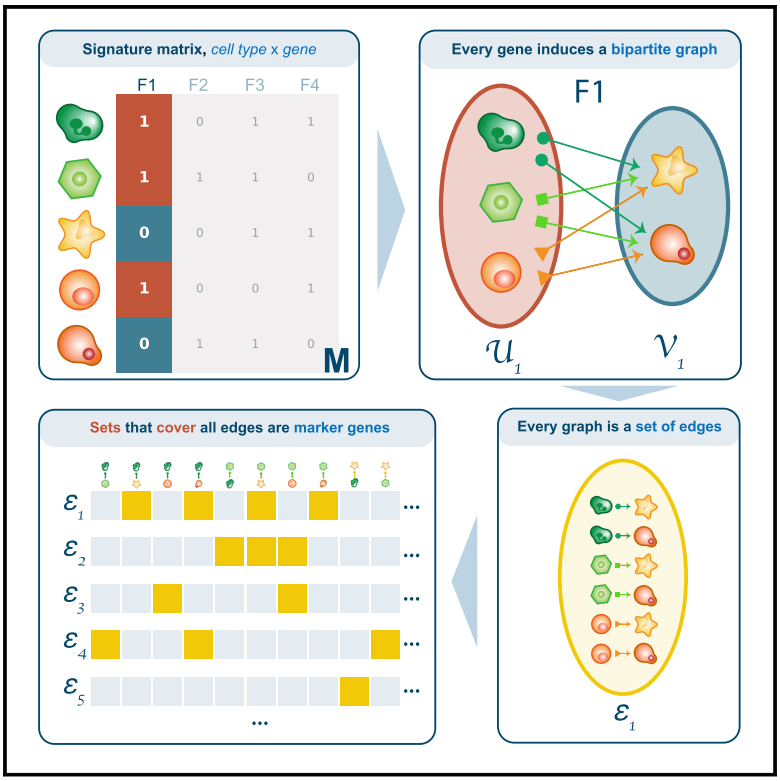
Cell Reports
Methods

Article

Multiset multicover methods for discriminative marker selection

Graphical abstract



Highlights

- Marker selection is critical to define cell types in large atlas studies
- We present phenotype cover as a conceptual formulation for the marker-selection problem
- Our algorithms for phenotype cover demonstrate high discriminatory power

Authors

Euxhen Hasanaj, Amir Alavi, Anupam Gupta, Barnabás Póczos, Ziv Bar-Joseph

Correspondence

zivbj@andrew.cmu.edu

In brief

Hasanaj et al. propose a marker-selection strategy for improving specificity and selectivity of atlas-scale cell-type assignments. They define an optimization problem for selecting a minimal set of such markers that covers all types. An analysis of the proposed approximation algorithms suggests that these marker sets have high discriminatory power.

Hasanaj et al., 2022, Cell Reports Methods 2, 100332
November 21, 2022
https://doi.org/10.1016/j.crmeth.2022.100332

CellPress

Cell Reports Methods

CellPress
OPEN ACCESS

Article

Multiset multicover methods for discriminative marker selection

Euxhen Hasanaj,[1] Amir Alavi,[2] Anupam Gupta,[3] Barnabás Póczos,[1] and Ziv Bar-Joseph[1,2,4,*]
[1]Machine Learning Department, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, USA
[2]Computational Biology Department, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, USA
[3]Computer Science Department, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, USA
[4]Lead contact
*Correspondence: zivbj@andrew.cmu.edu
https://doi.org/10.1016/j.crmeth.2022.100332

MOTIVATION To date, marker selection is based on methods that focus on each cell type separately and do not consider the relationship between different types. Such methods can select overlapping marker sets for different cell types, making it hard to discriminate between similar cell types. To address this issue and to improve the ability to select a discriminating set of markers, we defined an optimization function for biomarker selection that takes the overlap into account.

SUMMARY

Markers are increasingly being used for several high-throughput data analysis and experimental design tasks. Examples include the use of markers for assigning cell types in scRNA-seq studies, for deconvolving bulk gene expression data, and for selecting marker proteins in single-cell spatial proteomics studies. Most marker selection methods focus on differential expression (DE) analysis. Although such methods work well for data with a few non-overlapping marker sets, they are not appropriate for large atlas-size datasets where several cell types and tissues are considered. To address this, we define the phenotype cover (PC) problem for marker selection and present algorithms that can improve the discriminative power of marker sets. Analysis of these sets on several marker-selection tasks suggests that these methods can lead to solutions that accurately distinguish different phenotypes in the data.

INTRODUCTION

Several international efforts focus on characterizing gene expression in different tissues, organs, disease states, and more. Examples include HuBMAP, a large NIH effort to reconstruct a three-dimensional (3D) map of the human body at the single-cell resolution,[1] the Human Cell Atlas,[2,3] the Cancer Cell Atlas,[4] and the Brain Atlas.[5] One of the first steps of studies at the single-cell level is to characterize cell states or cell types. Typically, this relies on marker genes whose expression or co-expression with other such markers indicates a cell type.[6–8] To find such markers, researchers often perform differential expression (DE) testing, where a statistical hypothesis test is used to compare the expression of genes in one group of cells versus all other groups (one versus all). These groups are usually defined by cluster, cell type, or condition labels.

To date, marker selection has mainly focused on the most significant DE genes or proteins for each group. While this works well with a small number of distinct groups (e.g., major cell types), it may not work well when there is a much larger number of groups with overlapping DE genes. In such cases, markers are not just useful for defining a specific group or type but are also critical for discriminating between similar types. Consider these large multiorgan single-cell RNA sequencing (scRNA-seq) datasets. In such datasets, we may be interested in markers that are specific for both a cell type and a tissue (i.e., markers that are uniquely found only in cell types from this tissue). Such markers can be less significant than overall DE genes since they may only distinguish between two similar types, but they are still of major importance. An example is given by the Tabula Muris dataset,[9] a collection of scRNA-seq profiles of over 100,000 cells from over 20 different organs and tissues in *Mus musculus*. When analyzing these data, the authors used traditional clustering and DE analysis without considering the issue of cell-type/tissue combination. Another example are T cells, which mature in the thymus.[10,11] While T cells later migrate and reside in tissues throughout the body, the identification of T cells that have recently left the thymus (recent thymic emigrants [RTEs]) plays a role in treatment decisions.[12] Similarly, the role of resident and infiltrating immune cell types is still an active area of research for neurodegenerative diseases. A key challenge is the current inability to distinguish the resident central nervous system (CNS) immune cells and the

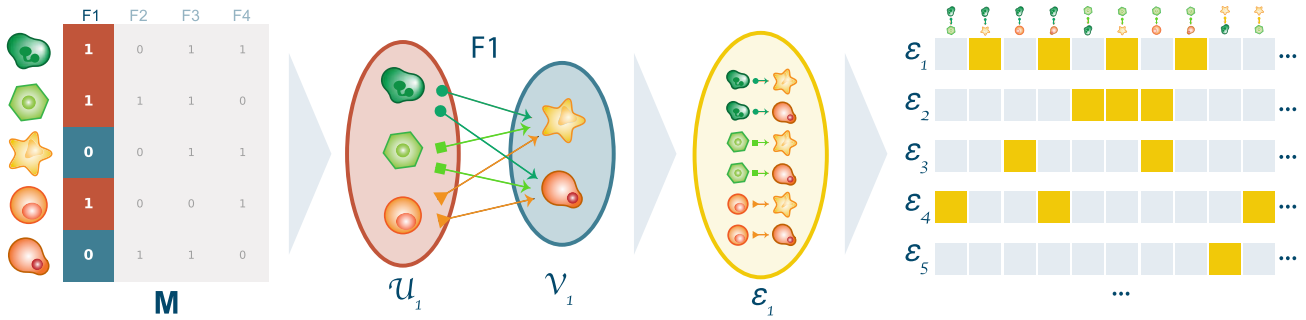


Figure 1. Graphical illustration of (binary) phenotype cover and its reformulation as a set cover problem

Given a binary score matrix (left), each feature induces a bipartite graph between classes (center left). Edges in this graph form a set ε_s. Multiset multicover is then performed on the collection of ε_s to select a small number of features that "distinguish" all phenotypic pairs (at least K times). The idea can be naturally extended to non-binary score matrices by assigning a multiplicity to each element $e_{i,j} \in \varepsilon_s$ (STAR Methods).

bone-marrow-derived immune cells.[13] Better signatures of CNS-specific immune cells and signatures of infiltrating immune cells are needed to understand the immune responses to therapies.

In addition to cell-type/tissue markers, multivariable label partitioning is central to many other questions in functional genomics. Cell type or disease states are often simultaneously considered when identifying markers,[14] and so state-specific markers for cell types are of interest. Deconvolution of cell types from bulk data is also highly dependent on the ability to select not just good markers for each individual cell type but also a set of discriminatory markers between all types.[15,16] Finally, a number of recent single-cell proteomics technologies, including CODEX[17] and Cell DIVE, require the pre-selection of markers to profile. The ability to identify a subset of markers that would suffice for distinguishing all cell types in the sample is a key criterion for such a selection.[18]

Broadly, marker selection represents a feature-selection problem. Feature-selection methods can be largely divided into three categories: filters, wrappers, and embedded.[19–21] Wrapper and embedded methods interact with a specific classifier. Wrapper methods select (often in a greedy manner) a subset of the features that lead to a classifier with the highest accuracy. Examples include sequential forward and backward selection methods.[22,23] Embedded methods use the output of the classifier itself, which comes in the form of an explicit ranking of the features or implicitly via a scoring system (e.g., information gain in decision trees[24]). Since these methods are geared toward classification, they may not be applicable to other problems, including deconvolution.

