MATTE: anti-noise module alignment for phenotype-gene-related analysis

Abstract

Purpose:

Although many transcriptome analysis methods find fundamental interactions or markers of some phenotypes, preservation of module or network is still a challenge.

Methods:

The study developed a method to directly compare the transcriptome data of phenotypes and present the differences modularly, called Module Alignment of TranscripTomE(MATTE).

Results:

MATTE performs better under high noise than differential co-expression(DC) clustering in the simulation experiments but still detects differential expression(DE) and DC genes. After subsequent annotation of cell types in single-cell data, MATTE obtained the best scores in both supervised and unsupervised learning, i. e. MATTE found meaningful markers. Finally, we apply MATTE in analyzing the transcriptome of Breast Cancer(BRCA). We have found five BRCA subtypes, and the characteristic of one subtype is detected in the form of a module network.

Conclusion:

MATTE can find meaningful genes and modules, thus facilitating the downstream analysis task to obtain insight into biology.

Keywords: Transcriptome, clustering, comparative analysis

Introduction

One of the significant objectives of the multi-omics analysis, including transcriptomics, is to identify genes or recognize related mechanisms associated with phenotypes. With the development of these years, researchers' attention has gradually shifted from the analysis in a perspective of a single gene to the module or network that can consider more complex relationships. For example, WGCNA[1] is a typical method to identify complex modules and networks from transcriptome data.

One tends to be concerned about which genes, modules, or network substructures result in a change corresponding to a different phenotype. Therefore, it is necessary to solve the problem of whether the constructed module or network will reappear in samples from different phenotypes, that is module's reproducibility or preservation. Whether a module is preserved or not can be in a biological

meanness. In disease, differentiated modules may be of more concern[2,3], whereas preserved modules in evolution correspond to conservative functions and are essential. Some studies have included more information like ortholog[4–6] in analyzing species to obtain unknown genes' functions from co-cluster.

However, there are several difficulties:

- The samples often come from different batches or data sources, in which noise is a crucial confounder.
- The results must be easy to interpret and analyze in both preservation and differentiation modules as the application scenarios differ.
- Complex presentation patterns can appear between modules, such as a part overlaps and the core structure of modules retains. Such patterns are different. A method is expected not to ignore these differences.
- A general analysis framework is needed to suit different cases.

Most current studies focused on how to map modules built from different phenotypes samples. Then the following problem is how to define a suitable metric for deciding whether a module is preserved. As summarized in the previous article[7], it can be roughly divided into network-based or cross-tabulation-based. Network-based approaches, however, have failed to address the problem that modules usually show their meaning only if they are preserved or differentiated. Although the cross-tabulation method can analyze the subdivision components of module differences, the analysis results are affected by many factors, such as clustering methods. Furthermore, step errors can be aggregated in the final results of these approaches.

Some studies also want to construct a differential module directly but can only extract one aspect of data, such as building a differential co-expression network—little research composite DE and DC to detect differential modules or networks. For example, COSINE[8] converges DE and DC on one network, but the problem of low computation efficiency appears in large data. This means that filtering is often required before computing can proceed. However, the filtered out genes can involve some functions, and whether such screening is reasonable also needs to be considered.

In this research, motivated by the natural language process where a word in a sentence can be represented by its context, we hypothesize that genes' expression should be considered in a global view. Relative differential expression(RDE) evaluates direct expression difference and global genes atlas. A gene's RDE calculates its expression difference to each gene in each phenotype. A similar strategy can also consider co-expression, defined as relative differential co-expression(RDC).

A Module Alignment of TranscripTomE (MATTE) pipeline is designed to cluster and perform module alignment. MATTE aligns modules directly by (Fig 1 b)

1. calculating RDE or RDC transformed data,
2. clustering to assign genes from each phenotype a label,

3. separating genes into preserved and differentiated modules by cross-tabulation.

This way, modules can be robustly built for specific phenotypes and aligned directly.

In the simulation, the advantages of relative difference embedding are demonstrated. MATTE shows a strong anti-noises ability to detect both differential expression(DE) and differential co-expression(DC) compared to other differential clustering methods. Then in single-cell RNA sequencing, transferable markers can be found by a modified MATTE for gene ranking. What is more, MATTE can find meaningful cancer subtypes by modules and explain the subtypes by a module network. These results show MATTE's robustness, stabilization, and explainability.

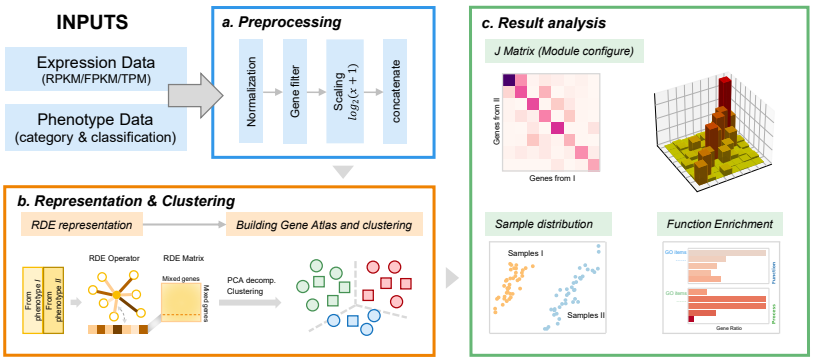

Figure 1: Work flow of MATTE. MATTE takes expression data and phenotype data as inputs. (a):Preprocessing: including normaliztion, filtering, scaling and finaly concatenate data. (b): representation and clustering: using relative difference to characterize genes, decomposition for easy calculations and clustering genes from different phenotypes. (c): Result analysis: relocate genes by their two labels(MC), and analysis specific gene set for function enrichment or sample presentation.

