Gene sets, including protein complexes and signalling 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 tumours, and finds concise gene sets for gene set enrichment analysis. We further show how Set2Gaussian allows us to identify a clinical prognostic and predictive subnetwork around neurofilament medium in sarcoma, which we validate in independent cohorts.

One of the most basic outcomes of biological data analysis is the discovery of gene sets. Such sets come from many sources\(^1\). Genome-wide association studies (GWASs) 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 metabolites that interact with one another into network neighbourhoods. In all of these cases, the hypothesis is that genes in the set are probably involved in the same biological process or function. Because of their ability to boost the signal-to-noise ratio and increase explanatory power, gene sets have been used in various downstream analyses, including disease signature identification\(^2\), drug pathway association prediction\(^3\), survival analysis\(^4\) and drug response prediction\(^5\). Unfortunately, a bottleneck for gene set-based analysis is the lack of computational tools that convert these gene sets into informative forms that can distinguish completely different gene sets deemed as the same by averaging embedding (Fig. 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 of high quality to avoid overfitting.

Molecular interaction networks provide novel insights into the functional interdependencies of genes and proteins\(^6\). 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\(^7\). Consequently, network-based approaches, including network propagation\(^8\)–\(^11\), network clustering\(^12\), network integration\(^13\)–\(^15\) and network regularization\(^16\), have been developed to efficiently analyse these networks. Among these, 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\(^17\)–\(^19\). Molecular interaction networks are noisy and incomplete, especially as they increase in size\(^18\)–\(^20\). Network embedding typically leverages dimensionality-reduction techniques to regularize high-dimensional network data. Before 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 Article, 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\(^2\) or drug nodes\(^2\).

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 (for example, the number of nodes in the network). Although many useful gene embeddings are now available\(^2\),\(^2\),\(^2\), learning representations for gene sets remains challenging. One simple approach to represent a gene set is to average 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 modelling, graph neural network models also use pooling to aggregate information from the neighbourhood 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. In 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 inaccuracies in the embedding space. Other methods aim at embedding densely connected subnetworks. For example, ComE performs community embedding and community detection simultaneously, using a community-aware high-order proximity\(^2\), while PathEmb models pathways as documents and then applies document embedding models to calculate pathway similarity\(^2\). 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 (Fig. 1b)\(^3\). We found that 86% of these gene sets were significantly enriched with more than one GO function (\(P < 0.05\) after Bonferroni correction). This indicates that genes in the same set frequently had different functions and were involved in multiple biological processes, but 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 (Fig. 1c), has been proposed to model the uncertainty of nodes\(^3\). Motivated by previous 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.

We thus present Set2Gaussian, a novel computational method that summarizes genes in the same set closely as a multivariate Gaussian distribution in the low-dimensional space (Fig. 1c). Set2Gaussian takes biological networks and a collection of gene sets as input. Each gene is represented as a single point in the low-dimensional space. Each gene set is represented 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 as in previous work\(^4\), we propose to use a low-rank matrix 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 allow us to identify a clinical prognostic and predictive subnetwork around neurofilament medium (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 (GSEA).

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 that summarizes both the location and uncertainty of each dimension, improving on the expressive power of existing node-embedding models. Set2Gaussian takes a network and a collection of gene sets as input (Fig. 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 that is parameterized by a mean vector and covariance matrix. In our work, we approximate the covariance matrix through low-rank matrix factorization 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 analysed 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.
as a result of 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 labelling manually is expensive and tedious, many computational approaches have been proposed to identify gene set members, each utilizing different information, including domain signatures, gene expression, and sequence data.

Set2Gaussian significantly outperformed the proposed baseline gene set representation approaches on identifying gene set members in all three datasets at all size categories ($P < 0.05$; Wilcoxon signed-rank test) (Fig. 2a–c). We first observed clear improvements over three pooling-based approaches in all three datasets. Network embedding approaches rely on finding similar contexts (for example, similar neighbours or similar diffusion states) to accurately embed nodes. However, in a hypergraph, a ‘gene set node’ could have a noisy neighbourhood structure due to a large number of neighbours, 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. Set2Gaussian showed improved performance over Set2Gaussian-diag (AUPRC of 0.24 versus 0.20, respectively) for medium sets (11–30) in Reactome (Fig. 2d). Moreover, the improvement of Set2Gaussian over Set2Gaussian-diag was larger for the MSigDB dataset than for the other two datasets (Supplementary Fig. 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 modelling 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 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 members of this pathway. These predictions were supported by the literature. NLRP1 activates a downstream protease caspase-1 (CASP1), while BAX, BCL2A1 and MCL1 belong to the BCL2 family, which inhibits NLRP1-induced CASP1 activation in a concentration-dependent manner.