Filter methods, on the other hand, are not tied to a specific classifier. For example, scGeneFit[25] selects those genes that maintain a separation of the different cell types similar to that of the original space. This method supports both a flat partition or a hierarchy of labels (e.g., major cell types and subtypes). RankCorr[26] works in a one-versus-all fashion and selects markers for a fixed cell type by performing a rank transformation. Another algorithm, Relief,[27] and its extension ReliefF[28] penalize features that cannot distinguish a given instance from its negative (having a different label) neighbors, while assigning high scores to features that take similar values among instances from the same class. Minimum-redundancy-maximum-relevance (mRMR) selects features that are relevant to the target class but are not similar to each other.[29] CIBERSORT[15] and a number of prior methods[16,30,31] analyzed a signature matrix of DE genes to identify submatrices with a low condition number for use in deconvolution of bulk mixtures. Thus, while these methods can successfully select discriminative features when the overlap between sets is small, the ability of such methods to select markers that discriminate all pairs of phenotypes has not been extensively studied.

In this article, we explore the problem of determining a global set of biomarkers. These represent features that collectively distinguish between higher context phenotypes. We assume we are given a phenotype × feature, binary or real score matrix **M**, whose (*i*,*s*) entry represents the relevance of feature *s* (e.g., average gene expression) for phenotype *i*. We formulate the task as a combinatorial optimization problem where the goal is to identify the smallest set of features such that for every phenotypic pair (*i*,*j*) there exists a set of features that can be used to "distinguish" between *i* and *j*. We term this problem phenotype cover (PC). We show that PC is equivalent to multiset multicover, which is nondeterministic polynomial-time (NP)-complete[32] and propose two algorithms that can approximate it in polynomial time (STAR Methods). The first is based on the extended greedy algorithm to set cover (G-PC),[33] and the second is based on the cross-entropy method (CEM-PC).[34,35] By analyzing several marker-selection problems, we show that the greedy algorithm outperforms competitors across a variety of tasks. We also analyze some of the specific markers selected by the method and discuss their ability to distinguish between similar cell types.

RESULTS

We developed methods to select discriminative features from a large set of (potentially overlapping) signatures. The goal of the features we select is to enable the separation of the different components in the set. This can either be for a supervised learning (for example, classification) or for other learning approaches such as deconvolution or dimensionality reduction. Our method takes as input a signature or score matrix **M**, which is used to estimate the importance of a feature for a phenotype of interest. Features are then selected by reformulating the problem as a multiset multicover instance where the goal is to select features such that every phenotypic pair is covered at least *K* times, for some positive *K*

Table 1. scRNA-seq datasets used in this study

Dataset	Genes	High var.	Cells	Tissues	Cell types	Reference
Idiopathic pulmonary fibrosis (IPF)	4,443	yes	96,301	1	33	Adams et al.[39]
Mouse cortex (MC)	20,006	no	3,005	1	7	Zeisel et al.[40]
Human cell atlas (HCA)	2,968	yes	84,363	15	7	He et al.[41]

For HCA, we consider a combination of tissues and cell types (85). For IPF, only healthy samples were kept. Endothelial-mural and astrocyte-ependymal pairs of cells were grouped for MC.

(Figure 1). We developed two solutions to the multiset multicover problem: the first is based on a greedy approach (G-PC), and the second based on the cross-entropy method (CEM-PC). See STAR Methods for details.

We tested G-PC and CEM-PC and compared them with eight prior methods: scGeneFit,[25] decision trees,[36] top differentially expressed genes (TopDE), RankCorr,[26] ReliefF,[28] mRMR,[29] ANOVA F values, and mutual information.[37,38] We used three scRNA-seq datasets from lung, mouse cortex, and a human cell atlas (Table 1). We vary the coverage factor K from 1 to 20 for the idiopathic pulmonary fibrosis (IPF) dataset, from 1 to 40 for mouse cortex (MC), and from 1 to 9 for human cell atlas (HCA). For all baselines but TopDE and RankCorr, we select a number of features that matches the solution size returned by G-PC. For TopDE, we take the union of the top k differentially expressed genes for each phenotype (k varying from 1 to <10). For RankCorr, we tuned the hyperparameters until a similar number of features was returned. Finally, for CEM-PC, all features with a probability score greater than 0.98 after convergence were chosen (Methods S1, alg. 3). We compare all methods in terms of phenotype classification performance, deconvolution of bulk mixtures, and feature stability. We also validate the features selected by G-PC by performing gene set enrichment analysis and comparing them with known markers in the literature.

Classification

We first test the ability of a classifier to predict the correct phenotype given only a subset of the features. For each method, we select a feature set S using a subset of the data, train a logistic regression model on the same subset, and evaluate performance on left-out data. G-PC exhibits strong performance on the IPF and MC datasets across a wide range of coverage factors. For example, when 42 genes are selected on the IPF data, G-PC obtains an F1 score of 0.70, followed by scGeneFit (0.65) and CEM-PC (0.61) (Figure 2A). On the MC data (Figure S1A), G-PC again performs best when 30–140 genes are selected (F1 $\approx$ 0.94–0.95). mRMR also performs well on these data except when the number of genes selected is small (<30). Decision trees, on the other hand, do not improve in performance when more than 30 genes are selected (F1 $\approx$ 0.92).

These two datasets are obtained from a single tissue. We thus next tested the ability of PC to differentiate between the same cell types across multiple tissues. For this, we used all tissue and cell-type combinations present in the HCA dataset. Decision trees outperform other methods on this classification task (Figure S2A). G-PC is the second-best method when more than 100 genes are selected, while scGeneFit is the second best when less than 100 genes are selected. scGeneFit, however, does not improve in performance when more than 100 genes are selected. At 235 genes, decision trees converge at 0.70, while G-PC and mutual information reach an F1 of 0.68.

We note that scGeneFit can take the hierarchy of labels into account and that the authors describe improved performance when cell subtypes are considered in the MC dataset. For a fair comparison, we ran three different variants of scGeneFit that take advantage of this hierarchical structure and evaluated performance by using a nearest centroid classifier fit on the entire data. All the hyperparameters we used were identical to those provided by the authors. While G-PC does not use cell subtype information, it still outperforms all three variants across a different number of markers (Figure S4B).

We also tested an additional classifier (k nearest neighbors) and observed very similar results to those obtained with logistic regression (Figures S3A–S3C). Finally, we tested the impact of batch effects by using two pancreas datasets[42,43] and observed that our method, G-PC, along with TopDE are the most robust to batch effects (Figure S4A).

Deconvolution

Inferring cell-type proportions from bulk transcriptomics data is an important task in understanding composition of tumors and other tissues. Many methods have been developed to perform deconvolution of bulk mixtures.[16,30,31,44] Deconvolution typically requires solving a linear equation of the form $m = Sp$, where m is a given mixture vector, S is a signature matrix containing cell-type-specific expression signatures (known), and p is the unknown class proportion vector. One widely used method for deconvolution is CIBERSORT,[15] which uses ν -support vector regression (ν -SVR). CIBERSORT constructs the signature matrix S by considering the top k DE genes for every cell-type subset (which leads to the exact same selection as the TopDE baseline we consider in this study). Next, CIBERSORT selects the k that leads to a signature matrix S with the lowest condition number. Finally, ν -SVR is fit on the data, and the regression coefficients in the solution are used to estimate p.

To test the usefulness of the features selected by our method for deconvolution, we constructed pseudo-bulk mixtures using the IPF, MC, and HCA datasets by averaging expression levels across all single cells in the test sets. The signature matrix S was constructed with features selected from the training set and deconvolution via ν -SVR was then applied to the pseudo-bulk mixtures. As recommended by the authors, we initialize three linear ν -SVR instances with $\nu \in \{0.25, 0.5, 0.75\}$ and save the model that achieves the lowest root-mean-square error between the deconvolution

