lead to biased diffusion states. Consequently, gene sets that have hub genes tend to have large variances and have less accurate low-dimensional representations.

Methods

Set2Gaussian framework

Problem definition. Let $A \in \boldsymbol{R}^{n \times n}$ be the adjacency matrix of a given network G, where n is the number of genes. V denotes the set of all genes. Let $H = \{h_1, h_2, \ldots, h_m\}$ be m gene sets defined on G, where each set of genes $h_i = \left\{v_1, v_2, \ldots, v_{|h_i|}\right\}, \forall v_i \in V$. Set2Gaussian aims to find a low-dimensional multivariate Gaussian distribution $N(\mu_h, \Sigma_h)$ for each gene set h with mean $\mu_h \in R^d$ and covariance matrix $\Sigma_h \in R^{d \times d}$, where $d \ll n$.

Random walk with restart from a gene set

In order to define the objective function, we first need to characterize the network topology that we want to preserve in the low-dimensional space. Here, we use the random walk with restart (RWR) to capture the network topology. RWR captures fine-grain topological properties that lie beyond direct neighbors[12,13]. When there are missing and spurious genes in a given gene set, RWR can correct the noise using network neighbors. RWR differs from the conventional random walk in that it introduces a predefined probability of restarting at the initial gene after every iteration.

Formally, we first calculate a transition matrix $\boldsymbol{B}$, which represents the probability of a transition from gene i to gene j. B is defined as:

$$B_{ij} = \frac{A_{ij}}{\Sigma_{j'} A_{ij'}}.$$

To run RWR from gene i, we define S_i^t as an n -dimensional distribution vector in which each entry j contains the probability of gene j being visited from gene i after t steps. RWR from gene i with restart probability p_S is defined as:

$$S_i^{t+1} = \left(1 - p_S\right) S_i^t B + p_S u_i,$$

where u_i is an n-dimensional distribution vector with $u_i(i) = 1$ and $u_i(j) = 0, \forall j \neq i$. We can obtain the stationary distribution S_i^∞ of RWR at the fixed point of this iteration. Consistent with the previous work[12,14,21,23], we define the diffusion state $S_i = S_i^\infty$ of each gene i to be the stationary distribution of an RWR starting at each gene. Here, the restart probability controls the relative influence of global and local topological information in the diffusion, where a larger value places greater emphasis on the local structure.

To run RWR from gene set k, we define Q_k^t as an n-dimensional distribution vector in which each entry contains the probability of a gene being visited from gene set k after t steps. RWR from gene set k with restart probability p_Q is defined as:

$$Q_k^{t+1} = \left(1 - p_Q\right) Q_k^t B + p_Q o_k,$$

where o_k is an n-dimensional distribution vector with $o_k\left(v\right) = \frac{1}{|h_k|}, \forall v \in h_k$ and $o_k(v) = 0, \forall v \notin h_k$. We can obtain the stationary distribution Q_k^∞ of RWR at the fixed point of this iteration. we define the diffusion state $Q_k = Q_k^\infty$ of each gene set k to be the stationary distribution of an RWR starting at each gene in k uniformly. When genes in the set are rank-ordered by importance, we can adjust o_k according to the gene weights.

Notably, a gene set could have missing or spurious genes. RWR can account for the noisy gene sets using network neighbors to characterize the network topology. The restart probability reflects our uncertainty of this gene set, where a smaller value encourages the gene set to extend its members with network neighbors. Set2Gaussian uses the diffusion state $S_i(Q_k)$ to represent the topological information of gene i (gene set k) in the network. The jth entry $S_{ij}(Q_{kj})$ stores the probability that RWR starts at gene i (gene set k) and ends up at gene j in equilibrium.

Representing gene sets as multivariate Gaussian distributions

The diffusion states of each gene and each gene set are used to find the low-dimensional representation. Set2Gaussian embeds genes and gene sets in the same low-dimensional space, where each gene is represented as a single point and each gene set is represented as a multivariate Gaussian distribution parameterized by a mean vector and covariance matrix.

Set2Gaussian optimizes two criteria to find the low-dimensional representation: 1) genes with similar diffusion states should be close to each other in the low-dimensional space, and 2) genes in a given gene set in the network should have higher probabilities in the Gaussian distribution of that gene set. The first criterion preserves the distance between genes and has been widely used in conventional node embedding approaches. The second criterion is unique to Set2Gaussian, and groups genes in the same set together as a multivariate Gaussian distribution. Through the use of the second criteria, Set2Gaussian explicitly leverages the prior knowledge that genes in the same set are likely to have similarities and thus should be closely located in the low-dimensional space. Formally, let L_{gene} and L_{set} represent the loss function based on the above two criteria. The loss function can be defined as:

$$L := L_{gene} + L_{set}.$$

To preserve the gene distance (criteria 1), we define L_{gene} as:

$$L_{gene} := \Sigma_{i=1}^{n} D_{KL}\left(S_i \middle|\middle| \widehat{S_l}\right),$$

where D_{KL} is the Kullback-Leibler (KL) divergence[48] and $\widehat{S_{lj}}$ is defined as:

$$\widehat{S_{lj}} := \frac{exp\left\{x_i^T w_j\right\}}{\Sigma_{j'} exp\left\{x_i^T w_{j'}\right\}}.$$

Here, x_i is the representation of gene i in the low-dimensional space and w_j is the context feature describing the network topology of gene j. If x_i and w_j are close in direction and have a large inner product, then it is likely that j is frequently visited in the random walk restarting from gene i. We optimize over w and x for all genes, using KL divergence as the objective function. One key improvement of Set2Gaussian is that all gene sets are trained jointly and the representation of genes is shared across all gene sets.