Results

Overview of MATTE

Most research on transcriptome comparing has been carried out by clustering separately on different phenotypes and aligning modules[9,10]. MATTE takes transcriptome data and phenotype data as inputs hoping to construct a space where each gene from different phenotypes is treated as an individual one. By doing so, changes in gene localization from different phenotypes are observed to determine the contribution of genes to phenotypic differences. The same description system for other genes and anti-noise strategy that deals with the

potential noise interference of data from different sources is needed to construct such a space.

Here we define the relative differential expression as the difference between each gene and other genes. The noise can be offset as a whole, and global information is delivered to a single gene in relative difference. Such relative differences can be applied to different statistics, such as the mean statistics to measure the degree of differential expression and Pearson correlation coefficients(PCC) to detect co-expression differences. To facilitate the follow-up calculation, we reduce the dimension of the corresponding data by the most common method of principal component analysis(PCA).

By transforming and clustering relative differences, each gene will assign two labels corresponding to two phenotypes. We can relocate genes to module configs(MCs) according to the labels. MC with the same labels is considered not to contribute to phenotype difference(differentiated), while others do(conserved). To evaluate a particular MC's contribution, signal to noise ratio(SNR) of the eigengene of the MC is calculated. And based on a module's SNR, we popularize the algorithm for multi-phenotypes gene ranking. And for an MC, gene set analysis can be performed to annotate biological function.

Simulations show relative difference's accuracy, robustness, and flexibility.

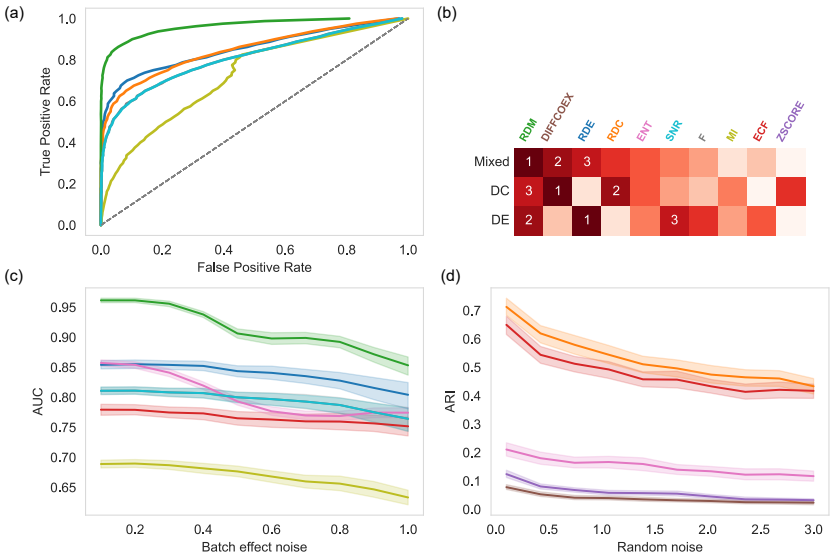

Figure 2: (a): ROC curves of methods in simulations with mixing DE and DC pattern. (b): Rank heatmap of methods in three expression patten simulations. (c): AUC value of ROC when adding batch-effect like noise. (d): ARI of clustering results of before and after adding random noise.

We designed several simulation experiments and compared MATTE with two kinds of methods. Some methods[11–13] aim to identify connection changes of a gene pair. In subsequent experiments, a sum of the distance between a specific gene and other genes is considered its score, and so are RDE and RDC.

The data with differential expression patterns are generated in a multivariate normal distribution. Nine differential expression patterns combine DE and DC in three levels: strong, weak, and none. In the simulation, DE genes are shown as the deviation of the mean value of genes in two conditions. DC genes are presented as a difference in connectivity to other genes in a differential co-expression network, where the PCC builds the graph. All DE and DC genes are labeled as positive and others as negative.

Here, two metrics are used to evaluate methods to find differential expression genes. One is the area under the curve (AUC) of the receiver operating characteristic curve (ROC), which assesses how well the score or rank of genes is consistent with the actual label. The other is the adjusted rand index(ARI) used to test the similarity between the two clustering results.

MATTE identified both DE and DC genes accurately In the following experiments, we try to verify the distance of genes in relative difference space reflects the DE and DC status. As shown in Fig 2ab, RDE and RDC achieved the top three AUC in DE and DC, respectively. By fusing the two indexes as RDM, DE, and DC in complex expression mode can be identified; And compared with other methods, it can get the highest AUC, while other methods that claim to be able to recognize both DE and DC have limited effect.

MATTE works robustly under a high noise We test two kinds of noise based on the above experiments, similar to the batch effect and random noise (see method). We randomly assigned a value to 20% of the genes in the generated data for batch-effect-like noise. As shown in Fig 2c, MATTE remains the highest AUC as the noise increases. In the random noise test, we randomly selected 100 genes from real data (100 times, each time the genes were re-extracted). In this test, clustering results using data with and without noise were compared. Compared with the traditional method of calculating the PCC from each edge, the correlation is easily affected by noise, and the RDC is more robust(Fig 2d).