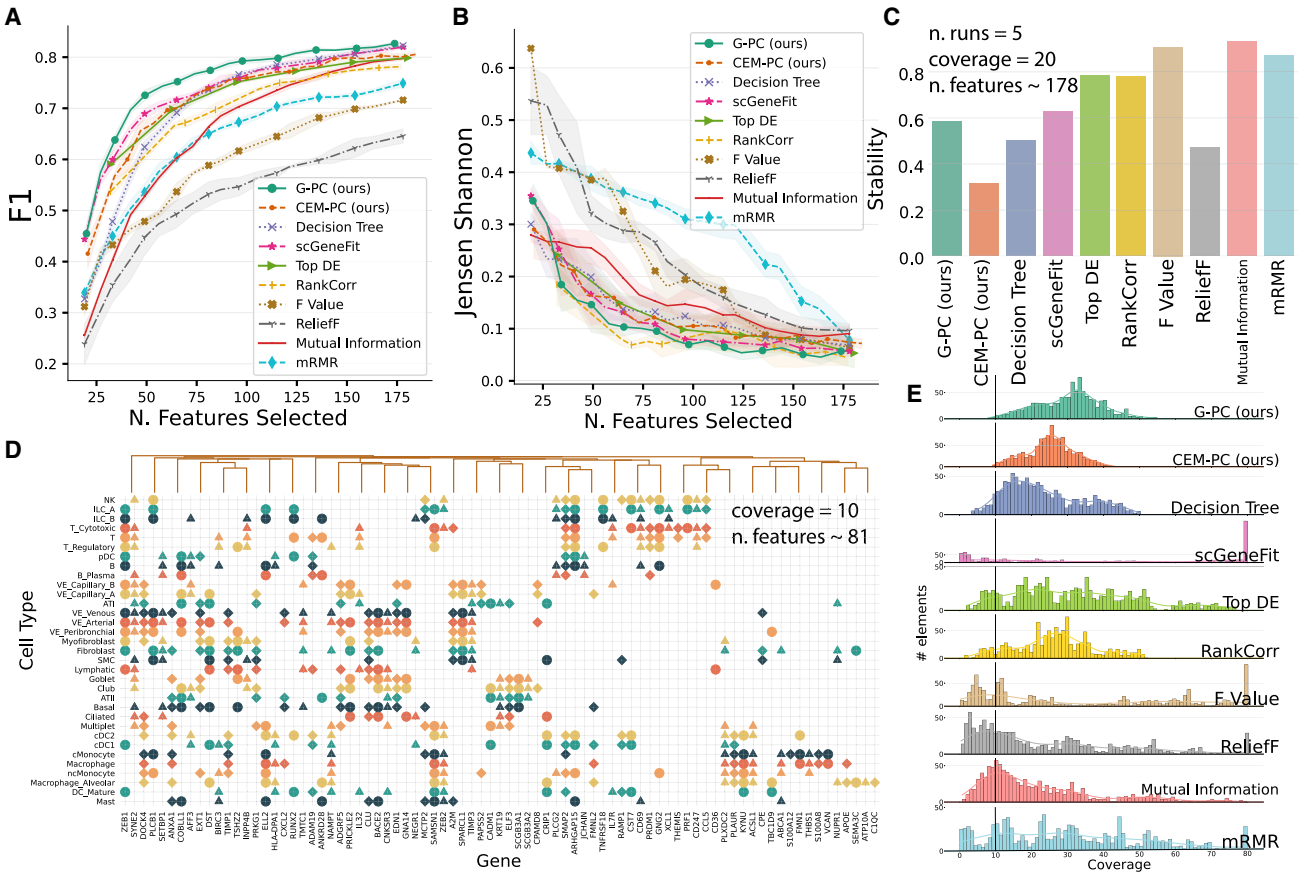


Figure 2. Comparison of feature selection methods for the IPF dataset

(A and B) Performance scores for (A) and (B) were averaged across five different random train and test splits. SD is shown as a shaded region.

(A) Performance of a logistic regression model trained on the selected features. G-PC achieves the highest F1 score across all coverage factors, followed by scGeneFit and CEM-PC.

(B) Jensen-Shannon divergence (lower is better) between CIBERSORT-predicted mixture proportions and the ground truth.

(C) Stability scores for all eight methods over 5 runs. Sequential methods like G-PC, decision trees, and CEM-PC suffer slightly in stability compared with other, more global methods. Nonetheless, G-PC shares about 70% of the features across runs.

(D) Selected biomarkers assigned to each cell type (rows [columns] are differentiated by color [shape]). Gene *s* (column) is assigned to cell type *i* (row) if there exists another cell type *j* such that $\mathbf{M}_{i,s} - \mathbf{M}j,s \geq 1$ (S. Biomarker validation). Rows and columns were ordered via hierarchical clustering.

(E) For every phenotypic pair, we compute the coverage (i.e., the score difference between the two phenotypes) provided by the selected gene set. A histogram of these coverage factors corresponding to a coverage of 10 is shown for each method. As can be seen, for G-PC and CEM-PC, which optimize for coverage, each element is covered at least 10 times. Other methods provide high coverage for some elements but miss out on others.

result *Sp* and *m*. We compute the Jensen-Shannon (JS) divergence[45] between the predicted mixture *p* and the ground truth. G-PC performs well on the IPF data, with RankCorr doing better only when 50–80 genes are selected (Figure 2B). For example, when 163 genes are selected, G-PC achieves an average JS = 0.045, followed by RankCorr (0.056) and scGeneFit (0.062). For the MC dataset, G-PC is also the top-ranking method, though TopDE and RankCorr also accurately resolve mixture proportions (Figure S1B). All three methods obtain a JS score of less than $\approx$0.025 across all *K*. CEM-PC performs well on some instances for both datasets; however, the results are unstable and vary between runs. None of the methods clearly outperforms all others on the HCA dataset (Figure S2B). These results demonstrate the challenges of trying to distinguish cell types across tissues.

Finally, we also tested another version of deconvolution that uses linear least squares (LLS) as the target. We observed that, for LLS, G-PC performs no worse than other methods on IPF and MC (Figures S3D–S3F).

Stability

The focus of the comparison so far has been on accuracy. However, other considerations are also important, especially when selecting features that will be used across different platforms and potentially modalities. One such important issue is feature stability.[46] The stability index measures the average size of the overlap divided by the size of the union for all pairs of feature sets (STAR Methods). To test stability, we randomly sample half the data and compute the stability index for the features selected by each method over 5 runs.

CellPress
OPEN ACCESS

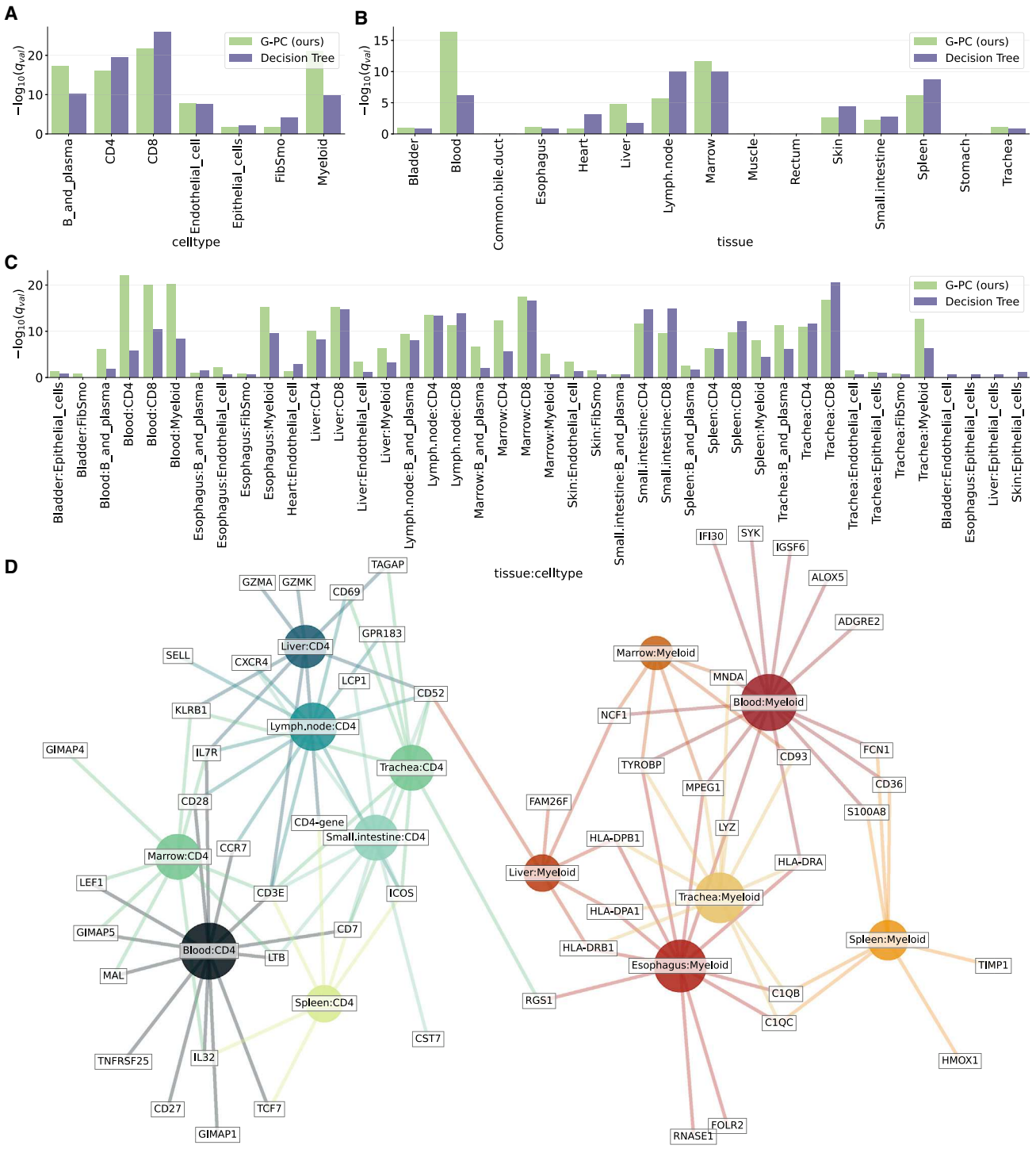


Figure 3. GSEA q values for the HCA dataset

(A and B) We select markers that provide coverage for each cell type for both G-PC and decision trees and perform gene set enrichment analysis (GSEA) using the HuBMAP ASCT + B gene set.[48] We first record q values for the top entry, which contains the correct cell type or the correct tissue independently. When comparing the ability of each method to assign the correct class, G-PC obtains a lower q value, i.e., higher-log(q value), for 42% (3/7) of the cell types (A) and 54% (6/11) of the tissues (B). We did not find markers for four tissues in the gene set (common bile duct, muscle, rectum, stomach).

(C) When tested for the ability to identify both the correct tissue and the correct cell type, G-PC obtained lower q values in 71% (30/42) of the cases. The remaining tissue/cell-type pairs (33) either belonged to a tissue that was not present in the marker set or was not identified by either method.