Similar to previous work, we relax the constraint that the entries in $\widehat{S_l}$ sum to one by dropping the normalization factor in the above equation[21,23]. As a result, $\widehat{S_{lj}}$ can be simplified as:

$$log\ \widehat{S_{lj}} = x_i^T w_j .$$

This simplification substantially reduces the computational complexity while still achieving comparable performance[21,23]. Since $\widehat{S_l}$ is no longer an n-dimensional probability simplex, we use the sum of squared errors instead of KL divergence as the new objective function. Therefore, L_{gene} is defined as:

$$L_{gene} := \Sigma_{i=1}^{n} \Sigma_{j=1}^{n} \left(log\ S_{ij} - x_i^T w_j \right)^2 .$$

Next, to preserve the distance between genes and gene sets, we define L_{set} as:

$$L_{set} := \Sigma_{k=1}^{m} D_{KL}\left(Q_k \middle|\middle| \widehat{Q_k} \right),$$

where $\widehat{Q_{kj}}$ is defined as :

$$\widehat{Q_{kj}} := \frac{f_k(j)}{\Sigma_{j'} f_k(j')} .$$

f_k is the multivariate Gaussian probability density function and $f_k(j)$ is the probability density of gene j:

$$f_k(j) = \frac{exp\left(-\frac{1}{2}\left(x_j - \mu_k\right)^T \Sigma_k^{-1} \left(x_j - \mu_k\right) \right)}{\sqrt{(2\Pi)^l |\Sigma_k|}} .$$

Here, we can optimize over the mean vector μ_k and the covariance matrix Σ_k to obtain the multivariate Gaussian distribution of gene set k.

Same as the simplification in L_{gene}, we also drop the normalization factor in the above equation. As a result, $\widehat{Q_{kj}}$ is simplified as:

$$log\ \widehat{Q_{kj}} = -\frac{1}{2}\left(x_j - \mu_k\right)^T \Sigma_k^{-1} \left(x_j - \mu_k\right) .$$

Notably, $\widehat{Q_{kj}}$ can also be viewed as the Mahalanobis distance of gene j from the mean μ_k and covariance matrix Σ_k. The Mahalanobis distance can account for different variances in each direction and reduces to Euclidean distance when Σ is an identity matrix. While matrix factorization approaches, such as singular value decomposition (SVD), also calculate a

diagonal matrix Σ, Set2Gaussian improves on this by optimizing different Σ_k for each gene set k in order to model the uncertainty of each gene set differently.

We then use the sum of squared errors as the objective function:

$$L_{set} = \Sigma_{k=1}^{m} \Sigma_{j=1}^{n} \left(log Q_{kj} + \frac{1}{2} (x_j - \mu_k)^T \Sigma_k^{-1} (x_j - \mu_k) \right)^2 .$$

Summing up two parts, the new loss function of our model is defined:

$$L = \Sigma_{i=1}^{n} \Sigma_{j=1}^{n} \left(log\ S_{ij} - x_i^T w_j \right)^2 + \Sigma_{k=1}^{m} \Sigma_{j=1}^{n} \left(log Q_{kj} + \frac{1}{2} (x_j - \mu_k)^T \Sigma_k^{-1} (x_j - \mu_k) \right)^2 .$$

While the first term preserves gene distance in the network, the second term forces genes in the same set to form a multivariate Gaussian distribution. Therefore, these biologically meaningful gene sets are used as prior knowledge by Set2Gaussian to infer the embedding of genes. By contrast, other methods, such as average embedding, are unable to leverage this prior knowledge.

Low-rank approximation of the covariance matrix

Set2Gaussian has the following parameters: μ, Σ, x, and w. The parameters μ, x, and w can be directly estimated with gradient descent. By contrast, since Σ is the covariance matrix of a multivariate Gaussian distribution, we need to assure that it is positive semi-definite. To achieve this, let Λ_k be the precision matrix of the multivariate Gaussian distribution for gene set k:

$$\Lambda_k := \Sigma_k^{-1} .$$

Instead of directly estimating the covariance matrix Σ_k, we estimate the precision matrix Λ_k to avoid numerical problems that arise in matrix inversion. We define $C_k \in R^{d \times d}$ to force Λ_k to be positive semi-definite:

$$\Lambda_k = C_k^T C_k .$$

Since a matrix multiplied by its transpose is positive semi-definite, Λ_k is thus a positive semi-definite matrix. This further ensures that its inverse Σ_k is also a positive semi-definite matrix. Since there is no constraint on C_k, we can use gradient descent to estimate C_k. However, directly optimizing over C_k introduces a substantial memory complexity of $O(md^2)$, which counteracts a key benefit of using a low-dimensional representation. To address this problem, we propose to factorize C_k by using a low-rank approximation:

$$C_k = Y_k Z_k^T ,$$

where $Z_k \in R^{d \times e}$, $Y_k \in R^{d \times e}$, and $e \ll d$. In our experiment, we found that setting e to 3 is expressive enough to obtain a good performance. This reduces Set2Gaussian's parameters to μ, Z, Y, x, and w. We estimate these parameters using Adam to find a local optimum[49].