MATTE can identify marker genes in single-cell RNA sequence data

These simulation experiments may not fully reflect the advantages of MATTE, mainly since the differential expression mode is limited to the generation of simulated data. We tested it on the single-cell data. The unique complexity of single-cell data is that many low-expressed genes can not be identified, known as dropouts, because of sequencing accuracy problems. Although some studies have tried to fill the data with deep learning methods, it still needs to consider whether the data will lose its original nature. Identifying cell types is necessary for analyzing single-cell data, and current marker-based single-cell annotation tools

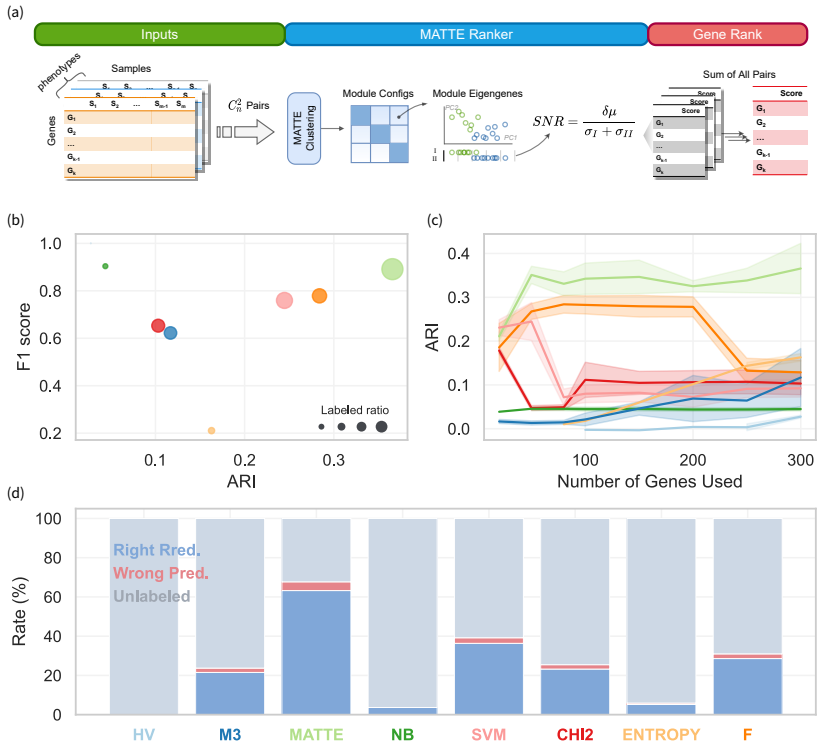

Figure 3: Gene Ranking and results of scRNA seq data.(a) Modified Pipeline of MATTE used in gene ranking: each phenotype pairs will be evaluated separately and in a calculation, SNR of each module will be take as a gene's score. Results from each turn will be added as final score. (b) Average ratio of prediction of methods: MATTE predict most cells. (c) With the number of marker genes increases, the performance of unsupervised learning task changes.(d) Comprehensive performance of methods in this task.

rely heavily on prior information[14], that is marker lists. In particular, immune cells can have different cell type annotation depths. As a result, current methods work poorly on it[14,15]. This part tried to explain the differences in expression in cell types by constructing the module. By building the corresponding modules, the expression of the gene individual can reduce the influence of the dropout event on the cell type annotation. A modified method of MATTE identified markers in the view of the module is shown in Fig 3a. In brief, a gene's score is calculated as the sum of SNR of the module the gene belongs to in each phenotype pair.

We applied MATTE, classical feature selection method, differential co-expression method, and differential gene identification method for single-cell data to

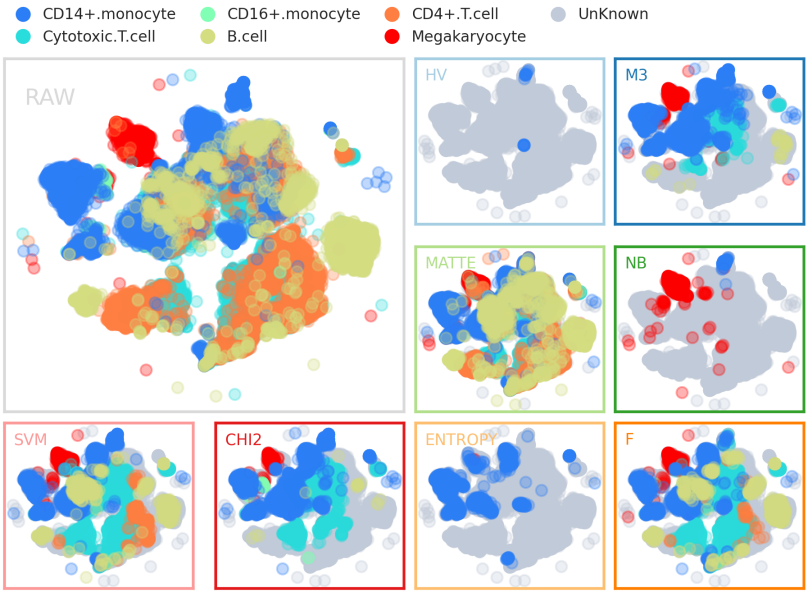

Figure 4: t-SNE Visulization of preditions of methods. Each color refers to a cell type, while grey presents unknown cell types(probability of predition is lower than 0.7).

PUMCBench[16]; the 10Xv2 data set was used as the training set to search for the critical marker genes in six cell types appearing in all data sets. The corresponding rank list is applied to the support vector machine(SVM)-based classifier with a linear kernel function. Based on the recommendations of the previous study[14], we labeled the sample with a prediction probability of less than 0.7 as unknown. And calculate F1 as an evaluation index in the labeled samples. To rule out the influence of the number of the best-selected genes on our test results, 20-300 genes were selected, and the best results were analyzed.