Set2Gaussian identifies novel subtypes and subnetworks in sarcoma. We next applied Set2Gaussian to tumour stratification and subnetwork identification in sarcoma. Tumour stratification aims to divide a heterogeneous population of tumours into clinically and biologically meaningful subtypes. We modelled each tumour as a gene set denoted by the tumour’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, whereas subtypes from the other three approaches did not (Fig. 3a). For example, for $k = 2$ subtypes, the survival times were significantly different between subtypes ($P < 0.03$, log-rank test), and five-year survival rates varied greatly (41% and 71%, respectively) (Fig. 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 tumours.

By finding the top differentially mutated genes between two subtypes that have the highest and lowest survival rates, we further
Because this subnetwork seemed to be associated with sarcoma subtype, we first collected gene expression profiles of 16 soft tissue cell lines from the Genomics of Drug Sensitivity in Cancer (GDSC) database. Among genes within the NEFM-subnetwork, we found that FREM2 expression was significantly correlated with paclitaxel sensitivity (Fig. 3c). Paclitaxel is a microtubule-stabilizing drug that has been used to treat Kaposi’s sarcoma, lung, ovarian and breast cancer. 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 < 0.0019, t-test); Fig. 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 < 0.004). Additionally, gene expression of NEFM itself was significantly over-expressed in the paclitaxel-sensitive group relative to the paclitaxel-resistant group (log2(fold change) of 7.93 versus 4.00, respectively, P < 3.12×10^{-3}, rank-sum test). Neurofilaments are known to modulate microtubule stability, and higher neurofilament levels have been observed to destabilize the microtubule network and 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. GSEA is a method for finding gene sets that are significantly different between two biological states. 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 obtained from gene set databases such as MSigDB. Although 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 have achieved promising results. 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 the NCI, MSigDB and Reactome datasets into the same low-dimensional space and then selected the ones that were close to a large number of genes (see Methods).

Our results are summarized in Fig. 4a–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% for All, 50.5% for Standard, 16.4% for Proportional and 15.1% for Hitting. A detailed comparison of Set2Gaussian with baselines is shown in Fig. 4b, which demonstrates the number of significantly enriched gene sets for a novel gene set, from a large collection 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 have achieved promising results. 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 the NCI, MSigDB and Reactome datasets into the same low-dimensional space and then selected the ones that were close to a large number of genes (see Methods).
sets in this collection. The highest percent improvement came from K=200 on 69 cell lines (P<7×10−10, 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). The Set2Gaussian-filtered previously defined gene sets are provided in Supplementary Table 1, which covers a variety of biological processes and molecular functions, such as cytokine receptor interaction, cell cycle and olfactory signalling pathways.

**Discussion**

Tens of thousands of gene sets are discovered in high-throughput experimental screening assays, providing novel insights into the underlying biological processes and molecular functions. Set2Gaussian has been 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 have shown how Set2Gaussian can be applied to high-dimensional somatic mutation data. Set2Gaussian allows us to stratify tumours into biologically meaningful subtypes and obtain an unrecognized NEFM-subnetwork, which 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 analyse 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 this limits the applications of Set2Gaussian, the network provides additional information regarding the functional interdependencies between genes and has been extensively used to embed genes. Biological networks are being generated on a huge scale via
high-throughput experimental techniques\textsuperscript{11} and scientific papers\textsuperscript{7}. 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\textsuperscript{4}. In addition, the quality of embeddings obtained by Set2Gaussian might be affected by noise in hub genes. Hub genes—genes that have a large number of neighbours 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 $A \in R^{n \times n}$ be the adjacency matrix of a given network $G$, where $n$ is the number of genes, $V$ denotes the set of all genes. Let $H = \{h_1, h_2, \ldots, h_n\}$ be $m$ gene sets defined on $G$, where each set of genes $h_i = \{v_1, v_2, \ldots, v_{n_i}\}; \forall v_i \in V$. Set2Gaussian aims to find a low-dimensional multivariate Gaussian distribution $N(\mu, \Sigma)$ for each gene set $h$ with mean $\mu_i \in R^d$ and covariance matrix $\Sigma_i \in R^{d \times d}$, where $d \leq n$.