After finding the low-dimensional representation using Set2Gaussian, we can calculate the distance between genes and gene sets in this space. The distance $D^k_{gene}(i)$ between gene i and gene set k is calculated according to the probabilistic density function of the multivariate Gaussian distribution for gene set k:

$$D^k_{gene}\left(i\right) = \frac{exp\left(-\frac{1}{2}(x_i - \mu_k)^T {\Sigma_k}^{-1}(x_i - \mu_k)\right)}{\sqrt{(2\Pi)^l |\Sigma_k|}}.$$

Using this formulation, the distance between a gene and a gene set depends not only on the mean vector μ (the location of this gene set) but also on the covariance matrix Σ. To calculate the distance $D^k_{set}(j)$ between gene set k and gene set j, we take the average asymmetric Kullback-Leibler divergence according to their Gaussian distributions:

$$D^k_{set}(j) = D_{KL}(N(\mu_k, \Sigma_k)||N(\mu_j, \Sigma_j)) + D_{KL}(N(\mu_j, \Sigma_j)||N(\mu_k, \Sigma_k)),$$

where $D_{KL}(N(\mu_k, \Sigma_k)||N(\mu_j, \Sigma_j))$ is calculated as:

$$D_{KL}(N(\mu_k, \Sigma_k)||N(\mu_j, \Sigma_j)) = \frac{1}{2}\left[tr\left(\Sigma_j^{-1}\Sigma_k\right) + (\mu_j - \mu_k)^T \Sigma_j^{-1}(\mu_j - \mu_k) - L - log\frac{|\Sigma_k|}{|\Sigma_j|}\right].$$

Sources of gene sets and molecular networks

We considered three public gene set databases widely used in gene set and pathway analyses, The Reactome Knowledgebase (Reactome)[4], National Cancer Institute Pathway Interaction Database (NCI)[1] and The Molecular Signatures Database (MSigDB)[3]. Reactome is a manually curated pathway collection composed of classical intermediary metabolism, signaling transduction, transport, DNA replication, and other key cellular processes. The pathways are classified into a pathway hierarchy. We obtained the lowest level of Homo sapiens pathways from this pathway hierarchy, resulting in 1771 pathways. We obtained all 223 pathways from NCI, which is a database of human cellular signaling and regulatory pathways that might be relevant to cancer research and treatment. We ignored interactions in these pathways from Reactome and NCI and used them as gene sets in our analysis. MSigDB contains annotated gene sets from eight sources, including oncogenic signatures, immunologic signatures, computationally inferred gene sets, curated gene sets, positional gene sets, gene ontology gene sets, motif gene sets, and hallmark gene sets. All sources except gene ontology gene sets were included in our analysis, resulting in 11892 gene sets.

Protein-protein interaction networks from the STRING database[11] were used for the PPI network structure. STRING uses a Bayesian integration algorithm to integrate many

different types of evidence for protein-protein interactions, including literature curation, computationally predicted interactions, interactions transferred from model organisms by orthology, interactions computed from genomic features such as gene-gene fusion events, and interactions based on functional or co-expression similarity. We downloaded the STRING v10, which has 15,108 genes and 3,621,168 edges. Each edge is associated with a confidence score between 0 and 1 assigned by the Bayesian integration algorithm.

Source of tumor somatic mutation, drug sensitivity, and differentially expressed gene sets

The Genomics of Drug Sensitivity in Cancer (GDSC) compound screening dataset spanned 990 human cancer cell lines[39]. We downloaded the compound response data, and gene expression profiles, and corresponding tissue of origin of the 990 cancer cell lines. To examine the identified sarcoma subnetwork, we used all 16 soft tissue cell lines that were screened under the exposure to Paclitaxel.

We downloaded somatic mutation profiles of TCGA sarcoma tumors from the GDAC Firehose website (http://gdac.broadinstitute.org, February 11th 2016)[50]. In each tumor, a gene was classified as either wild type (0) or altered (1), where altered is any non-silent mutation. We excluded tumors with less than 10 mutations, leaving a total of 237 tumors with 10,618 mutations in 15,108 genes. The survival of patients with these tumors was downloaded from TCGA.

The LINCS project consists of gene expression profiles of human cell lines exposed to perturbations[51]. These perturbations included treatment with more than 20,000 unique compounds and 69 cell lines. For each cell line, we first randomly selected 200 perturbation experiments from LINCS level 4 signature which normalized across samples based on two or more characteristics. We then obtained a gene set from each experiment by filtering for genes with an absolute level 4 signature scores greater than 2, resulting in 13,800 gene sets across 69 cell lines.

Gene set member identification

Intuitively, when projecting gene sets and genes into the same low-dimensional space, a gene set should be closer to its member genes than others in the low-dimensional space. As a result, we anticipate that a good gene set representation can accurately identify gene set members according to the distance between a gene set and a candidate gene in this space. We are not aware of any other methods for learning compact gene set representations; thus, for comparison, we proposed four competitive approaches. These approaches were derived according to the existing literature of aggregating set information in other research areas[26,27,52]. Pooling methods relied on a two-step optimization approach. In the first stage, these methods used an existing network embedding approach to obtain the embedding vector of genes in the network. In the second stage, these methods used the mean (*Mean*), max (*Max*), or weighted mean (*Weighted mean*) of the embedding vectors of gene set members to represent that gene set. *Weighted mean* used the random walk diffusion score as the weight for each gene set member. Because all of these three pooling-based approaches used a single vector to represent a gene set, they could not model the uncertainty in each dimension. Moreover, such two-stage optimization approaches might lead to suboptimal

results. Hypergraph embedding (*Hypergraph*) jointly optimized the gene and gene set embedding in a newly constructed hypergraph. In this method, each gene set was added as a new node to the original network that was connected to its member genes. Network embedding was applied to this heterogeneous network to embed gene set nodes and gene nodes into the same low-dimensional space. Similar to these methods, our approach took protein-protein interaction networks as input and generated embedding representations for all gene sets and genes. We evaluated Set2Gaussian on three gene set collections (NCI, Reactome, and MSigDB), each was further grouped into three categories according to the number of genes in the gene set (small [3–10], medium [11–30], and large [31–1000]). We chose the cutoffs that enabled the numbers of gene sets in each category close to each other. We used the Area Under Precision-Recall Curve (AURPC) as the evaluation metric which was shown to be a robust metric on different sparsity levels [53].