(D) Connected by an edge are known markers for CD4 and myeloid cells that were assigned to the correct tissue/cell-type pair by G-PC. Some markers are assigned to multiple cell types (multiple outgoing edges), while others are pair specific.

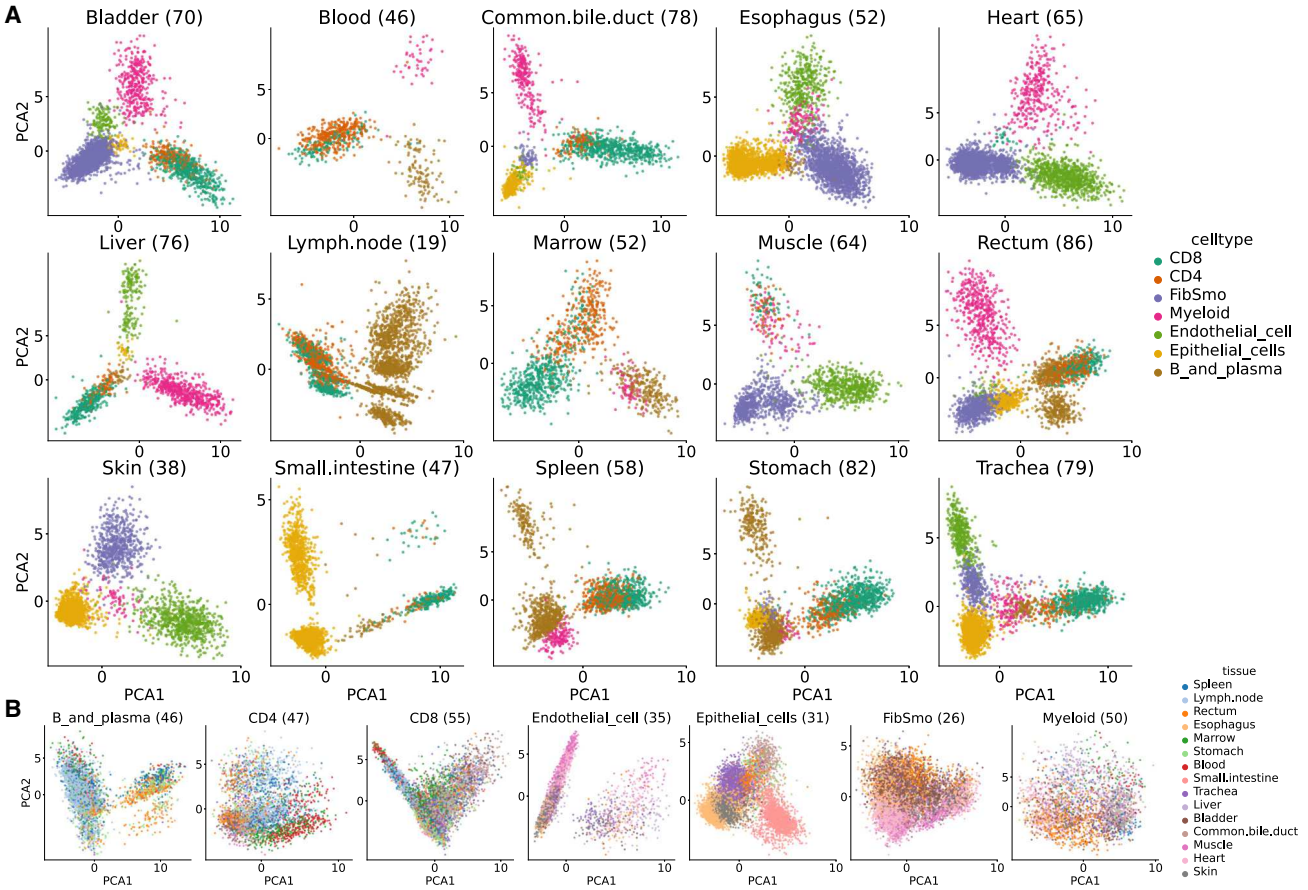


Figure 4. Principal-component analysis (PCA) plots of the selected markers for the different phenotypes present in the HCA dataset

(A and B) A total of 121 markers were selected via G-PC (coverage = 5). For every tissue (A) and cell type (B), the top two principal components of the markers providing coverage (≥ 1) for that phenotype are plotted. The exact number of markers used is shown in parentheses. There is visible separation between classes.

Stability scores are shown in Figures 2C, S1C, and S2C. G-PC is more stable than decision trees for IPF and MC. However, due to their greedy sequential nature, both G-PC and decision trees are less stable than more global methods such as ReliefF and F values. Nonetheless, G-PC uses from 60% to 70% of the same genes across all runs. Perhaps not surprisingly, due to its random sampling nature, CEM-PC is the least stable method.

Biomarker validation

To validate the set of biomarkers $\mathcal{S}$ selected by G-PC and decision trees, we performed enrichment analysis for the HCA dataset. We fix a coverage of 8, and for every phenotype i, we select from the solution $\mathcal{S}$ all those genes s for which there exists some phenotype j satisfying $M_{i,s} \geq M_{j,s} + 1$. We consider each of these sets as a biomarker set for the given phenotype for both G-PC (Figures 2D and S6) and decision trees.

We next performed gene set enrichment analysis (GSEA)[47] using the HuBMAP ASCT + B marker set[48] to determine if the selected marker sets for a specific cell type are enriched for pathways associated with these cell types. We test the ability of G-PC and decision trees to identify the correct (1) tissue, (2) cell type, and (3) tissue/cell type combination. G-PC obtains lower q values for 42% (3/7) of the cell types and 54% (6/11) of the tissues (Figures 3A and 3B). No markers were found for four tissues. When tested against the correct tissue and cell-type pair, G-PC obtained lower q values for 71% (30/42) of the pairs (Figure 3C). The remaining combinations (33) either belonged to a tissue that was not present in the marker set or was not identified by either method. Some known markers assigned correctly by G-PC are shown in Figure 3D. Esophagus and trachea tissues were mapped to respiratory system in the ASCT + B set. The top two principal components of the markers that provide coverage for a given tissue or cell type show visible separation between different classes (Figure 4).

Due to limitations of the marker set we are using, only 20 cell types could be identified for IPF. Among these, G-PC obtains lower q values for 12 (60%) (Figure S5A). We observe good agreement between genes selected using our greedy procedures and genes known to be involved in specific cell types. For example, G-PC correctly assigns *KRT19* and *ADGRF5* to type I and II epithelial cells (ATI and ATII[49–51]). *CD69* is assigned to both B and T cells,[52,53] *COBLL1* is assigned to B cells,[54] *JCHAIN* to B and B plasma cells,[55] *CXCL2* to macrophages,[56] and *CCL5*,

Table 2. Runtimes of all feature selection methods for all three datasets used in this study (s)

Data	G-PC	CEM-PC	DT	scGF	DE	RC	FVal	ReliefF	MI	mRMR
IPF	1.3	55	304	102	41	90	1.5	388	1039	980
MC	0.17	227	5	39	2.7	3	0.15	11	116	139
HCA	1.25	88	50	52	30	95	0.8	340	790	310

Method names were abbreviated. 178 features were selected for IPF, 66 for MC, and 121 for HCA. Our C++ implementation of G-PC takes less than 2 s for all three datasets, making it the fastest along with F value computation. Performance tests are conducted on a machine with a 2.3 GHz 8-Core Intel Core i9 CPU and 32 GB memory.

PRF1, and *CD247* to natural killer cells.[57–59] See Figure 2D for a larger list of identified markers.

In addition to selecting known cell-type markers, G-PC is also able to select markers that distinguish between similar cell types. For example, it assigns *CXCL2* to ATII and not to ATI[60] and *CD69* and *AFF3* to B cells and not plasma cells.[52,53,61] Another example is *A2M* and *CST7*, which are assigned to cytotoxic T cells,[62] whereas *NAMPT* and *TNFRSF18* are assigned to regulatory T cells.[63]

DISCUSSION

Selection and use of markers is a common step in many analysis pipelines. Most recently, this topic received increased attention due to the large number of new cell types that have been identified and characterized using scRNA-seq data.[64–67] To date, such selection was mainly based on methods that focused on each cell type separately and did not consider the relationship between markers selected for different types. Such methods can select overlapping marker sets for different cell types, making it hard to discriminate between similar cell types. This is especially important for large datasets where multiple cell types in multiple tissues are being profiled.[9,41]

To address this issue and improve the ability to select a discriminating set of markers, we defined a new optimization function for biomarker selection that takes the overlap into account. Specifically, we defined the PC problem that aims to optimize the accuracy of identifying different sets when using the selected markers. We presented two heuristic filter methods since these lead to solutions that can be used in several different analysis pipelines including classification, deconvolution, experimental design, and more. The first is based on a greedy approximation algorithm (G-PC) and the second is based on the cross-entropy method (CEM-PC).

We evaluated these methods and compared them with prior methods developed for marker selection using several high-throughput scRNA-seq datasets. Our analysis indicates that G-PC assigns equal importance to all different phenotypes in the data and is affected less by class imbalance as shown by the F1 score. Other methods tend to select features that discriminate only dominating classes. Furthermore, G-PC can be used with signature matrices rather than direct expression measurements. In such cases, there is only a single score for all phenotype/gene pairs, which makes using other methods difficult. This allows G-PC to construct signature matrices for deconvolution, which leads to an accurate estimation of cell-type proportions from bulk mixtures. While G-PC is slightly less stable than some other methods, it nonetheless retains most of the features (~70%) across runs.