The results showed that MATTE could identify essential genes representing cell types and scored significantly higher on the classification task than other methods(Fig3 b). The MATTE classifier had the fewest unlabeled cell types, and the top three overall predicted F1, which predicted the most significant number of cells(Fig3 d).

Some methods based on expression variation, such as the depth-adjusted negative binominal model (NB) and Brennecke's highly variable gene method (HV), have higher F1 values. Still, the ratio of labeled cells is much smaller than MATTE. This may indicate the existence of some distinct cell populations with significant differences in the expression of a few genes.

As the number of selected genes increases, the prediction accuracy of previous

classification tasks will also gradually increase because of the low weight of the confounder genes in the SVM classifier. The classification cannot reflect the existence of confounders in the rank. Therefore, we tested it further in an unsupervised task with hierarchical clustering in the euclidean distance. And ARI is used as evaluating indicator. Then, if the features include features that cannot reflect the cell type, the corresponding ARI will decrease. As a result, there are many confounding genes in the marker list identified by most methods; that is, ARI decreases with the increase of the number of used genes. In contrast, the MATTE-sequenced list contained fewer confounding factors. ARI may even rise as the number of genes increases(Fig3 c).

Case study: transcriptome of breast cancer

We further hope to show how to analyze a specific biological case using MATTE. Here we select Breast Cancer(BRCA) in the TCGA project. Breast cancer is one of the most common cancers of women, and the detection of subtypes of BRCA has attracted many studies' attention[17,18]. Most used PAM50[19] subtyping can be used to select an appropriate treatment for patients. In contrast, triple-negative breast cancer(TNBC) lacks therapeutic targets, and drugs are limited, causing the challenge of medicine. In recent years, many studies have focused on finding druggable targets and developing drugs available for TNBC[20,21]. Here, we hope to compare the normal and cancer transcriptome of breast cancer, understand the potential biological information, and promote the development of relevant, targeted drugs.

We first performed routine MATTE analysis on cancer and normal samples in TCGA. Still, to better mine the data characteristics, we ran the MATTE pipeline of detecting DE and DC, respectively, for cancer data to contain both. The module then characterizes the sample. In short, it is to extract the sample difference information contained in the genes in the module through PCA, that is, module eigengene(ME). The results of the two processes are combined. Subsequently, the module with a low signal-to-noise ratio(SNR) was deleted (< 0.5). Then we use hierarchy clustering and correlation distance to cluster the samples.

After obtaining the subtypes of the sample, we then analyzed the overall survival time. The log-rank test results showed that the p-value was 0.006, which showed significant survival time differences between different subtypes. Specifically, the prognosis of G3 is better, and that of G4 is worse. Later, we compare the corresponding clusters with the PAM50 and find that the G2 and PAM50-Basel are consistent and related. At present, PAM50-Basel is considered synonymous with TNBC in many pieces of research.

We hope to analyze the G2 typing from the perspective of modules to obtain some new conclusions. Then we extract the module network that significantly distinguishes G2 (Fig. 5c). The genes in the core network were analyzed by functional enrichment. The enrichment analysis can be divided into three parts.

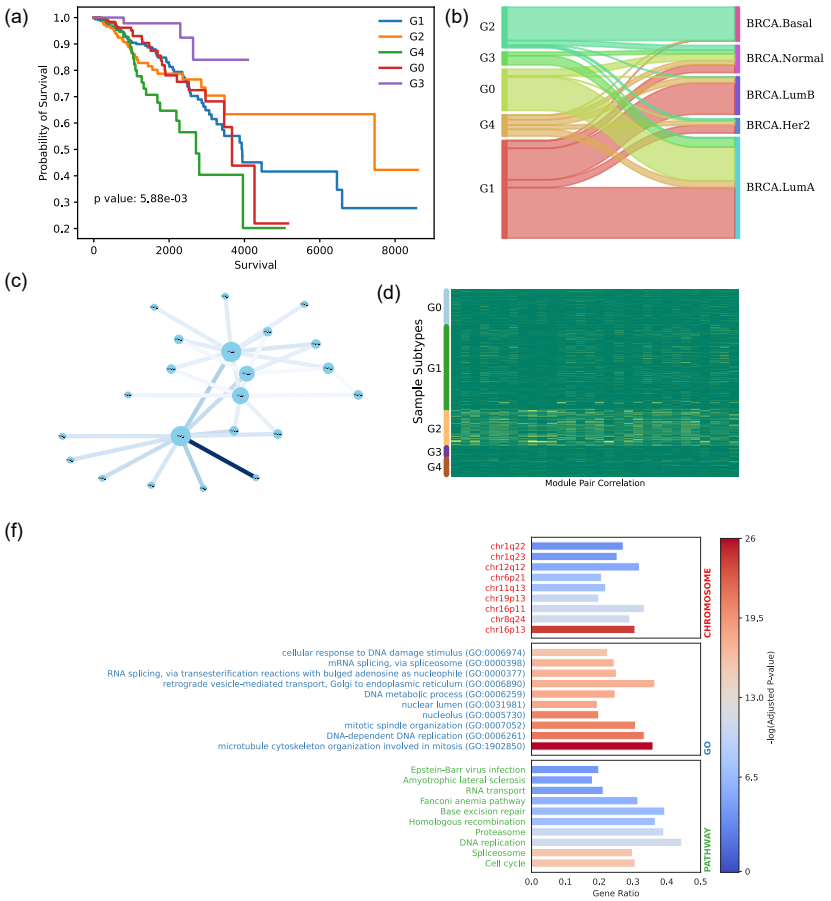

Figure 5: Subtypes of BRCA detected by MATTE.(a) Survivals of five BRCA subtypes, getting p-value 0.006 by multivariate log-rank test. (b) Sanky plot showing the overlaps of our subtypes with PAM50. (c) Core module network that specify G2. The color of edge of the graph represents the difference of G2 and other subtypes. (d) Correlation of pairs of modules in the (c). (f) Function enrich analysis of genes in the core modules network as (c) draws.