Mashup, a recently developed node embedding algorithm, was the underlying node embedding method for all comparison approaches[21]. Similar to Mashup's approach, we used cosine similarity to calculate the proximity between a gene set and a gene in the low-dimensional space. We set the dimension of Mashup as 500 which achieved the best performance in a range of dimensions from 100 to 1000 in our experiments. For our method, we set e to 3, d to 300 and p_Q to 0.8. We observed that the performance of our method is stable for e between 1 and 5, d greater than 100, and p_Q greater than 0.5.

Tumor stratification and subnetwork identification

Computationally, tumor stratification is performed by assessing the similarity between molecular profiles such as somatic mutation profiles. However, stratifying tumors based on somatic mutation is challenging due to the extreme sparsity and heterogeneity in somatic mutation profiles[54]. To address this sparsity and heterogeneity, a large number of network-based approaches have been proposed to aggregate gene mutations into higher-level functions[15,22,55].

Here, we asked whether Set2Gaussian could enhance tumor stratification accuracy and identify the underlying driver subnetworks. Formally, we modeled each tumor as a gene set denoted by the tumor's set of mutations. We then used Set2Gaussian to project these gene sets into Gaussian distributions located in the same low-dimensional space. These low-dimensional Gaussian distributions were later used as features to divide tumors into subtypes. By forcing each tumor's mutation set to form a Gaussian distribution, Set2Gaussian might be less biased by noise from networks and passenger mutations; and therefore, identify clinically meaningful subtypes. We compared Set2Gaussian with three comparison tumor stratification approaches. Network-based stratification (NBS) used protein-protein interaction networks to aggregate mutation signals within network regions[22]. *Mutation Profile* directly clustered tumors based on the somatic mutation profile without considering the network structure[56]. *Mutation Load* clustered tumors according to the number of mutations in each tumor[57]. We applied Set2Gaussian and the three comparison methods to stratify 235 Sarcoma tumors from TCGA. We used the Protein-protein interaction networks from the STRING database to identify the submodule for Sarcoma.

Since the co-expression values between genes in the STRING network are measured based on another cohort, the same co-expression may not be observed on GDSC cell lines.

For tumor stratification, we obtained the implementation of NBS from a recent paper[8]. To cluster tumors, we adopted the same clustering approach as described in Wang et al.[8] We first projected the representations of all tumors into a low dimensional space using truncated SVD. A tumor similarity matrix was then constructed by calculating the cosine similarity between the columns of truncated SVD. We adopted the *k*-means++ clustering algorithm to cluster tumors using the cosine tumor similarity matrix[58]. For K-means, the maximum number of iterations was set to 100 and the number of random starts was set to 200. Predefining the correct number of subtypes in a cancer cohort is difficult. Therefore, we compared a variety of subtypes from 2 to 6.

Identification of concise previously defined gene sets

Several computational approaches focused on either finding a subset of existing previously defined gene sets[59] or dynamically selecting gene sets according to the queried gene set[60,61]. The key insight behind these approaches is to find gene sets that cover a large number of genes and reduce redundancy within selected gene sets. Therefore, greedy algorithms were used to select gene sets sequentially, leading to heuristic and sensitive results. For example, Stoney et al. proposed three set cover approaches based on different objectives: removal of pathway redundancy, controlling pathway size, and coverage of the gene set[59]. Intuitively, these informative gene sets can be effectively derived from the Gaussian distributions of Set2Gaussian, because these Gaussian distributions inherently capture the redundancy, size, and coverage of gene sets.

To generate such previously defined gene sets, we first used Set2Gaussian to project all gene sets of NCI, MSigDB, and Reactome into the same low-dimensional space. We then defined a *gene set informative score* for each gene set according to its distance to all existing genes. A gene set will have a higher informative score if it is close to a large number of genes. The intuition is that informative gene sets are surrounded by many genes and located in the dense region in the low-dimensional space. The top K gene sets with the highest informative scores then became the Set2Gaussian-filtered concise previously defined gene sets. We set K to 200 and observed a stable performance of K from 100 to 1000, as demonstrated by our experiments.

To evaluate our approach, we ran GSEA for a given new gene set by comparing it with previously defined gene sets. We compared five different previously defined gene sets: Set2Gaussian-derived previously defined gene sets (Set2Gaussian, 200 gene sets), standard set cover-derived previously defined gene sets (Standard, 291 gene sets)[59], proportional set cover-derived previously defined gene sets (Proportional, 890 gene sets)[59], hitting set cover-derived previously defined gene sets (Hitting, 711 gene sets)[59], and all previously defined gene sets by combining sets from NCI, Reactome and MSigDB (All, 13868 gene sets). previously defined gene sets of Standard set cover, proportional set cover, and hitting set cover are obtained from Stoney et al.[59] which selected gene sets based on different objectives: removal of pathway redundancy, controlling pathway size and coverage of the gene set. Although a larger number of previously defined gene sets leads to a better chance

to find a significant enrichment, it is also more likely to commit a Type I error. Therefore, good previously defined gene sets should be able to find significant enrichment for more queried gene sets after correcting for Type I error using Bonferroni correction. We generated queried gene sets by mining large-scale gene expression data. In particular, we collected 69 cell lines from a large gene expression compendium LINCS. For each cell line, we used differential expression analysis to identify 200 differentially expressed gene sets grouped by drug exposure across 41,729 small molecules. We then ran GSEA on these 13800 gene sets by comparing to the above five previously defined gene sets using Fisher's exact test.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgments

This work is supported by NIH TR002515, GM102365, LM005652 and the Chan-Zuckerberg Biohub.