Decision trees outperform G-PC with regard to the F1 score in one of the datasets we analyzed (HCA). However, even for HCA, G-PC seems to obtain a more accurate list of cell-type markers based on enrichment analysis. We note that our method is best suited for datasets that require detailed annotations, which usually means that several cell types partially overlap in their markers. In contrast, for large datasets where the focus is on coarser cell types, we see less advantage compared to standard marker selection methods. Finally, we provide a C++ implementation of G-PC with Python bindings, which makes it the fastest method we tested (Table 2). Speed is an important consideration when working with large scRNA-seq datasets.

We observed that CEM-PC sometimes selects a smaller set of genes that achieves the same coverage as G-PC. However, due to its random sampling nature, CEM-PC is very unstable and can lead to a completely different set of features across runs.

Limitations of study

The biological analysis relied on computing the overlap with existing gene marker lists that may be incomplete. A more thorough biological analysis of the selected genes and their relation to each cell type might provide more insight into the performance of our algorithms.

While G-PC worked well for the data analyzed in this paper, it does not provide an optimal solution. It is interesting to see if other approximation algorithms that optimize for coverage will lead to better results when tested on biological data.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- RESOURCE AVAILABILITY
  - Lead contact
  - Materials availability
  - Data and code availability
- METHOD DETAILS
  - Notation
  - Problem formulation and complexity
  - Approximating a solution to phenotype cover
  - Baselines

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.crmeth.2022.100332.

ACKNOWLEDGMENTS

This work was partially supported by NIH grants OT2OD026682, 1U54AG075931, and 1U24CA268108 and by NSF grant CBET2134998 to Z.B.-J.

AUTHOR CONTRIBUTIONS

Z.B.-J. and E.H. designed the study. E.H. and A.G. derived the theoretical results. Z.B.-J., E.H., A.A., and B.P. designed the empirical analysis and analyzed the results. E.H. wrote the software and performed the analysis. All authors contributed to manuscript writing. All authors read and approved the final manuscript.

DECLARATION OF INTERESTS

The authors declare no competing interests.

Received: May 31, 2022
Revised: August 12, 2022
Accepted: October 18, 2022
Published: November 11, 2022

REFERENCES

1. HuBMAP Consortium (2019). The human body at cellular resolution: the NIH Human Biomolecular Atlas Program. Nature *574*, 187–192. https://doi.org/10.1038/s41586-019-1629-x.
2. Regev, A., Teichmann, S.A., Lander, E.S., Amit, I., Benoist, C., Birney, E., Bodenmiller, B., Campbell, P., Carninci, P., Clatworthy, M., et al. (2017). The human cell atlas. Elife *6*, e27041. https://doi.org/10.7554/eLife.27041.
3. Rozenblatt-Rosen, O., Stubbington, M.J.T., Regev, A., and Teichmann, S.A. (2017). The human cell atlas: from vision to reality. Nature *550*, 451–453. https://doi.org/10.1038/550451a.
4. Cancer Genome Atlas Research Network; Weinstein, J.N., Collisson, E.A., Mills, G.B., Shaw, K.R.M., Ozenberger, B.A., Ellrott, K., Shmulevich, I., Sander, C., and Stuart, J.M. (2013). The cancer genome atlas pan-cancer analysis project. Nat. Genet. *45*, 1113–1120. https://doi.org/10.1038/ng.2764.
5. Hawrylycz, M.J., Lein, E.S., Guillozet-Bongaarts, A.L., Shen, E.H., Ng, L., Miller, J.A., van de Lagemaat, L.N., Smith, K.A., Ebbert, A., Riley, Z.L., et al. (2012). An anatomically comprehensive atlas of the adult human brain transcriptome. Nature *489*, 391–399. https://doi.org/10.1038/nature11405.
6. Usoskin, D., Furlan, A., Islam, S., Abdo, H., Lönnerberg, P., Lou, D., Hjerling-Leffler, J., Haeggström, J., Kharchenko, O., Kharchenko, P.V., et al. (2015). Unbiased classification of sensory neuron types by large-scale single-cell RNA sequencing. Nat. Neurosci. *18*, 145–153. https://doi.org/10.1038/nn.3881.
7. Lo Giudice, Q., Leleu, M., La Manno, G., and Fabre, P.J. (2019). Single-cell transcriptional logic of cell-fate specification and axon guidance in early-born retinal neurons. Development *146*, dev178103. https://doi.org/10.1242/dev.178103.
8. Bassett, E.A., and Wallace, V.A. (2012). Cell fate determination in the vertebrate retina. Trends Neurosci. *35*, 565–573. https://doi.org/10.1016/j.tins.2012.05.004.
9. Tabula Muris Consortium; Overall coordination; Logistical coordination; Organ collection and processing; Library preparation and sequencing; Computational data analysis; Cell type annotation; Writing group; Supplemental text writing group; Principal investigators (2018). Single-cell transcriptomics of 20 mouse organs creates a Tabula Muris. Nature *562*, 367–372. https://doi.org/10.1038/s41586-018-0590-4.
10. Charles A Janeway, J., Travers, P., Walport, M., and Shlomchik, M.J. (2001). Generation of Lymphocytes in Bone Marrow and Thymus, 5th Edition (Immunobiol. Immune Syst. Health Dis.).
11. Heath, W.R. (1998). T lymphocytes. In Encyclopedia of Immunology, Second Edition, P.J. Delves, ed. (Elsevier), pp. 2341–2343.
12. Ravkov, E., Slev, P., and Heikal, N. (2017). Thymic output: assessment of CD4+ recent thymic emigrants and T-Cell receptor excision circles in infants. Cytometry B Clin. Cytom. *92*, 249–257. https://doi.org/10.1002/cyto.b.21341.
13. Ronning, K.E., Karlen, S.J., Miller, E.B., and Burns, M.E. (2019). Molecular profiling of resident and infiltrating mononuclear phagocytes during rapid adult retinal degeneration using single-cell RNA sequencing. Sci. Rep. *9*, 4858. https://doi.org/10.1038/s41598-019-41141-0.
14. Gawel, D.R., Serra-Musach, J., Lilja, S., Aagesen, J., Arenas, A., Asking, B., Bengnér, M., Björkander, J., Biggs, S., Ernerudh, J., et al. (2019). A validated single-cell-based strategy to identify diagnostic and therapeutic targets in complex diseases. Genome Med. *11*, 47. https://doi.org/10.1186/s13073-019-0657-3.
15. Newman, A.M., Liu, C.L., Green, M.R., Gentles, A.J., Feng, W., Xu, Y., Hoang, C.D., Diehn, M., and Alizadeh, A.A. (2015). Robust enumeration of cell subsets from tissue expression profiles. Nat. Methods *12*, 453–457. https://doi.org/10.1038/nmeth.3337.
16. Gong, T., Hartmann, N., Kohane, I.S., Brinkmann, V., Staedtler, F., Letzkus, M., Bongiovanni, S., and Szustakowski, J.D. (2011). Optimal deconvolution of transcriptional profiling data using quadratic programming with application to complex clinical blood samples. PLoS One *6*, e27156. https://doi.org/10.1371/JOURNAL.PONE.0027156.
17. Goltsev, Y., Samusik, N., Kennedy-Darling, J., Bhate, S., Hale, M., Vazquez, G., Black, S., and Nolan, G.P. (2018). Deep profiling of mouse splenic architecture with CODEX multiplexed imaging. Cell *174*, 968–981.e15. https://doi.org/10.1016/J.CELL.2018.07.010.
18. Chattopadhyay, P.K., and Roederer, M. (2012). Cytometry: today's technology and tomorrow's horizons. Methods *57*, 251–258. https://doi.org/10.1016/J.YMETH.2012.02.009.
19. Bolón-Canedo, V., Sánchez-Maroño, N., Alonso-Betanzos, A., Benítez, J., and Herrera, F. (2014). A review of microarray datasets and applied feature selection methods. Inf. Sci. *282*, 111–135. https://doi.org/10.1016/j.ins.2014.05.042.
20. Tadist, K., Najah, S., Nikolov, N.S., Mrabti, F., and Zahi, A. (2019). Feature selection methods and genomic big data: a systematic review. J. Big Data *6*, 79–24. https://doi.org/10.1186/S40537-019-0241-0/TABLES/6.
21. Saeys, Y., Inza, I., and Larrañaga, P. (2007). A review of feature selection techniques in bioinformatics. Bioinformatics *23*, 2507–2517. https://doi.org/10.1093/BIOINFORMATICS/BTM344.
22. Whitney, A.W. (1971). A direct method of nonparametric measurement selection. IEEE Trans. Comput. *C-20*, 1100–1103. https://doi.org/10.1109/T-C.1971.223410.
23. Marill, T., and Green, D. (1963). On the effectiveness of receptors in recognition systems. IEEE Trans. Inf. Theory *9*, 11–17. https://doi.org/10.1109/TIT.1963.1057810.
24. Breiman, L., Friedman, J.H., Olshen, R.A., and Stone, C.J. (1984). Classification and Regression Trees (Routledge). https://doi.org/10.1201/9781315139470.
25. Dumitrascu, B., Villar, S., Mixon, D.G., and Engelhardt, B.E. (2021). Optimal marker gene selection for cell type discrimination in single cell analyses. Nat. Commun. *12*, 1186. https://doi.org/10.1038/s41467-021-21453-4.