**Random walk with restart from a gene set.** 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-grained topological properties that lie beyond direct neighbours\textsuperscript{13,14}. When there are missing and spurious genes in a given gene set, RWR can correct the noise using network neighbours. 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 $B$, which represents the probability of a transition from gene $i$ to gene $j$. $B$ is defined as

$$B_{ij} = \frac{A_{ij}}{\sum_j A_{ij}}$$

To run RWR from gene $i$, we define $S_i^{(t)}$, 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 $P_i$ is defined as

$$S_i^{(t+1)} = (1 - P_i)S_i^{(t)}B + P_i \mathbf{a}_i$$

where $\mathbf{a}_i$ is an $n$-dimensional distribution vector with $a_i(j) = 1$ and $a_i(j) = 0, \forall j \neq i$. We can obtain the stationary distribution $S_i^*$ of RWR at the fixed point of this iteration. Consistent with previous work\textsuperscript{13,14}, we define the diffusion state $S_i = S_i^*$ 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 $Q_k$, 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 $P_k$ is defined as

$$Q_k^{(t+1)} = (1 - P_k)Q_k^{(t)}B + P_k \mathbf{a}_k$$

where $\mathbf{a}_k$ is an $n$-dimensional distribution vector with $a_k(v) = 1, \forall v \in h_k$ and $a_k(v) = 0, \forall v \notin h_k$. We can obtain the stationary distribution $Q_k^*$ of RWR at the fixed point of this iteration. We define the diffusion state $Q_k = Q_k^*$ 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 ranked order by importance, we can adjust $\mathbf{a}_k$ 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 neighbours 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 neighbours. Set2Gaussian uses the diffusion state $S_i(\mathbf{Q}_k)$ to represent the topological information of gene $i$ (gene set $k$) in the network. The $j$th entry $S_i(j)$ 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 criterion, 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 $L_{\text{gene}}$ and $L_{\text{set}}$ represent the loss function based on the above two criteria. The loss function can be defined as

$$L := L_{\text{gene}} + L_{\text{set}}$$

To preserve the gene distance (criterion 1), we define $L_{\text{gene}}$ as

$$L_{\text{gene}} := \sum_{i=1}^{n} D_{KL}(\tilde{S}_i | S_i)$$

where $D_{KL}$ is the Kullback–Leibler (KL) divergence\textsuperscript{22} and $\tilde{S}_i$ is defined as

$$\tilde{S}_i := \frac{\exp\{x_i^T w_i\}}{\sum_j \exp\{x_j^T w_j\}}$$

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 $S$ sum to one by dropping the normalization factor in the above equation\textsuperscript{11,13}. As a result, $\tilde{S}_i$ can be simplified as

$$\log \tilde{S}_i = x_i^T w_j$$

This simplification substantially reduces the computational complexity while still achieving comparable performance\textsuperscript{13}. Because $S_i$ 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, $L_{\text{gene}}$ is defined as

$$L_{\text{gene}} := \sum_{i=1}^{n} \sum_{j=1}^{n} (\log S_{ij} - x_i^T w_j)^2$$

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

$$L_{\text{set}} := \sum_{k=1}^{m} D_{KL}(\tilde{Q}_k | Q_k)$$

where $Q_k$ is defined as

$$\tilde{Q}_k = \frac{f_k(j)}{\sum_j f_k(j)}$$

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

Notably, $Q_k$ can also be viewed as the Mahalanobis distance of gene $j$ from the mean vector $\mu$ and covariance matrix $\Sigma_k$. The Mahalanobis distance can account for different variances in each direction and reduces to Euclidean distance when $\Sigma$ is an identity matrix. Although matrix factorization approaches, such as singular value decomposition (SVD), also calculate a diagonal matrix $\Sigma$, Set2Gaussian improves on this by optimizing different $\Sigma_k$ for each gene set $k$ to model the uncertainty of each gene set differently.

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

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

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

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

Although 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: \( \mu, \Sigma, x \) and \( w \). Parameters \( \mu, x \) and \( w \) can be directly estimated with gradient descent. By contrast, because \( \Sigma \) is the covariance matrix of a multivariate Gaussian distribution, we need to ensure that it is positive semi-definite. To achieve this, let \( A_k \) be the precision matrix of the multivariate Gaussian distribution for gene set \( k \):

\[
A_k = \Sigma_k^{-1}
\]

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

\[
A_k = C_k^T C_k
\]

Because a matrix multiplied by its transpose is positive semi-definite, \( A_k \) is thus a positive semi-definite matrix. This further ensures that its inverse \( \Sigma_k \) is also a positive semi-definite matrix. Given that 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 \( O(n^2d) \), 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:

\[
C_k = Y_k D_k Z_k^T
\]

where \( Z_k \in R^{d \times r}, Y_k \in R^{r \times r} \) and \( r \ll d \). In our experiment, we found that setting \( r \) to 3 is expressive enough to obtain a good performance. This reduces Set2Gaussian’s parameters to \( \mu, Z_k, Y_k, x \) and \( w \). We estimate these parameters using Adam to find a local optimum.