Reference

1. Schaefer CF et al. PID: the Pathway Interaction Database. Nucleic Acids Res. 37, D674–9 (2009). [PubMed: 18832364]
2. Hewett M. PharmGKB: the Pharmacogenetics Knowledge Base. Nucleic Acids Res. 30, 163–165 (2002). [PubMed: 11752281]
3. Liberzon A et al. Molecular signatures database (MSigDB) 3.0. Bioinformatics 27, 1739–1740 (2011). [PubMed: 21546393]
4. Croft D et al. The Reactome pathway knowledgebase. Nucleic Acids Res. 42, D472–7 (2014). [PubMed: 24243840]
5. Holden M, Deng S, Wojnowski L & Kulle B. GSEA-SNP: applying gene set enrichment analysis to SNP data from genome-wide association studies. Bioinformatics 24, 2784–2785 (2008). [PubMed: 18854360]
6. Wang S et al. Deep functional synthesis: a machine learning approach to gene functional enrichment. bioRxiv 824086 (2019) doi:10.1101/824086.
7. Wang S et al. Identification of pathways associated with chemosensitivity through network embedding. PLoS Comput. Biol. 15, e1006864 (2019).
8. Wang S et al. Typing tumors using pathways selected by somatic evolution. Nat. Commun. 9, 4159 (2018). [PubMed: 30297789]
9. Bateman AR, El-Hachem N, Beck AH, Aerts HJWL & Haibe-Kains B. Importance of collection in gene set enrichment analysis of drug response in cancer cell lines. Sci. Rep. 4, 4092 (2014). [PubMed: 24522610]
10. Menche J et al. Disease networks. Uncovering disease-disease relationships through the incomplete interactome. Science 347, 1257601 (2015).
11. Szklarczyk D et al. STRING v10: protein–protein interaction networks, integrated over the tree of life. Nucleic Acids Res. 43, D447–D452 (2015). [PubMed: 25352553]
12. Cao M et al. New directions for diffusion-based network prediction of protein function: incorporating pathways with confidence. Bioinformatics 30, i219–27 (2014). [PubMed: 24931987]
13. Navlakha S & Kingsford C. The power of protein interaction networks for associating genes with diseases. Bioinformatics 26, 1057–1063 (2010). [PubMed: 20185403]
14. Cao M et al. Going the distance for protein function prediction: a new distance metric for protein interaction networks. PLoS One 8, e76339 (2013).
15. Cowen L, Ideker T, Raphael BJ & Sharan R. Network propagation: a universal amplifier of genetic associations. Nat. Rev. Genet. 18, 551–562 (2017). [PubMed: 28607512]

16. Patkar S, Magen A, Sharan R & Hannenhalli S. A network diffusion approach to inferring sample-specific function reveals functional changes associated with breast cancer. PLoS Comput. Biol. 13, e1005793 (2017).
17. Leiserson MDM et al. Pan-cancer network analysis identifies combinations of rare somatic mutations across pathways and protein complexes. Nat. Genet. 47, 106–114 (2015). [PubMed: 25501392]
18. Kim Y-A, Wuchty S & Przytycka TM Identifying causal genes and dysregulated pathways in complex diseases. PLoS Comput. Biol. 7, e1001095 (2011).
19. Liu Y, Gu Q, Hou JP, Han J & Ma J. A network-assisted co-clustering algorithm to discover cancer subtypes based on gene expression. BMC Bioinformatics 15, 37 (2014). [PubMed: 24491042]
20. Wang B et al. Similarity network fusion for aggregating data types on a genomic scale. Nat. Methods 11, 333–337 (2014). [PubMed: 24464287]
21. Cho H, Berger B & Peng J. Compact Integration of Multi-Network Topology for Functional Analysis of Genes. Cell Syst 3, 540–548.e5 (2016). [PubMed: 27889536]
22. Hofree M, Shen JP, Carter H, Gross A & Ideker T. Network-based stratification of tumor mutations. Nat. Methods 10, 1108–1115 (2013). [PubMed: 24037242]
23. Wang S, Cho H, Zhai C, Berger B & Peng J. Exploiting ontology graph for predicting sparsely annotated gene function. Bioinformatics 31, i357–64 (2015). [PubMed: 26072504]
24. Zitnik M, Agrawal M & Leskovec J. Modeling polypharmacy side effects with graph convolutional networks. Bioinformatics 34, i457–i466 (2018). [PubMed: 29949996]
25. Wieting J, Bansal M, Gimpel K & Livescu K. Towards Universal Paraphrastic Sentence Embeddings. arXiv [cs.CL] (2015).
26. Krizhevsky A, Sutskever I & Hinton GE ImageNet classification with deep convolutional neural networks. Communications of the ACM vol. 60 84–90 (2017).
27. Xu K, Hu W, Leskovec J & Jegelka S. How Powerful are Graph Neural Networks? arXiv [cs.LG] (2018).
28. Cavallari S, Zheng VW, Cai H, Chang KC-C & Cambria E Learning Community Embedding with Community Detection and Node Embedding on Graphs. in Proceedings of the 2017 ACM on Conference on Information and Knowledge Management - CIKM '17 (2017). doi:10.1145/3132847.3132925.
29. Zhang J, Kwong S, Liu G, Lin Q & Wong K-C PathEmb: Random Walk based Document Embedding for Global Pathway Similarity Search. IEEE J Biomed Health Inform (2018) doi:10.1109/JBHI.2018.2830806.
30. Ashburner M et al. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. Nat. Genet. 25, 25–29 (2000). [PubMed: 10802651]
31. Bojchevski A & Günnemann S. Deep Gaussian Embedding of Graphs: Unsupervised Inductive Learning via Ranking. arXiv [stat.ML] (2017).
32. He S, Liu K, Ji G & Zhao J Learning to Represent Knowledge Graphs with Gaussian Embedding. in Proceedings of the 24th ACM International on Conference on Information and Knowledge Management - CIKM '15 (2015). doi:10.1145/2806416.2806502.
33. Dos Santos L, Piwowarski B & Gallinari P Multilabel Classification on Heterogeneous Graphs with Gaussian Embeddings. in Lecture Notes in Computer Science 606–622 (2016).
34. Fröhlich H, Fellmann M, Sültmann H, Poustka A & Beissbarth T. Predicting pathway membership via domain signatures. Bioinformatics 24, 2137–2142 (2008). [PubMed: 18676972]
35. Kim K, Jiang K, Teng SL, Feldman LJ & Huang H. Using biologically interrelated experiments to identify pathway genes in Arabidopsis. Bioinformatics 28, 815–822 (2012). [PubMed: 22271267]
36. García-Jiménez B, Pons T, Sanchis A & Valencia A. Predicting Protein Relationships to Human Pathways through a Relational Learning Approach Based on Simple Sequence Features. *IEEE/ACM Trans*. Comput. Biol. Bioinform. 11, 753–765 (2014).
37. Chavarría-Smith J & Vance RE The NLRP1 inflammasomes. Immunological Reviews vol. 265 22–34 (2015). [PubMed: 25879281]