26. Vargo, A.H.S., and Gilbert, A.C. (2020). A rank-based marker selection method for high throughput scRNA-seq data. BMC Bioinf. *21*, 477. https://doi.org/10.1186/s12859-020-03641-z.

27. Kira, K., and Rendell, L.A. (1992). A practical approach to feature selection. In Proceedings of the ninth international workshop on Machine learning ML92 (Morgan Kaufmann Publishers Inc.), pp. 249–256.

28. Kononenko, I. (1994). Estimating attributes: analysis and extensions of RELIEF. In Machine Learning: ECML-94 Lecture Notes in Computer Science, F. Bergadano and L. Raedt, eds. (Springer Berlin Heidelberg), pp. 171–182. https://doi.org/10.1007/3-540-57868-4_57.

29. Peng, H., Long, F., and Ding, C. (2005). Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. IEEE Trans. Pattern Anal. Mach. Intell. *27*, 1226–1238. https://doi.org/10.1109/TPAMI.2005.159.

30. Abbas, A.R., Wolslegel, K., Seshasayee, D., Modrusan, Z., and Clark, H.F. (2009). Deconvolution of blood microarray data identifies cellular activation patterns in systemic lupus erythematosus. PLoS One *4*, e6098. https://doi.org/10.1371/JOURNAL.PONE.0006098.

31. Gong, T., and Szustakowski, J.D. (2013). DeconRNASeq: a statistical framework for deconvolution of heterogeneous tissue samples based on mRNA-Seq data. Bioinformatics *29*, 1083–1085. https://doi.org/10.1093/BIOINFORMATICS/BTT090.

32. Vazirani, V.V. (2003). Approximation Algorithms (Springer). https://doi.org/10.1007/978-3-662-04565-7.

33. Rajagopalan, S., and Vazirani, V.V. (1993). Primal-dual RNC approximation algorithms for (multi)-set (multi)-cover and covering integer programs. In Proceedings of 1993 IEEE 34th Annual Foundations of Computer Science, pp. 322–331. https://doi.org/10.1109/SFCS.1993.366855.

34. Rubinstein, R.Y. (1997). Optimization of computer simulation models with rare events. Eur. J. Oper. Res. *99*, 89–112. https://doi.org/10.1016/S0377-2217(96)00385-2.

35. De Boer, P.T., Kroese, D.P., Mannor, S., and Rubinstein, R.Y. (2005). A tutorial on the cross-entropy method. Ann. Oper. Res. *134*, 19–67. https://doi.org/10.1007/S10479-005-5724-Z.

36. Quinlan, J.R. (1986). Induction of decision trees. Mach. Learn. *1*, 81–106. https://doi.org/10.1007/BF00116251.

37. Kozachenko, L.F., and Leonenko, N.N. (1987). Sample estimate of the entropy of a random vector. Probl. Peredachi Infor. *23*, 9–16.

38. Kraskov, A., Stögbauer, H., and Grassberger, P. (2004). Estimating mutual information. Phys. Rev. E Stat. Nonlin. Soft Matter Phys. *69*, 066138. https://doi.org/10.1103/PhysRevE.69.066138.

39. Adams, T.S., Schupp, J.C., Poli, S., Ayaub, E.A., Neumark, N., Ahangari, F., Chu, S.G., Raby, B.A., Deluliis, G., Januszyk, M., et al. (2020). Single-cell RNA-seq reveals ectopic and aberrant lung-resident cell populations in idiopathic pulmonary fibrosis. Sci. Adv. *6*, eaba1983. https://doi.org/10.1126/sciadv.aba1983.

40. Zeisel, A., Muñoz-Manchado, A.B., Codeluppi, S., Lönnerberg, P., La Manno, G., Juréus, A., Marques, S., Munguba, H., He, L., Betsholtz, C., et al. (2015). Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. Science *347*, 1138–1142. https://doi.org/10.1126/SCIENCE.AAA1934/SUPPL_FILE/ZEISEL-SM.PDF.

41. He, S., Wang, L.-H., Liu, Y., Li, Y.-Q., Chen, H.-T., Xu, J.-H., Peng, W., Lin, G.-W., Wei, P.-P., Li, B., et al. (2020). Single-cell transcriptome profiling of an adult human cell atlas of 15 major organs. Genome Biol. *21*, 294. https://doi.org/10.1186/s13059-020-02210-0.

42. Segerstolpe, Å., Palasantza, A., Eliasson, P., Andersson, E.-M., Andréasson, A.C., Sun, X., Picelli, S., Sabirsh, A., Clausen, M., Bjursell, M.K., et al. (2016). Single-cell transcriptome profiling of human pancreatic islets in health and type 2 diabetes. Cell Metab. *24*, 593–607. https://doi.org/10.1016/j.cmet.2016.08.020.

43. Muraro, M.J., Dharmadhikari, G., Grün, D., Groen, N., Dielen, T., Jansen, E., van Gurp, L., Engelse, M.A., Carlotti, F., de Koning, E.J.P., and van Oudenaarden, A. (2016). A single-cell transcriptome atlas of the human pancreas. Cell Syst. *3*, 385–394.e3. https://doi.org/10.1016/j.cels.2016.09.002.

44. Tsoucas, D., Dong, R., Chen, H., Zhu, Q., Guo, G., and Yuan, G.-C. (2019). Accurate estimation of cell-type composition from gene expression data. Nat. Commun. *10*, 2975. https://doi.org/10.1038/s41467-019-10802-z.

45. Lin, J. (1991). Divergence measures based on the Shannon entropy. IEEE Trans. Inf. Theory *37*, 145–151. https://doi.org/10.1109/18.61115.

46. Nogueira, S., Sechidis, K., and Brown, G. (2018). On the stability of feature selection algorithms. J. Mach. Learn. Res. *18*, 1–54.

47. Subramanian, A., Tamayo, P., Mootha, V.K., Mukherjee, S., Ebert, B.L., Gillette, M.A., Paulovich, A., Pomeroy, S.L., Golub, T.R., Lander, E.S., and Mesirov, J.P. (2005). Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. Proc. Natl. Acad. Sci. USA *102*, 15545–15550. https://doi.org/10.1073/pnas.0506580102.

48. Börner, K., Teichmann, S.A., Quardokus, E.M., Gee, J.C., Browne, K., Osumi-Sutherland, D., Herr, B.W., Bueckle, A., Paul, H., Haniffa, M., et al. (2021). Anatomical structures, cell types and biomarkers of the Human Reference Atlas. Nat. Cell Biol. *23*, 1117–1128. https://doi.org/10.1038/s41556-021-00788-6.

49. Coulombe, P.A., and Wong, P. (2004). Cytoplasmic intermediate filaments revealed as dynamic and multipurpose scaffolds. Nat. Cell Biol. *6*, 699–706. https://doi.org/10.1038/ncb0804-699.

50. Saha, S.K., Kim, K., Yang, G.M., Choi, H.Y., and Cho, S.G. (2018). Cytokeratin 19 (KRT19) has a role in the reprogramming of cancer stem cell-like cells to less aggressive and more drug-sensitive cells. Int. J. Mol. Sci. *19*, E1423. https://doi.org/10.3390/IJMS19051423.

51. Kubo, F., Ariestanti, D.M., Oki, S., Fukuzawa, T., Demizu, R., Sato, T., Sabirin, R.M., Hirose, S., and Nakamura, N. (2019). Loss of the adhesion G-protein coupled receptor ADGRF5 in mice induces airway inflammation and the expression of CCL2 in lung endothelial cells 11 Medical and Health Sciences 1102 Cardiorespiratory Medicine and Haematology. Respir. Res. *20*, 11–21. https://doi.org/10.1186/S12931-019-0973-6/FIGURES/11.

52. Vazquez, B.N., Laguna, T., Carabana, J., Krangel, M.S., and Lauzurica, P. (2009). CD69 gene is differentially regulated in T and B cells by evolutionarily conserved promoter-distal elements. J. Immunol. *183*, 6513–6521. https://doi.org/10.4049/JIMMUNOL.0900839.

53. Ziegler, S.F., Ramsdell, F., and Alderson, M.R. (1994). The activation antigen CD69. Stem Cell. *12*, 456–465. https://doi.org/10.1002/STEM.5530120502.

54. Plešingerová, H., Janovská, P., Mishra, A., Smyčková, L., Poppová, L., Libra, A., Plevová, K., Ovesná, P., Radová, L., Doubek, M., et al. (2018). Expression of COBLL1 encoding novel ROR1 binding partner is robust predictor of survival in chronic lymphocytic leukemia. Haematologica *103*, 313–324. https://doi.org/10.3324/HAEMATOL.2017.178699.

55. Castro, C.D., and Flajnik, M.F. (2014). Putting J-chain back on the map: how might its expression define plasma cell development? J. Immunol. *193*, 3248–3255. https://doi.org/10.4049/JIMMUNOL.1400531.

56. De Plaen, I.G., Han, X.B., Liu, X., Hsueh, W., Ghosh, S., and May, M.J. (2006). Lipopolysaccharide induces CXCL2/macrophage inflammatory protein-2 gene expression in enterocytes via NF-kappaB activation: independence from endogenous TNF-alpha and platelet-activating factor. Immunology *118*, 153–163. https://doi.org/10.1111/J.1365-2567.2006.02344.X.

57. Robertson, M.J. (2002). Role of chemokines in the biology of natural killer cells. J. Leukoc. Biol. *71*, 173–183.

58. Molleran Lee, S., Villanueva, J., Sumegi, J., Zhang, K., Kogawa, K., Davis, J., and Filipovich, A.H. (2004). Characterisation of diverse PRF1 mutations leading to decreased natural killer cell activity in North American families with haemophagocytic lymphohistiocytosis. J. Med. Genet. *41*, 137–144. https://doi.org/10.1136/JMG.2003.011528.