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

\[
D^\text{gene}_{ij} = \exp \left( -\frac{1}{2}(x_i - \mu_j)^T \Sigma_j^{-1} (x_i - \mu_j) \right) \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma_j|^{\frac{1}{2}}}
\]

Using this formulation, the distance between a gene and a gene set depends not only on the mean vector \( \mu \) (the location of this gene set) but also on the covariance matrix \( \Sigma \). To calculate the distance \( D^\text{set}_{ij} \) between gene set \( k \) and gene set \( j \), we take the average asymmetric KL divergence according to their Gaussian distributions:

\[
D^\text{set}_{ij} = \text{KL}(N(\mu_i, \Sigma_i)||N(\mu_j, \Sigma_j)) + \text{KL}(N(\mu_j, \Sigma_j)||N(\mu_i, \Sigma_i))
\]

where

\[
\text{KL}(N(\mu_i, \Sigma_i)||N(\mu_j, \Sigma_j)) = \frac{1}{2} \left[ (\Sigma_j^{-1} \Sigma_i)^{-1} + (\mu_i - \mu_j)^T \Sigma_j^{-1} (\mu_i - \mu_j) - \log |\Sigma_j^{-1} \Sigma_i| - d \right]
\]

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), the National Cancer Institute Pathway Interaction Database (NCI) and the Molecular Signatures Database (MSigDB). Reactome is a manually curated pathway collection composed of classical intermediaries in metabolism, signalling 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 1,771 pathways. We obtained all 223 pathways from NCI, which is a database of human cellular signalling and regulatory pathways that might be relevant to cancer research and treatment. We ignored intersections 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, GO gene sets, motif gene sets and hallmark gene sets. For sources except GO gene sets were included in our analysis, resulting in 11,892 gene sets.

Protein–protein interaction networks from the STRING database 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 biochemical proximity, co-occurrence in co-expression data, physical 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 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 tumour somatic mutation, drug sensitivity and differentially expressed gene sets. The GDSC compound screening dataset spanned 990 human cancer cell lines. 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 time of 48 h.

We downloaded somatic mutation profiles of The Cancer Genome Atlas (TCGA) sarcoma tumours from the GDAC Firehose website. In each tumour, a gene was classified as either wild type (0) or altered (1), where altered is any non-silent mutation. We excluded tumours with fewer than 10 mutations, leaving a total of 238 tumours with 10,618 mutations in 15,108 genes. The survival of patients with these tumours was downloaded from TCGA.

The Library of Integrated Network-Based Cellular Signatures (LINCS) project consists of gene expression profiles of human cell lines exposed to perturbations. These perturbations included treatment with more than 20,000 unique compounds and drug and 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 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 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. As we are not aware of current methods for learning compact gene set representations, for comparison we proposed four competitive approaches. These approaches were derived according to the existing literature of aggregating set information in other research areas. 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, 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 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 (31–1,000)). We chose the cutoffs that enabled the numbers of gene sets in each category to be close to each other. We used the AUPRC as the evaluation metric, which was shown to be a robust metric on different sparsity levels.

Mashup, a recently developed node-embedding algorithm, was the underlying node-embedding method for all comparison approaches. 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 1,000 in our experiments. For our method, we set to 3, to 300 and to 0.8. We observed that the performance of our method is stable for between 1 and 5, greater than 100 and greater than 0.5.

Tumour stratification and subnetwork identification. Computationally, tumour stratification is performed by assessing the similarity between molecular profiles such as somatic mutation profiles. However, stratifying tumours based on somatic mutation profiles is a challenging task due to the high degree of heterogeneity and the sparsity of the mutation profiles. To address this sparsity and heterogeneity, a large number of network-based approaches have been proposed to aggregate gene mutations into higher-function genes.

