EMT Multi-Omics

Deconvolution of proteomics data using scRNA-Seq

Ahmed Youssef

October 27, 2021

Contents

Introduction

The fundamental unit of all living organisms is the cell, and recent technological advances have granted us unprecedented opportunities to study life at this principal level. Proteins, through their networks of interactions, carry out most of the vital biological processes governing cellular functions, yet remain largely unexplored in the single-cell space, representing crucial gaps in our knowledge of cell biology. While single-cell proteomics methods are still in their infancy, single-cell RNA sequencing (scRNA-Seq) has emerged in recent years as a powerful technology for defining cell states on a large scale, enabling breakthroughs in many areas of cell biology research, and begging the question of whether it can be used for making inferences at the protein level. **In this report, I explore the deconvolution of bulk proteomics data to the single-cell level using scRNA-Seq data.**

Experiment summary

Epithelial-to-mesenchymal transition (EMT) is a biological process in which epithelial cells gradually lose their adhesion and transition into mesenchymal cells. As one of the hallmarks of cancer progression, it is one of the long-standing interests of the biomedical research community. Towards profiling this process, protein and RNA samples were extracted from cells at 8 different timepoints during EMT and multiple layers of omics data were generated. These omics layers include proteomics, transcriptomics, phosphoproteomics, secretome, exosome among others. A pre-print with more details on the experiment and generated data can be found on bioRxiv here (Paul et al, 2021). This report is interested in the scRNA-Seq, microarray, and proteomics datasets generated in this study.

Approach

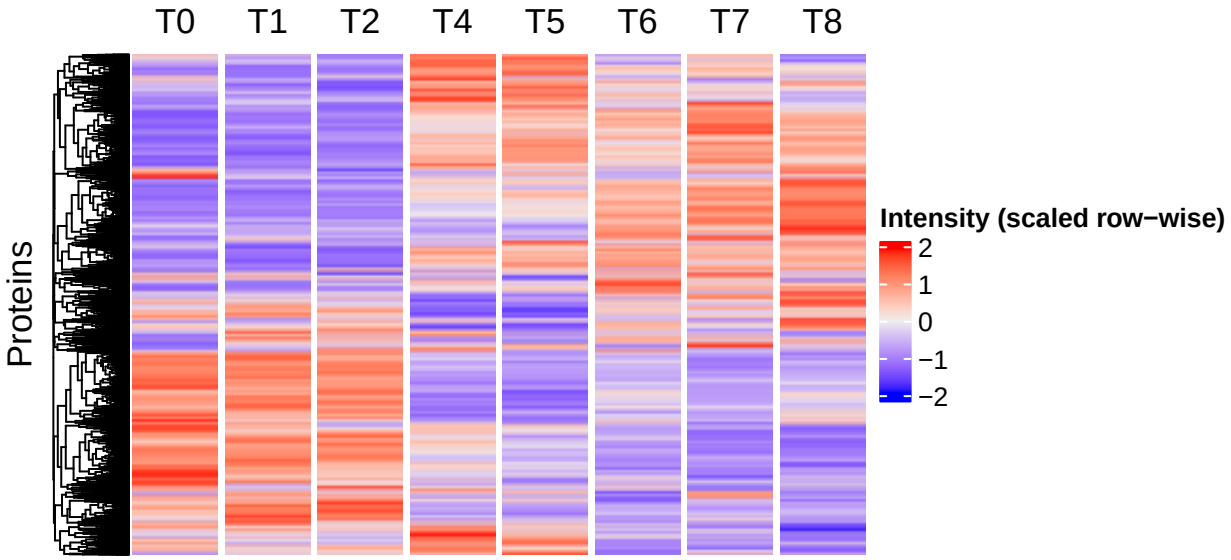
Bulk proteomics data gives a view of the aggregated protein abundance from all cell types within a sequenced sample. Using single-cell data, derived from the same samples, we can investigate the sample heterogeneity by estimating proportions of cell types within the bulk sample. We cannot reliably use these proportions to directly estimate the contribution of each population to each gene/protein's expression at the bulk-level however, since there is low correlation between RNA and protein levels of the same genes due to multiple biological factors, such as alternative splicing and post-translational modifications. Leveraging the timepoints present in this dataset, which conveniently show shifts in cell type abundances across time, we can instead look for changes in cell-type proportions and corresponding changes in bulk-level protein abundance as suggestive of relationships between specific cell types and specific proteins. This information can then potentially be used to estimate the contribution of individual cell types to the bulk proteomics measurements.