Regarding the enrichment of a chromosome location, chr16p13 is exceptionally significant. Many studies have confirmed that mutations and abnormal expression of the region are highly correlated with breast cancer[22–24]. The result of gene enrich in gene ontology(GO) shows the microtubule cytoskeleton involved in mitosis is highly correlated with the development of TNBC, which is highly consistent with the clinical practice and some research[25,26].

Discuss

Although the perspective of modules has been applied to many studies and methods, aligning modules has not been considered in recent years. We do not try to analyze and align the modules that have been detected but identify the modules directly from the raw data. In this way, it is hoped to reduce the error caused by building modules separately and accumulate in the final result. In our method, the change of genes in the clustering results replaces the specific distance calculation to reduce the impact of random disturbance on the final results. The number of clusters is the scale or resolution of the analysis. We believe that on this issue, based on the theory of causal emergence[27], more information can be obtained than a single gene.

There are all kinds of noise in transcriptome data. Although there are various methods to eliminate the corresponding noise, it is still valuable to construct methods with the ability of anti-noise. The method based on module alignment also makes MATTE robust to deal with fluctuations. It is conceivable that if noise is generated for the whole, such noise will be eliminated in the alignment between a single gene and the others, while those genes that produce differences will make the differences more evident compared with context genes.

We hope to introduce network theory and deep learning methods into MATTE in the future. As mentioned above, the expression of a single gene is considered in a broad context, but not all genes can affect gene expression. It is attractive to introduce transformer models, attention strategy, and graph neural networks into this field.

Materials and Methods

Overview of MATTE calculation

The core idea of MATTE is to construct a special space in which each gene from each phenotype has a location. By comparing the difference in gene location between the two phenotypes, we can determine whether the gene has an apparent difference between the two phenotypes. This comparison can not be carried out directly in the original data space because of the sample size disparity.

The average value is used in this study to eliminate the influence of sample order. It is worth noting that it is possible to use other statistics to describe the whole, such as the median.

Here, we use a relative differential expression to embed genes. When considering the DE, the difference between each gene and other genes is calculated in absolute value to convert this value into the distance. It is worth noting that using different statistics to describe the whole is possible, such as the median. In the condition of DCE, a normalized spectral embedding is performed based on the graph constructed from Person correlation ecoefficiency.

This representation is consistent with the idea that the distance from three points in a two-dimensional plane can determine the position of an issue. Similarly, in NLP, words are expressed in the context of a phrase.

A new problem arose: a distance in this space can be a differential change. We perform clustering to define the change of location of genes. Compared to setting a threshold of distance, clustering is thought not to overfit and has an ability to anti-noise. Moreover, genes do not work alone; they cooperate. The results in the form of gene sets can make users easy to understand the biological function difference between input phenotypes.

Specifically, genes from different phenotypes are considered independent and clustered together, further distinguished by analyzing changes in the labels of the homogenous gene in different phenotypes. If two labels of a gene corresponding to two phenotypes are the same, the gene is considered not to contribute to the divergences of two phenotypes. We constructed a matrix showing module overlaps for a more detailed analysis, drawing on the cross-tabulation method, where each unit is defined as Module Config (MC). It can be concluded that the differences of each MC lead to phenotypic differences.

Detailed Steps

Data preprocess Data preprocessing is included by default in the pipeline. The default preprocessing is as follows:

1. Normalization In bulk RNA sequencing data, fragments per kilobase of exon model per million mapped fragments(FPKM) or reads per kilobase of exon per million reads mapped(RPKM) in the data are converted to transcripts per kilobase million(TPM).

   $$X_i^{'} = \frac{X_i}{\sum_j X_{ij}} \times C$$

   Where $X \in \mathbb{R}^{n \times m}$ is transcriptome data of a phenotype with n genes and m samples; i is a specific gene, j is a particular sample; $C$is a constant. When converted to TPM, $C = 10 ^ 6 $(default). In single-cell data, $C = \frac{\sum x}{n \times m}$.

2. Gene filtering Delete genes with low expression. In this study, genes' expression lower than 1 in over 90% of samples will be dropped.

3. Scaling Use log transform to reduce the influence of genes with extremely high or low expression.

$$X' = log_2(X + 1)$$

Gene Embedding Gene embedding is based on the difference between a gene and all other genes. For DE, absolute of means' subtract is used. And use a PCA to decompose the matrix.

$$RDE = PCA(\left|\bar{X}_i - \bar{X}_j\right|)$$

For DC, a network where the nodes are genes and the edge is the PCC between genes is first constructed.

$$A^p_{ij} = corr(X^p_i, X^p_j)$$

Then perform a spectral embedding where a normalized Laplacian matrix is used to decompose. And finally, concatenate embeddings from two phenotypes.

$$RDC = [L^{p1}, L^{p2}]$$

Clustering In this study, the absolute value distance of the PCC is used to cluster the reduced-dimension data with K-Means.

$$r = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{x_i - \bar{x}}{\sigma_x}\right)\left(\frac{y_i - \bar{y}}{\sigma_y}\right)$$

$$d_A = 1 - |r|$$

Analysis After clustering, each gene has two labels, l_{p1} and l_{p2}, corresponding to two phenotypes. By the labels, all genes were further divided into a single MC that contains genes with the same two labels.

$$M_{lp1,lp2} = \{g^1_{p1,p2}, g^2_{p1,p2}, ...\}$$

$$label(g^n_{p1,p2}) = \{l_{p1}, l_{p2}\}$$

where *label* is a function that maps genes to their labels.