Here, we asked whether Set2Gaussian could enhance tumour stratification accuracy and predict the underlying driver subnetworks. Specifically, we modelled each tumour as a gene set denoted by the tumour’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 tumours into subtypes. By forcing each tumour’s mutation set to form a Gaussian distribution, Set2Gaussian might be less biased by noise from networks and pathways. We compared this approach with two meaningful subtypes. We compared Set2Gaussian with three comparison tumour stratification approaches. Network-based stratification (NBS) used 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 (31–1,000)). We chose the cutoffs that enabled the numbers of gene sets in each category to be close to each other. We used the AUPRC as the evaluation metric, which was shown to be a robust metric on different sparsity levels.
interaction networks to aggregate mutation signals within network regions. Mutation Profile directly clustered tumours based on the somatic mutation profile without considering the network structure. Mutation Load clustered tumours according to the number of mutations in each tumour. We applied Set2Gaussian and the three comparison methods to stratify 235 sarcoma tumours from TCGA. We used the protein–protein interaction networks from the STRING database to identify the submodule for sarcoma. Because 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 tumour stratification, we obtained the implementation of NBS from a recent paper. To cluster tumours, we adopted the same clustering approach as described in Wang and others. We first projected the representations of all tumours into a low-dimensional space using truncated SVD. A tumour 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 tumours using the cosine tumour similarity matrix. 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. We thus compared a variety of subtypes from 2 to 6.

**Identification of concise previously defined gene sets.** Several computational approaches have focused on either finding a subset of existing previously defined gene sets or dynamically selecting gene sets according to the queried gene set. The key insight behind these approaches is to find gene sets that cover a large number of previously defined gene sets with minimal redundancy when selected gene sets. Therefore, greedy algorithms have been used to select gene sets sequentially, leading to heuristic and sensitive results. For example, Stoney and others proposed three set cover approaches based on different objectives: removal of pathway redundancy, controlling pathway size and coverage of the gene set. 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 gene sets. A gene set will have a higher informative score if it is close to a large number of gene sets. The intuition is that informative gene sets are surrounded by many gene sets and located in the dense region in the low-dimensional space. The top K gene sets with the highest informative scores then become the Set2Gaussian-filtered concise previously defined gene sets. We set K to 200 and observed a stable performance of K from 100 to 1,000, 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), proportional set cover-derived previously defined gene sets (Proportional, 890 gene sets), hitting set cover-derived previously defined gene sets (Hitting, 711 gene sets) and all previously defined gene sets by combining sets from NCI, Reactome and MSigDB (All, 13,868 gene sets). Previously defined gene sets of standard set cover, proportional set cover and hitting set cover were obtained from ref. 32, 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 mixing 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 13,800 gene sets by comparing to the above five previously defined gene sets using Fisher’s exact test.

**Reporting Summary.** Further information on research design is available in the Nature Research Reporting Summary linked to this Article.

**Data availability**

We provide pretrained 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 Set2Gaussian is is available at https://doi.org/10.5281/zenodo.3827929.

Received: 21 November 2019; Accepted: 15 May 2020; Published online: 15 June 2020

**References**

1. Schaefer, C. F. et al. PID: The Pathway Interaction Database. Nucleic Acids Res. 37, D674–D679 (2009).

2. Hewett, M. PharmGKB: the Pharmacogenomics Knowledge Base. Nucleic Acids Res. 30, 163–165 (2002).

3. Liberzon, A. et al. Molecular signatures database (MSigDB) 3.0. Bioinformatics 27, 1739–1740 (2011).

4. Croft, D. et al. The Reactome pathway knowledgebase. Nucleic Acids Res. 42, D472–D477 (2014).

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).

6. Wang, S. et al. Deep functional synthesis: a machine learning approach to gene functional enrichment. Preprint at https://doi.org/10.1101/624086 (2019).

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).

9. Bateman, A. R., El-Hachem, N., Beck, A. H., Aerts, H. J. W. L. & Haibe-Kains, B. Importance of collection in gene set enrichment analysis of drug response in cancer cell lines. Sci. Rep. 4, 4092 (2014).

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).

12. Cao, M. et al. New directions for diffusion-based network prediction of protein function: incorporating pathways with confidence. Bioinformatics 30, i219–i227 (2014).

13. Navlakha, S. & Kingsford, C. The power of protein interaction networks for associating genes with diseases. Bioinformatics 26, 1057–1063 (2010).

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, B. J. & Sharan, R. Network propagation: a universal amplifier of genetic associations. Nat. Rev. Genet. 18, 551–562 (2017).

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, M. D. M. et al. Pan-cancer network analysis identifies combinations of rare somatic mutations across pathways and protein complexes. Nat. Genet. 47, 106–115 (2015).

18. Kim, Y.-A., Wuchty, S. & Przytycka, T. M. Identifying causal genes and dysregulated pathways in complex diseases. PLoS Comput. Biol. 7, e1001095 (2011).