59. Valés-Gómez, M., Esteso, G., Aydogmus, C., Blázquez-Moreno, A., Marín, A.V., Briones, A.C., Garcillán, B., García-Cuesta, E.M., López Cobo, S., Haskologlu, S., et al. (2016). Natural killer cell hyporesponsiveness and impaired development in a CD247-deficient patient. J. Allergy Clin. Immunol. *137*, 942–945.e4. https://doi.org/10.1016/J.JACI.2015.07.051.

60. Vanderbilt, J.N., Mager, E.M., Allen, L., Sawa, T., Wiener-Kronish, J., Gonzalez, R., and Dobbs, L.G. (2003). CXC chemokines and their receptors are expressed in type II cells and upregulated following lung injury. Am. J. Respir. Cell Mol. Biol. *29*, 661–668. https://doi.org/10.1165/RCMB.2002-0227OC.

61. Shi, Y., Zhao, Y., Zhang, Y., Aierken, N., Shao, N., Ye, R., Lin, Y., and Wang, S. (2018). AFF3 upregulation mediates tamoxifen resistance in breast cancers. J. Exp. Clin. Cancer Res. *37*, 254. https://doi.org/10.1186/S13046-018-0928-7.

62. Maher, K., Konjar, S., Watts, C., Turk, B., and Kopitar-Jerala, N. (2014). Cystatin F regulates proteinase activity in IL-2-activated natural killer cells. Protein Pept. Lett. *21*, 957–965. https://doi.org/10.2174/0929866521666140403124146.

63. Ronchetti, S., Ricci, E., Petrillo, M.G., Cari, L., Migliorati, G., Nocentini, G., and Riccardi, C. (2015). Glucocorticoid-induced tumour necrosis factor receptor-related protein: a key marker of functional regulatory T cells. J. Immunol. Res. *2015*, 171520. https://doi.org/10.1155/2015/171520.

64. Fu, Y., Huang, X., Zhang, P., van de Leemput, J., and Han, Z. (2020). Single-cell RNA sequencing identifies novel cell types in Drosophila blood. J. Genet. Genomics Yi Chuan Xue Bao *47*, 175–186. https://doi.org/10.1016/j.jgg.2020.02.004.

65. Shekhar, K., and Menon, V. (2019). Identification of cell types from single-cell transcriptomic data. In Computational Methods for Single-Cell Data Analysis Methods in Molecular Biology, G.-C. Yuan, ed. (Springer), pp. 45–77. https://doi.org/10.1007/978-1-4939-9057-3_4.

66. Wilkerson, B.A., Zebroski, H.L., Finkbeiner, C.R., Chitsazan, A.D., Beach, K.E., Sen, N., Zhang, R.C., and Bermingham-McDonogh, O. (2021). Novel cell types and developmental lineages revealed by single-cell RNA-seq analysis of the mouse crista ampullaris. Elife *10*, e60108. https://doi.org/10.7554/eLife.60108.

67. Wu, H., Kirita, Y., Donnelly, E.L., and Humphreys, B.D. (2019). Advantages of single-nucleus over single-cell RNA sequencing of adult kidney: rare cell types and novel cell states revealed in fibrosis. J. Am. Soc. Nephrol. *30*, 23–32. https://doi.org/10.1681/ASN.2018090912.

68. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: machine learning in Python. J. Mach. Learn. Res. *12*, 2825–2830.

69. Johnson, J., Douze, M., and Jégou, H. (2017). Billion-scale similarity search with GPUs. Preprint at arXiv. https://doi.org/10.48550/arXiv.1702.08734.

70. Johnson, D.S. (1974). Approximation algorithms for combinatorial problems. J. Comput. Syst. Sci. *9*, 256–278. https://doi.org/10.1016/S0022-0000(74)80044-9.

71. Chvatal, V. (1979). A greedy heuristic for the set-covering problem. Math. Oper. Res. *4*, 233–235.

72. Rubinstein, R. (1999). The cross-entropy method for combinatorial and continuous optimization. Methodol. Comput. Appl. Probab. *1*, 127–190. https://doi.org/10.1023/A:1010091220143.

73. Welch, B.L. (1947). The generalisation of student's problems when several different population variances are involved. Biometrika *34*, 28–35. https://doi.org/10.1093/biomet/34.1-2.28.

74. Wolf, F.A., Angerer, P., and Theis, F.J. (2018). SCANPY: large-scale single-cell gene expression data analysis. Genome Biol. *19*, 15. https://doi.org/10.1186/s13059-017-1382-0.

75. Kullback, S., and Leibler, R.A. (1951). On information and sufficiency. Ann. Math. Statist. *22*, 79–86. https://doi.org/10.1214/aoms/1177729694.

76. Chen, E.Y., Tan, C.M., Kou, Y., Duan, Q., Wang, Z., Meirelles, G.V., Clark, N.R., and Ma'ayan, A. (2013). Enrichr: interactive and collaborative HTML5 gene list enrichment analysis tool. BMC Bioinf. *14*, 128. https://doi.org/10.1186/1471-2105-14-128.

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited data		
Idiopathic Pulmonary Fibrosis (IPF)	Adams et al.[39]	GEO: GSE136831
Mouse Cortex (MC)	Zeisel et al.[40]	GEO: GSE60361
Human Cell Atlas (HCA)	He et al.[41]	GEO: GSE159929
Software and algorithms		
Multiset multicover algorithm	This paper	Zenodo: https://doi.org/10.5281/zenodo.7158750
Phenotype cover algorithms (G-PC, CEM-PC)	This paper	Zenodo: https://doi.org/10.5281/zenodo.7158780
Experiments in this paper	This paper	Zenodo: https://doi.org/10.5281/zenodo.7158788
scGeneFit algorithm	Dumitrascu et al.[25]	GitHub: https://github.com/solevillar/scGeneFit-python
Decision Trees, ANOVA F-values, Mutual Information, Logistic Regression	Pedregosa et al.,[68] scikit-learn	Zenodo: https://doi.org/10.5281/zenodo.6968622
T-test for differentially expressed genes	Theis Lab; PI: Fabian Theis	GitHub: https://github.com/theislab/diffxpy
RankCorr algorithm	Vargo et al.[26]	GitHub: https://github.com/ahsv/RankCorr
mRMR algorithm	Peng et al.[29]	GitHub: https://github.com/smazzanti/mrmr
Approximate nearest neighbors algorithm	Johnson et al.[69]	GitHub: https://github.com/facebookresearch/faiss
GSEA algorithm	Subramanian et al.,[47] GSEApy	Zenodo: https://doi.org/10.5281/zenodo.3748084
Python version 3.8	Python Software Foundation	https://www.python.org/

RESOURCE AVAILABILITY

Lead contact

Further information and requests for resources should be directed to and will be fulfilled by the lead contact, Ziv Bar-Joseph (zivbj@andrew.cmu.edu).

Materials availability

This study did not generate new unique reagents.

Data and code availability

- This paper analyzes existing, publicly available data. These accession numbers for the datasets are listed in the key resources table.
- We implemented a general-purpose package for running the greedy multiset multicover algorithm in C++ and expose it to Python. The code has been deposited at https://github.com/euxhenh/multiset-multicover. The G-PC and CEM-PC algorithms for feature selection can be found at https://github.com/euxhenh/phenotype-cover. Installation instructions are available in each repository. The code for running experiments in this paper is available from https://github.com/euxhenh/phenotype-cover-experiments. DOIs are listed in the key resources table.
- Any additional information required to reanalyze the data reported in this paper is available from the lead contact upon request.

METHOD DETAILS

Notation

Let $\mathbf{M} \in \mathbb{R}^{P \times F}$ represent a score matrix. We denote by P the number of phenotypes (e.g., cell types) and by F the number of features (e.g., genes). In this paper, we use scRNA-seq read count data denoted by $\mathbf{X} \in \mathbb{R}_{\geq 0}^{N \times G}$. Here, N denotes the number of cells and G denotes the number of genes. Given a known vector $\boldsymbol{y}$ of length N representing class labels, we derive a matrix $\mathbf{M}$ from $\mathbf{X}$ by averaging expression values of cells with the same class label. In this case, P = {number of distinct classes} and $F = G$. We denote by $[n]$ the set $\{1, 2, \ldots, n\}$. Finally, let $(x)^+ = \max\{x, 0\}$.

Problem formulation and complexity

Phenotype cover (PC)

Given a score (signature) matrix $\mathbf{M} \in \mathbb{R}^{P \times F}$, find a subset $\mathcal{S} \subset [F]$ of minimal cardinality, such that for every $i, j \in [P]$ with $i \neq j$, and some fixed positive K, the following holds

$$\sum_{s \in \mathcal{S}} (\mathbf{M}_{i,s} - \mathbf{M}_{j,s})^+ \geq K$$

PC is asking for a small subset of features such that for *any* given ordered pair of phenotypes (i,j), one can find enough features which collectively distinguish i from j by a factor of at least K. This problem allows the selection of a gene which could cover several phenotypic pairs, e.g., multiple cell subtypes vs another major cell type, but also demands sufficient coverage between subtypes themselves. The straightforward solution of iterating over all possible feature subsets satisfying the requirements above and selecting the one with the smallest cardinality, suffers from an exponential complexity in the number of subsets considered. In fact, PC is equivalent to multiset multicover which is NP-complete.[33]

To establish this equivalence, it may help to first consider a simplified version of the problem where we restrict $\mathbf{M}$ to be binary and $K = 1$; call this problem PC-B. In this case, we require a small subset of features $\mathcal{S}$, such that for any two phenotypes $i \neq j$ there exists some index $s \in \mathcal{S}$ where $\mathbf{M}_{i,s} - \mathbf{M}_{j,s} = 1$. Note that in this simplified form, every feature s induces a bipartite graph $G_s = (u_s, v_s, \varepsilon_s)$, where

$$u_s = \{i | \mathbf{M}_{i,s} = 1\},\ v_s = \{j | \mathbf{M}_{j,s} = 0\} = [P] \backslash u_s$$

Every edge $e \in \varepsilon_s$ corresponds to an ordered pair of phenotypes (Figure 1).