For each MC, module eigengene(ME) is computed to characterize samples in the view of the module. Referring to WGCNA, ME is the first component of data. The data is the expression of genes in the MC when considering DE. While considering DC, the data is the inter-individual correlation(IIC) referring to the previous studies[28,29].

$$ME_M = PCA^{(1)}(X_i), i \in M$$

Here we use IIC for most variant gene pairs in the MC, that is:

$$|corr(g^{p1}_{k1}, g^{p1}_{k2}) - corr(g^{p2}_{k1}, g^{p2}_{k2})| \geq \theta$$

$$IIC_k = \frac{(y_{k1} - \bar{y}_{k1})(y_{k2} - \bar{y}_{k2})}{\sigma(y_{k1})\sigma(y_{k2})}$$

Where y_{k1} and y_{k2} are the expressions of genes g_{k1} and g_{k2}, two genes are from the MC.

Gene ranking

Based on distance This study calculates the Euclidean distance of two genes corresponding to phenotypes based on gene embedding described previously. To detect both DE and DC genes, we merged distances of RDC and RDE to Relative Differential Merged expression(RDM) based on standardization.

$$d_i = \sqrt{(E_i^{p1})^2 + (E_i^{p2})^2}$$
$$d_{RDM} = z(d_{RDC}) + z(d_{RDE})$$
$$z(X) = \frac{X - \bar{X}}{\sigma_X}$$

Based on module Based on the MATTE framework, we design a module-based gene rank method. The core idea is to represent the score of genes in the module by the total score of the module. This study uses the SNR to calculate the scores. For the SNR of module M:

$$SNR_M^{pi,pj} = \frac{\left|mean(ME_M^A) - mean(ME_M^B)\right|}{\sigma(ME_M^A) + \sigma(ME_M^B)}$$

Where $A = \{s_1^i, s_2^i, ..., s_n^i\}$, $p(s_n^i) = p_i$ is a set that contains all samples of one phenotype.

When analyzing multiple phenotypes, scores of phenotype pairs are added:

$$SNR_g = \sum_{pi \in \mathbf{p}, pj \in \mathbf{p}, pi \neq pj} SNR_{Mg}^{pi,pj} g \in Mg$$

Simulations

Generation of data The generation of the simulation data is based on the `numpy` package's multivariate normal distribution method. We consider genes' expression pattern as a mixture of two modes: DE and DC. There are nine mode combinations as each mode has strong, weak, and none three levels. The genes with no DE and DC are the negative genes. The method needs to provide a semidefinite matrix as the covariance matrix of the variables. The following describes how the covariance matrix is generated.

It is known that the covariance matrix is the result of the PCC and standard deviation of two variables. Therefore, the covariance matrix can be obtained by generating the PCC matrix. In this study, it is considered that $\sigma_X = \sigma_Y = 1$

The properties of PCC can be summarized:

1. Is a positive semidefinite matrix
2. Diagonal elements are one.

The covariance matrix can be obtained by generating an arbitrary matrix E.

$$A = CC^T$$
$$C_{ij} = \sqrt{\frac{D_{ij}}{\sum_{j=1}^{j=n} D_{ij}}}$$
$$D_{ij} = \begin{cases} 0, e_{ij} \geqslant \lambda \\ 1, e_{ij} < \lambda \end{cases}, e_{ij} \sim U(0,1)$$

It can be proved that A is a positive semidefinite matrix whose diagonal element is 1, the non-diagonal element is less than 1, and is expected to be $\$\lambda\$.A_{ii} = \sum C_{ij}^2 = 1$

$$\mathbb{E}(A_{ij}) = \lambda, i \neq j$$

Noise simulation

Random Noise In this study, it is considered that the noise conforms to the normal distribution:

$$\varepsilon \sim \mathcal{N}(0, I)$$
$$X' = X + \varepsilon$$

Where I represent the variance of the normal distribution of the noise, a different I value is taken to simulate the noise intensity. In this study, value is assumed to simulate the noise intensity.

Batch-effects-like noise We randomly select a certain proportion of genes for the batch-effects-like noise and add a fixed amount of expression to them.

$$G_b \in G$$
$$X' = X + \lambda \times \mathbb{B}(X)$$
$$\mathbb{B}(X_i) = \begin{cases} 0, i \notin G_b \\ 1, i \in G_b \end{cases}$$

Evaluations

Classification In this step, the connectivity of the gene is taken as the score for the gene.

$$D_i = \sum_{gj \in G} A_{ij}$$

Where the A is the weight of the edge, the Area under the Curve(AUC) of the receiver operating characteristic curve(ROC) is used as the ranking index.

$$AUC = \frac{\sum_{i \in \text{ DiffClass}} \text{rank}_i - \frac{M(1+M)}{2}}{M \times N}$$

M and N are the number of positive and negative labels, respectively. $rank$ is the result of gene sequencing for a particular method.

Clustering Adjusted Rand Index(ARI) is used as the indicator to compare the labels obtained from the data with and without noise.

$$ARI = \frac{RI - E[RI]}{max(RI) - E[RI]}$$
$$RI = \frac{a+b}{C_2^{\#samples}}$$

Where a, b is the number of pairs of elements in the same class and not in the same category in both tags.