19. Liu, Y., Gu, Q., Hou, J. P., Han, J. & Ma, J. A network-assisted co-clustering algorithm to discover cancer subtypes based on gene expression. BMC Bioinformatics 15, 37 (2014).

20. Wang, B. et al. Similarity network fusion for aggregating data types on a genomic scale. Nat. Methods 11, 333–337 (2014).

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).

22. Hofree, M., Shen, J. P., Carter, H., Gross, A. & Ideker, T. Network-based stratification of tumor mutations. Nat. Methods 10, 1108–1113 (2013).

23. Wang, S., Cho, H., Zhai, C., Berger, B. & Peng, J. Exploiting ontology graph for predicting sparsely annotated gene function. Bioinformatics 31, i357–i364 (2015).

24. Zitnik, M., Agrawal, M. & Leskovec, J. Modeling polypharmacy side effects with graph convolutional networks. Bioinformatics 34, i457–i466 (2018).

25. Wieting, J., Bansal, M., Gimbel, K. & Livescu, K. Towards universal paraphrastic sentence embeddings. Preprint at https://arxiv.org/pdf/1511.08198.pdf (2015).

26. Krizhevsky, A., Sutskever, I. & Hinton, G. E. ImageNet classification with deep convolutional neural networks. Commun. ACM 60, 84–90 (2017).

27. Xu, K., Hu, W., Leskovec, J. & Jegelka, S. How powerful are graph neural networks? Preprint at https://arxiv.org/pdf/1810.00826.pdf (2018).

28. Cavallari, S., Zheng, W. V., Cai, H., Chang, K. C.-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 ‘17 377–386 (2017).

29. Zhang, J., Kwong, S., Liu, G., Lin, Q. & WongK.-C. PathEmb: random walk based document embedding for global pathway similarity search. IEEE J. Biomed. Health Inform 23, 1329–1335 (2018).

30. Ashburner, M. et al. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. Nat. Genet. 25, 25–29 (2000).

31. Bojchevski, A. & Günnemann, S. Deep Gaussian embedding of graphs: unsupervised inductive learning via ranking. Preprint at https://arxiv.org/pdf/1707.03815.pdf (2017).
Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Research policies, see our Editorial Policies and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

- n/a | Confirmed
- The exact sample size (n) for each experimental group/condition, given as a discrete number and unit of measurement
- A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
- The statistical test(s) used AND whether they are one- or two-sided
- Only common tests should be described solely by name; describe more complex techniques in the Methods section.
- A description of all covariates tested
- A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
- A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
- For null hypothesis testing, the test statistic (e.g. F, t, r) with confidence intervals, effect sizes, degrees of freedom and P value noted
- Give P values as exact values whenever possible.
- For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
- For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
- Estimates of effect sizes (e.g. Cohen's d, Pearson's r), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection
We use python2.7 to process and download gene sets from MSigDB, Reactome and NCI.

Data analysis
We used the scipy.stats Python package implementation of the one-sided independent t-test, Pearson correlation statistics, Spearman correlation statistics, and associated p-values used in this study. We used the scikit-learn Python package implementation of one-vs-rest logistic regression, silhouette coefficients, AUROC, and Cohen's kappa statistics used in this study.

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. Git-Hub). See the Nature Research guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:
- Accession codes, unique identifiers, or web links for publicly available datasets
- A list of figures that have associated raw data
- A description of any restrictions on data availability

We collected Reactome pathways from https://reactome.org/. We collected NCI pathways from http://www.ndexbio.org/. We collected MSigDB pathways from http://software.broadinstitute.org/gsea/msigdb/index.jsp.
Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences  ☐ Behavioural & social sciences  ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

| Sample size   | 11892 gene sets |
|---------------|-----------------|
| Data exclusions | No data was excluded |
| Replication   | All computational results can be replicated by using our software |
| Randomization | N.A.            |
| Blinding      | N.A.            |

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

| Materials & experimental systems | Methods |
|---------------------------------|---------|
| n/aobraled in the study         | n/aInvolved in the study |
| ☒ Antibodies                    | ☒ ChiP-seq |
| ☒ Eukaryotic cell lines         | ☒ Flow cytometry |
| ☒ Palaeontology and archaeology | ☒ MRI-based neuroimaging |
| ☒ Animals and other organisms   |         |
| ☒ Human research participants   |         |
| ☒ Clinical data                 |         |
| ☒ Dual use research of concern  |         |