Now, given the collection of sets $\varepsilon = \{\varepsilon_s | s \in [F]\}$, set cover asks to find the smallest subset $\varepsilon_{sol} \subset \varepsilon$ such that for every element $e \in \cup\ \varepsilon_s$, there exists a set in ε_{sol} which contains e. It is easy to see that the features corresponding to ε_{sol} are the solution to PC-B.

So far, we only considered a binary score matrix. However, a solution to the binary problem can be naturally extended to solve non-binary scoring matrices by assigning multiplicities to the elements of ε_s. To every $e = (i,j)$ we assign the multiplicity $(\mathbf{M}_{i,s} - \mathbf{M}_{j,s})^+$ and view ε_s as a *multiset*. Note that since we are working with real numbers, we need to round the multiplicities to integers. Higher precision can be easily obtained by first scaling both $\mathbf{M}$ and K by some scalar c and performing the rounding after. Finally, the requirement $K = 1$ can also be relaxed by solving for a *multicover*, where we require each element to be contained at least K times in $\cup\ \varepsilon_{sol}$ (counting multiplicities).

Approximating a solution to phenotype cover

Given the NP-Completeness of PC, we present two greedy solutions that run in polynomial time.

Greedy phenotype cover (G-PC)

First, we consider the well-known greedy approach to solving set cover that iteratively picks the set which covers the greatest number of elements not covered yet.[70,71] The algorithm can be trivially extended to solve multiset multicover.[33] The full algorithm is presented in Methods S1, algorithms 1 and 2. Every time we select a set, we need to correct the multiplicities of all the remaining $O(F)$ sets, each of which may contain up to $O(P^2)$ elements (all phenotypic pairs). Therefore, if we denote the solution size by k, the run-time complexity of G-PC is $O(kP^2F)$. In practice, P is small and $k \ll F$, therefore, the method is almost linear in the number of features considered. The approximation accuracy for this solution was previously analyzed and it was shown that the greedy algorithm for multiset multicover is upper bounded by a factor of H_m increase in the solution size, where $H_m = 1 + \frac{1}{2} + \ldots + \frac{1}{m} \leq \log(m) + 1$ and m is the cardinality of the largest multiset.[33]

Cross-entropy method phenotype cover (CEM-PC)

In addition to the greedy multiset multicover approach, we developed a new method based on cross-entropy (CEM).[34] CEM was originally used to estimate probabilities of rare events and it was later extended to solve combinatorial problems.[72] Roughly, CEM consists of two steps: 1) generate a random sample based on a specific distribution, and 2) update distribution parameters such that "high-scoring" samples are more likely to be produced in the next iteration. This two-step procedure is repeated until convergence, or until a maximal number of iterations is reached. The final parameters determine the solution to the combinatorial problem (in our case, selecting features whose probability is greater than some threshold). For a more detailed analysis of CEM, the reader may refer to the excellent tutorial of De Boer et al.[35]

We present a variant of CEM for solving set cover by introducing a scoring function that encourages high coverage but penalizes a large number of features (Methods S1, alg. 3). The run-time complexity of CEM-PC depends on the maximum number of iterations I, the number of random samples per iteration R_s, and the complexity of the scoring function (in this case, the smallest coverage attained per random sample). This leads to a total run-time complexity of $O(IR_sP^2F)$. In this paper, we use I = 500 and R_s = 1000. In practice, convergence is attained in fewer iterations.

Baselines

As mentioned above, several prior methods have been developed for marker and feature selection. We thus compared our method against several baselines on traditional supervised learning tasks, ability to construct signature matrices for deconvolution of bulk mixtures, and feature stability. Specifically, we compare our method to scGeneFit[25] and RankCorr[26] which were used for discriminative marker selection. We use the implementations provided by the authors of each method. For scGeneFit, we used a redundancy of 0.1 and kept the remaining parameters at defaults. We compare against an embedded method that uses decision trees with the Gini Index criterion to rank features. Note that here we use decision trees as a feature selection method and not as a classifier. The performance of decision trees as a classifier was worse than that of Logistic Regression using the same features, hence, we excluded these results from the manuscript. We also compare against several other filter methods. We consider the union of the top differentially expressed genes per phenotype as determined by Welch's t-test[73] (TopDE). We compare against ReliefF[28] which uses nearest neighbors' information to update feature weights. Since computing exact neighbors is slow for the single cell data we are using, we developed a variant of ReliefF that uses approximate neighbors based on the faiss package.[69] We compute 30 neighbors per sample. ANOVA F-values and mutual information between gene expression and phenotype are also computed using the popular package scikit-learn.[68] Finally, we compare against minimum-redundancy-maximum-relevance (mRMR).[29] For mRMR, we use the open-source Python package mrmr (https://github.com/smazzanti/mrmr) which measures relevance via the F-value and measures redundancy via Pearson's correlation. For all the baselines but TopDE and RankCorr, we take the top k scoring features, where k equals the size of the solution returned by G-PC.

Datasets and preprocessing

We use three public scRNA-seq datasets to validate our method (Table 1). For all three datasets we remove classes with less than 50 cells. This leads to 75 tissue/cell type pairs for HCA. We also filter for genes expressed in at least 10 cells, and for runtime efficiency purposes, we only consider highly variable genes for IPF and HCA for all methods. Also, scGeneFit was slow for MC, so we considered only highly variable genes for MC when running this method. Each dataset is normalized using Scanpy[74] so that the total counts for all cells are equal. The data is then log(x+1) transformed and each feature scaled to unit variance and zero mean. scGeneFit performed very poorly when the data was scaled, hence, for a fair comparison we skipped the scaling step when running scGeneFit. Log-transforming and scaling the data had a positive effect on the F1 score for all the other methods. We show these results for the MC dataset in Figure S4D. On the other hand, deconvolution via CIBERSORT works best if the data is in linear space as recommended by the authors, hence, we did not log the data during deconvolution. Feature selection, however, is applied on logged data.

We split all datasets into a train and test set of equal size in a stratified fashion. To obtain a signature matrix **M** for G-PC and CEM-PC, we average expression values for every phenotype. While it is true that this operation summarizes the data and leads to information loss, we note that our goal is not reconstruction or dimensionality reduction but rather marker selection. We argue that for such a task the individual cell-based expression is less important since we are looking for markers that are generally observed across most or all cells. Furthermore, commonly used DE tests such as t-test also rely on a small set of sufficient statistics.

Regarding the choice of K, in this paper we test the performance of our methods across multiple values of K. In practice, a single value for K could be obtained in a cross-validation fashion.

QUANTIFICATION AND STATISTICAL ANALYSIS

To compare the performance of Logistic Regression classifiers, we use the macro-average F1 score. This score equally weighs the F1 score of each class, which is desirable as we are interested in finding markers for all phenotypes, regardless of any class imbalance in the data. For a single class p, the F1 score is the harmonic mean between precision and recall

$$\mathrm{F1}_p = \frac{2}{1/\mathrm{Precision}_p + 1/\mathrm{Recall}_p} = 2\frac{\mathrm{Precision}_p \cdot \mathrm{Recall}_p}{\mathrm{Precision}_p + \mathrm{Recall}_p}$$

The macro-average F1 score is simply the unweighted mean of per-class F1 scores

$$\mathrm{F1}_{\mathrm{macro}} = \frac{1}{P}\sum_{p=1}^{P} \mathrm{F1}_p$$

To evaluate deconvolution performance, we use the Jensen-Shannon divergence[45] which is a symmetric measure between two probability distributions. Given two discrete probability distributions P and Q, the Kullback-Leibler divergence[75] is given by

$$\mathrm{KL}(P\|Q) = \sum_{x \in \chi} P(x) \log\left(\frac{P(x)}{Q(x)}\right)$$

where χ is a probability space. Letting $M = \frac{1}{2}(P + Q)$, the Jensen-Shannon divergence is

$$\mathrm{JS}(P\|Q) = \frac{1}{2}\mathrm{KL}(P\|M) + \frac{1}{2}\mathrm{KL}(Q\|M)$$

Feature stability computes the average size of the overlap divided by the size of the union for all pairs of feature sets. More precisely, given a collection of feature sets $\varepsilon = \{S_1, \ldots S_k\}$, stability is given by

$$s = \frac{2}{k(k-1)} \sum_{i=1}^{k} \sum_{j>i}^{k} \frac{|S_i \cap S_j|}{|S_i \cup S_j|}$$

Finally, we performed gene set enrichment analysis (GSEA) using the Python package GSEApy (https://gseapy.readthedocs.io/) and the Enrichr API.[76] We used the *HuBMAP_ASCTplusB_augmented_2022* gene set.[48] All p values reported in this paper were corrected for multiple testing.