Single-cell RNA sequencing Analysis

One of the challenges of single-cell sequencing data is the dropout of low-expressed genes due to instrument accuracy. A dropout can be thought of as a specific type of noise, making labeling and annotating cells difficult. Some of these methods rely on a priori information that influences the performance decisively. We hope to apply the modified MATTE for gene ranking and other multiple methods to this problem, select several features based on ranking, and annotate the cell type according to the chosen features.

Specifically, the top k features of the ranking ($k \in \{20, 50, 80, 100, 150, 200, 250, 300\}$) is used in the following two parts. The first one is a supervised learning task, where a linear kernel support vector machine(SVM) with probability calibration is used to label cell types. Label cells with a maximum prediction probability less than 0.7 as unknown cells and the F_1 score as the indicator.

$$F_1 = \frac{1}{|L|} \sum_{l \in L} \frac{2 \times precision \times recall}{precision + recall}$$
$$precision = \frac{TP}{TP + FP}$$
$$recall = \frac{TP}{TP + FN}$$

TP is the true-positive ratio, FP is the false-positive ratio, FN is the false-negative ratio, and L is all cell types.

The SVM classifier can recognize important features by assigning the linear kernel's weight as the number of features increases. The experiment does not reveal whether the gene in front of the sequence contains some of the confounders. Thus, the second part is designed where cells are clustered on K-means using Euclidean distance. Both parts were tested on six cells common to seven data sets of PumcBench. 10xv2 data set is selected as training data and verified on the other six data sets.

BRCA bulk RNA sequencing Analysis

For DE and DC, the routine process of mate was calculated. Then, we consider that the ME calculated by two results is spliced after the Z transform.

The ME of modules whose SNR is higher than 0.5 is used as the samples' feature in sample clustering. A hierarchy clustering with PCC distance groups samples into five classes.

Overall survival(OS) analysis is based on python package `lifelines`. A multivariate log-rank test shows if different sample classes have different prognoses.

Subsequently, we compared our BRCA genotyping with PAM50 and found a high overlap between G2 and PAM50-Basel. So we go further and analyze the characteristics of G2. The feature connections of G2 are extracted, and the modular network is constructed. Network visualization is based on the python package `networkx`.

Function enrich the analysis of core network is based on `gseapy`, where gene set data is from `enrichr`.

Briefings to the compared methods

Differential Co-expression related

1. Expected conditional F statistic(ECF)[11] calculates the F statics under the expected condition. ECF combines DE and DC. Python implementation refers to the R package `cosine`[8].

The following three methods' python implementation refers to R package `dcanr`[30].

2. Z-score[12] converts the PCC of gene pairs into statistics of gene triplets.
3. Entropy[31] of PCC can be calculated based on probabilistic graphical models.
4. DiffCoEX[32] constructs a scale-free network as WGCNA does and uses the topological overlap to calculate distance.

Feature ranking methods

1. ANOVA F-value is based on the sum of squares and the ratio of intra-and inter-group deviations in different label groups. In this study, the features are ranked by the value of the F statistic.
2. The chi-square value measures the dependency and independence between data and labels.

DE methods for scRNA M3drop is an R package for single-cell expression data's DE analysis. Three differential expression methods were used in this study.

1. M3Drop: Under the null hypothesis, it is considered that the dropout ratio and average gene expression follow the Michaelis–Menten equation.

2. The negative binomial model (NB) models each observation as a negative binomial distribution.
3. Brennecke's highly variable gene method(HVG) looks for genes with significant changes in gene expression. It is considered that there is a linear relationship between the square of the coefficient of variation and the average amount of expression.

Source code and data availability

BRCA transcriptome can be downloaded from the TCGA project. The single-cell sequencing data set is collated in the previous article[14,16]. Six cell types that appeared in all data sets were selected for validation and testing from the single-cell data set.

Machine learning methods are based on `sklearn`[33] and `Biopython`[34]. The source code of MATTE and case analysis can be found on Github(https://github.com/zjupgx/MATTE). The package can be downloaded and installed via PyPi(https://pypi.org/project/MATTE/) and the MATTE documentation(https://mattedoc.readthedocs.io/en/latest/) is available for users to refer to.

References

1. Langfelder, P. & Horvath, S. WGCNA: An R package for weighted correlation network analysis. *BMC Bioinformatics* **9**, 559 (2008).
2. Sun, P. *et al.* Molecular Subtyping of Cancer Based on Distinguishing Co-Expression Modules and Machine Learning. *Front. Genet.* **13**, (2022).
3. Harutyunyan, A., Jones, N. C., Kwan, P. & Anderson, A. Network Preservation Analysis Reveals Dysregulated Synaptic Modules and Regulatory Hubs Shared Between Alzheimer's Disease and Temporal Lobe Epilepsy. *Front. Genet.* **13**, (2022).
4. Sun, Y. E., Zhou, H. J. & Li, J. J. Bipartite tight spectral clustering (BiTSC) algorithm for identifying conserved gene co-clusters in two species. *Bioinformatics* **37**, 1225–1233 (2021).
5. Yan, K.-K. *et al.* OrthoClust: An orthology-based network framework for clustering data across multiple species. *Genome Biol.* **15**, R100 (2014).
6. Crow, M., Suresh, H., Lee, J. & Gillis, J. Coexpression reveals conserved gene programs that co-vary with cell type across kingdoms. *Nucleic Acids Research* gkac276 (2022) doi:10.1093/nar/gkac276.
7. Langfelder, P., Luo, R., Oldham, M. C. & Horvath, S. Is My Network Module Preserved and Reproducible? *PLOS Computational Biology* **7**, e1001057 (2011120).
8. Ma, H., Schadt, E. E., Kaplan, L. M. & Zhao, H. COSINE: COndition-SpecIfic sub-NEtwork identification using a global optimization method. *Bioinformatics* **27**, 1290–1298 (2011).