38. Faustin B et al. Mechanism of Bcl-2 and Bcl-XL inhibition of NLRP1 inflammasome: Loop domain-dependent suppression of ATP binding and oligomerization. Proc. Natl. Acad. Sci. U. S. A. 106, 3935–3940 (2009). [PubMed: 19223583]
39. Iorio F et al. A Landscape of Pharmacogenomic Interactions in Cancer. Cell 166, 740–754 (2016). [PubMed: 27397505]
40. Saville MW et al. Treatment of HIV-associated Kaposi's sarcoma with paclitaxel. Lancet 346, 26–28 (1995). [PubMed: 7603142]
41. Millecamps S & Julien J-P Axonal transport deficits and neurodegenerative diseases. Nat. Rev. Neurosci. 14, 161–176 (2013). [PubMed: 23361386]
42. Yadav P et al. Neurofilament depletion improves microtubule dynamics via modulation of Stat3/stathmin signaling. Acta Neuropathol. 132, 93–110 (2016). [PubMed: 27021905]
43. Subramanian A et al. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. Proc. Natl. Acad. Sci. U. S. A. 102, 15545–15550 (2005). [PubMed: 16199517]
44. Huang DW, Sherman BT & Lempicki RA Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists. Nucleic Acids Res. 37, 1–13 (2009). [PubMed: 19033363]
45. Hie B, Cho H, DeMeo B, Bryson B & Berger B. Geometric Sketching Compactly Summarizes the Single-Cell Transcriptomic Landscape. Cell Syst 8, 483–493.e7 (2019). [PubMed: 31176620]
46. Cho H, Berger B & Peng J. Generalizable and Scalable Visualization of Single-Cell Data Using Neural Networks. Cell Syst 7, 185–191.e4 (2018). [PubMed: 29936184]
47. Poon H, Quirk C, DeZiel C & Heckerman D. Literome: PubMed-scale genomic knowledge base in the cloud. Bioinformatics 30, 2840–2842 (2014). [PubMed: 24939151]
48. Kullback S & Leibler RA On Information and Sufficiency. Ann. Math. Stat. 22, 79–86 (1951).
49. Kingma DP & Ba J. Adam: A Method for Stochastic Optimization. arXiv [cs.LG] (2014).
50. Cancer Genome Atlas Research Network et al. The Cancer Genome Atlas Pan-Cancer analysis project. Nat. Genet. 45, 1113–1120 (2013). [PubMed: 24071849]
51. Subramanian A et al. A Next Generation Connectivity Map: L1000 Platform and the First 1,000,000 Profiles. Cell 171, 1437–1452.e17 (2017).
52. Arora S, Liang Y & Ma T A Simple but Tough-to-Beat Baseline for Sentence Embeddings. (2016).
53. Davis J & Goadrich M The Relationship Between Precision-Recall and ROC Curves. in Proceedings of the 23rd International Conference on Machine Learning 233–240 (ACM, 2006).
54. Lawrence MS et al. Mutational heterogeneity in cancer and the search for new cancer-associated genes. Nature 499, 214–218 (2013). [PubMed: 23770567]
55. Cho A et al. MUFFINN: cancer gene discovery via network analysis of somatic mutation data. Genome Biol. 17, 129 (2016). [PubMed: 27333808]
56. Kim S, Sael L & Yu H. A mutation profile for top-k patient search exploiting Gene-Ontology and orthogonal non-negative matrix factorization. Bioinformatics 32, 2081 (2016). [PubMed: 27153726]
57. Samstein RM et al. Tumor mutational load predicts survival after immunotherapy across multiple cancer types. Nat. Genet. 51, 202–206 (2019). [PubMed: 30643254]
58. Arthur D & Vassilvitskii S k-means++: The advantages of careful seeding. in Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms 1027–1035 (Society for Industrial and Applied Mathematics, 2007).
59. Stoney RA, Schwartz J-M, Robertson DL & Nenadic G. Using set theory to reduce redundancy in pathway sets. BMC Bioinformatics 19, 386 (2018). [PubMed: 30340461]
60. Simillion C, Liechti R, Lischer HEL, Ioannidis V & Bruggmann R. Avoiding the pitfalls of gene set enrichment analysis with SetRank. BMC Bioinformatics 18, 151 (2017). [PubMed: 28259142]
61. Lu Y, Rosenfeld R, Simon I, Nau GJ & Bar-Joseph Z. A probabilistic generative model for GO enrichment analysis. Nucleic Acids Res. 36, e109 (2008).