9. Watson, M. CoXpress: Differential co-expression in gene expression data. *BMC Bioinformatics* **7**, 509 (2006).

10. Arbet, J., Zhuang, Y., Litkowski, E., Saba, L. & Kechris, K. Comparing Statistical Tests for Differential Network Analysis of Gene Modules. *Front. Genet.* **12**, 748 (2021).

11. Lai, Y., Wu, B., Chen, L. & Zhao, H. A statistical method for identifying differential gene-gene co-expression patterns. *Bioinformatics* **20**, 3146–3155 (2004).

12. Zhang, J., Ji, Y. & Zhang, L. Extracting three-way gene interactions from microarray data. *Bioinformatics* **23**, 2903–2909 (2007).

13. Ho, Y.-Y., Cope, L., Dettling, M. & Parmigiani, G. Statistical Methods for Identifying Differentially Expressed Gene Combinations. in *Gene Function Analysis* (ed. Ochs, M. F.) 171–191 (Humana Press, 2007). doi:10.1007/978-1-59745-547-3_10.

14. Abdelaal, T. *et al.* A comparison of automatic cell identification methods for single-cell RNA sequencing data. *Genome Biology* **20**, 194 (2019).

15. Liu, X. *et al.* Knowledge-based classification of fine-grained immune cell types in single-cell RNA-Seq data. *Briefings in Bioinformatics* **22**, bbab039 (2021).

16. Ding, J. *et al.* Systematic comparison of single-cell and single-nucleus RNA-sequencing methods. *Nat Biotechnol* **38**, 737–746 (2020).

17. Yang, Y., Tian, S., Qiu, Y., Zhao, P. & Zou, Q. MDICC: Novel method for multi-omics data integration and cancer subtype identification. *Briefings in Bioinformatics* bbac132 (2022) doi:10.1093/bib/bbac132.

18. Poirion, O. B., Jing, Z., Chaudhary, K., Huang, S. & Garmire, L. X. DeepProg: An ensemble of deep-learning and machine-learning models for prognosis prediction using multi-omics data. *Genome Med.* **13**, 112 (2021).

19. Parker, J. S. *et al.* Supervised Risk Predictor of Breast Cancer Based on Intrinsic Subtypes. *J. Clin. Oncol.* **27**, 1160–1167 (2009).

20. Xiao, Y. *et al.* Comprehensive metabolomics expands precision medicine for triple-negative breast cancer. *Cell Res* **32**, 477–490 (2022).

21. Loibl, S., Poortmans, P., Morrow, M., Denkert, C. & Curigliano, G. Breast cancer. *The Lancet* **397**, 1750–1769 (2021).

22. Andersen, C. L. *et al.* Frequent genomic loss at Chr16p13.2 is associated with poor prognosis in colorectal cancer. *Int. J. Cancer* **129**, 1848–1858 (2011).

23. Consortia Oslo Breast Cancer Research Consortium (OSBREAC) *et al.* Breast cancer quantitative proteome and proteogenomic landscape. *Nat Commun* **10**, 1600 (2019).

24. Dorantes-Gilardi, R., García-Cortés, D., Hernández-Lemus, E. & Espinal-Enríquez, J. K-core genes underpin structural features of breast cancer. *Sci Rep* **11**, 16284 (2021).

25. Kwon, A. *et al.* Potent Small-Molecule Inhibitors Targeting Acetylated Microtubules as Anticancer Agents Against Triple-Negative Breast Cancer. *Biomedicines* **8**, 338 (2020).

26. Sahni, J. M. *et al.* Mitotic vulnerability in triple-negative breast cancer associated with LIN9 is targetable with BET inhibitors. *Cancer Res* **77**, 5395–5408 (2017).

27. Hoel, E. P., Albantakis, L. & Tononi, G. Quantifying causal emergence shows that macro can beat micro. *Proc. Natl. Acad. Sci.* **110**, 19790–19795 (2013).

28. Lea, A. *et al.* Genetic and environmental perturbations lead to regulatory decoherence. *eLife* **8**, e40538 (2019).

29. Yu, H., Wang, L., Chen, D., Li, J. & Guo, Y. Conditional transcriptional relationships may serve as cancer prognostic markers. *BMC Medical Genomics* **14**, 101 (2021).

30. Bhuva, D. D., Cursons, J., Smyth, G. K. & Davis, M. J. Differential co-expression-based detection of conditional relationships in transcriptional data: Comparative analysis and application to breast cancer. *Genome Biol.* **20**, 236 (2019).

31. Segal, E. *et al.* Module networks: Identifying regulatory modules and their condition-specific regulators from gene expression data. *Nat. Genet.* **34**, 166–176 (2003).

32. Tesson, B. M., Breitling, R. & Jansen, R. C. DiffCoEx: A simple and sensitive method to find differentially coexpressed gene modules. *BMC Bioinformatics* **11**, 497 (2010).

33. Pedregosa, F. *et al.* Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* (2011).

34. Cock, P. J. A. *et al.* Biopython: Freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics* **25**, 1422–1423 (2009).