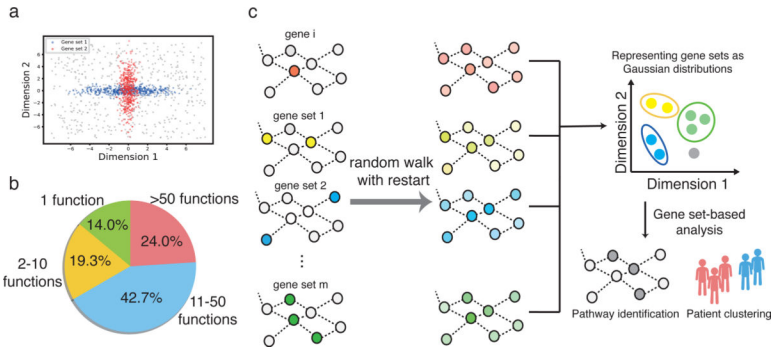

Figure 1.

Overview of Set2Gaussian. (a). 2-D t-SNE plot showing two very different gene sets embedded in the same point (0,0) by simply averaging embeddings of individual genes. (b). Pie chart showing the percent of 150 drug response correlated gene sets that are significantly enriched with different numbers of Gene Ontology functions (P-value < 0.05 after Bonferroni correction). (c). Flowchart showing Set2Gaussian embedding process and downstream applications. First, RWR is used to calculate the diffusion states of each gene and gene set. These diffusion states are then embedded into a low-dimensional space where genes are represented as single points and gene sets are represented as Gaussian distributions. These representations are then applied to a variety of gene set-based analyses.

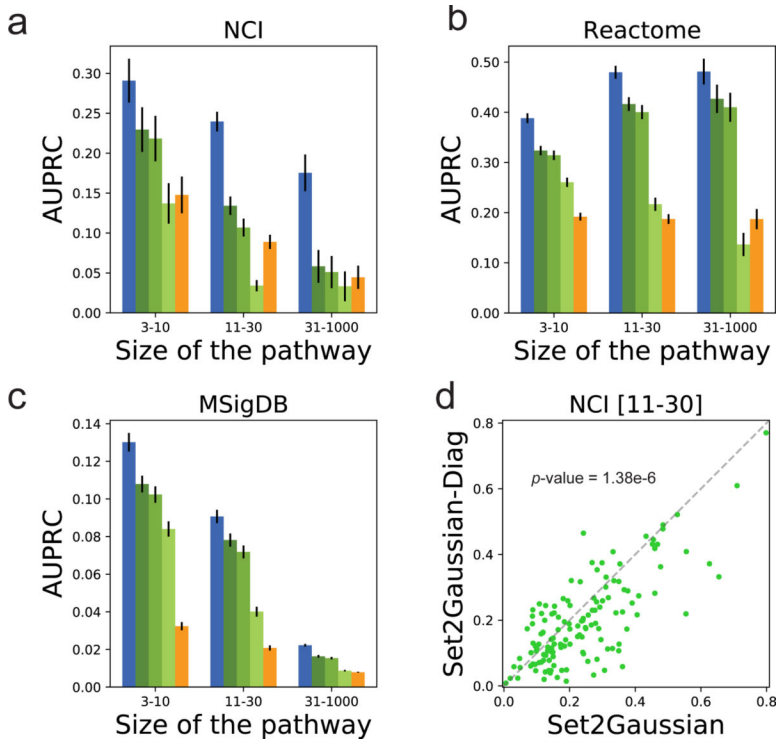

Figure 2.

Application of Set2Gaussian to gene set member identification. Comparison of Set2Gaussian with four other approaches in identifying gene set members in NCI (a), Reactome (b), and MSigDB (c). Gene sets are grouped into three categories according to the number of genes in the gene set (small [3–10], medium [11–30], large [31–1000]). (d). Comparison of Set2Gaussian with Set2Gaussian-diag on NCI for medium gene sets ([11–30]). Error bars represent the standard error on the mean.

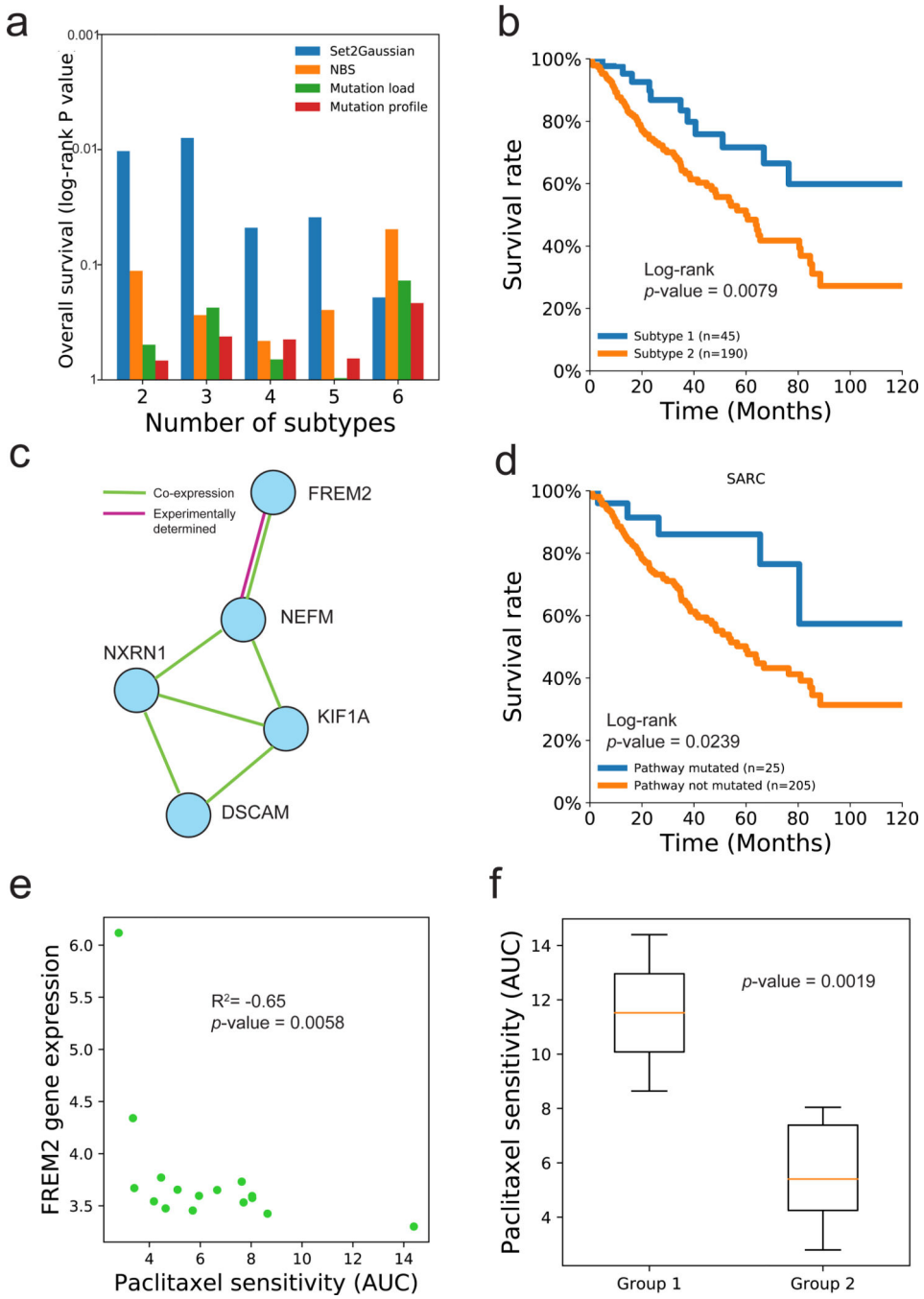

Figure 3.

Application of Set2Gaussian to cancer subtyping. (a). Comparison of Set2Gaussian with other approaches in tumor stratification on 235 sarcoma tumors at varying numbers of subtypes. (b). KM-plot showing the clustering of tumors into 2 subtypes using all the genes. (c). NEFM-subnetwork identified by Set2Gaussian in sarcoma. (d). KM-plot showing the clustering of tumors into 2 subtypes by only using the NEFM-subnetwork. (e). Scatter plot showing the comparison between the expression values of FREM2 and the drug responses to Paclitaxel on 16 soft-tissue cell lines collected from CTRP. (f). Box plot showing the

Paclitaxel sensitivity of two groups of cell lines clustered by using the expression of NEFM-subnetwork.

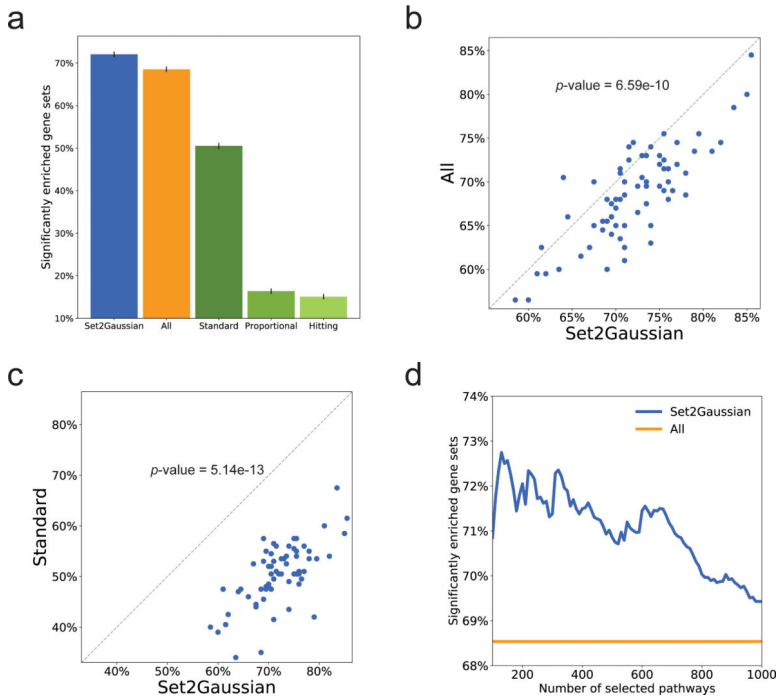

Figure 4.

Application of Set2Gaussian to finding concise gene sets. (a). Comparison of using five different previously defined gene sets in gene set enrichment analysis. (b). Comparison of using Set2Gaussian-filtered previously defined gene set to using all previously defined gene sets. Each point is a cell line, where x-axis (y-axis) shows how many gene sets of this cell line can find at least one significant enriched gene set from Set2Gaussian (all) previously defined gene sets. (c). Comparison of using Set2Gaussian-filtered previously defined gene set to using standard cover set-derived previously defined gene sets. (d). The number of significantly enriched gene sets at varying number of previously defined gene sets selected by Set2Gaussian. Error bars represent the standard error on the mean.