Data summary - Proteomics

The bulk proteomics data was generated in the Emili Lab using standard mass-spectrometry. Summary of the dataset follows:

- 6,967 proteins
- 10 different timepoints
- Three replicates

The average intensity across replicates was computed for each protein in each timepoint. Timepoints 3 and 9 were removed since they are not present in the scRNA data. The data was normalized such that each timepoint sums to 1.



Data summary - scRNA-Seq

The bulk proteomics data was generated in the Emili Lab using standard mass-spectrometry. Summary of the dataset follows:

- 9,785 genes
- 1,913 cells (~200 cells per timepoint)
- 8 different timepoints

Prior to this summation, genes with zero variance as well as those with non-zero counts in less than 5% of all cells were removed. This removed 17 genes (0.2% of all genes). The data was also normalized such that each cell sums to 1.

Table 1: Number of cells in each timepoint

T0	T1	T2	T4	T5	T6	T7	T8
367	208	246	190	247	217	236	202

Data summary - Bulk mRNA

The bulk mRNA data comes from a microarray experiment. Summary of the dataset follows:

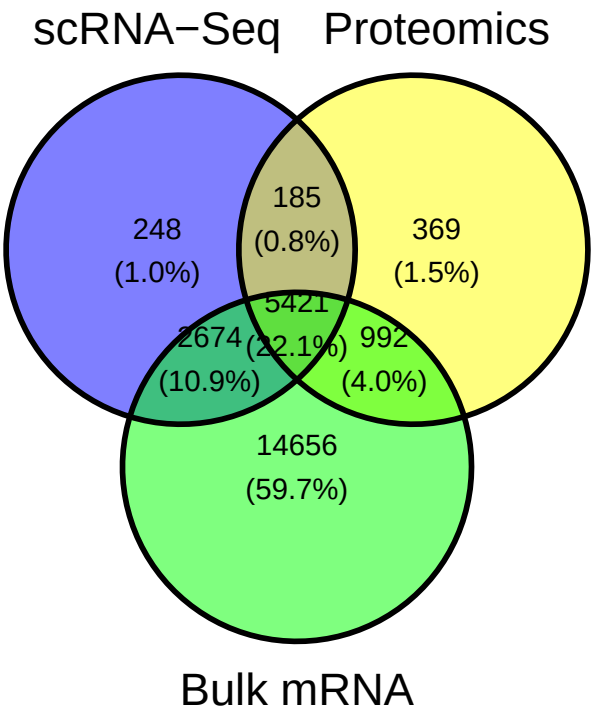
- 23,743 genes

- 10 different timepoints
- Three replicates

The average intensity across replicates was computed for each protein in each timepoint. Timepoints 3 and 9 were removed since they are not present in the scRNA data. The data was normalized such that each timepoint sums to 1.

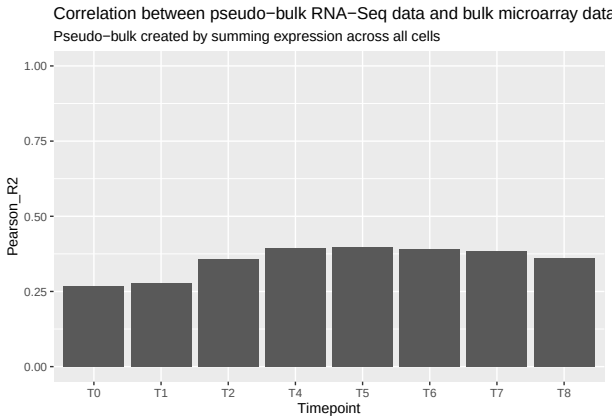
Protein overlap

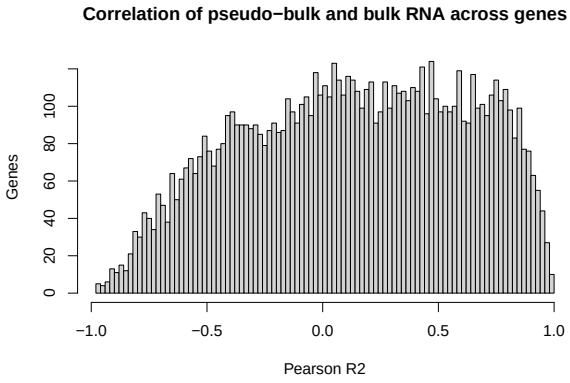
The venn diagram below shows the overlap of the identified proteins in the datasets.



Pseudo-bulk RNA vs bulk RNA

A pseudo-bulk RNA dataset is created for each timepoint by summing the gene counts of all cells within the timepoint. This pseudo-bulk data is then compared to the actual microarray bulk mRNA data present for each timepoint. The cross-timepoint measurements for each genes were also correlated. The distributions of these correlations are showed below.



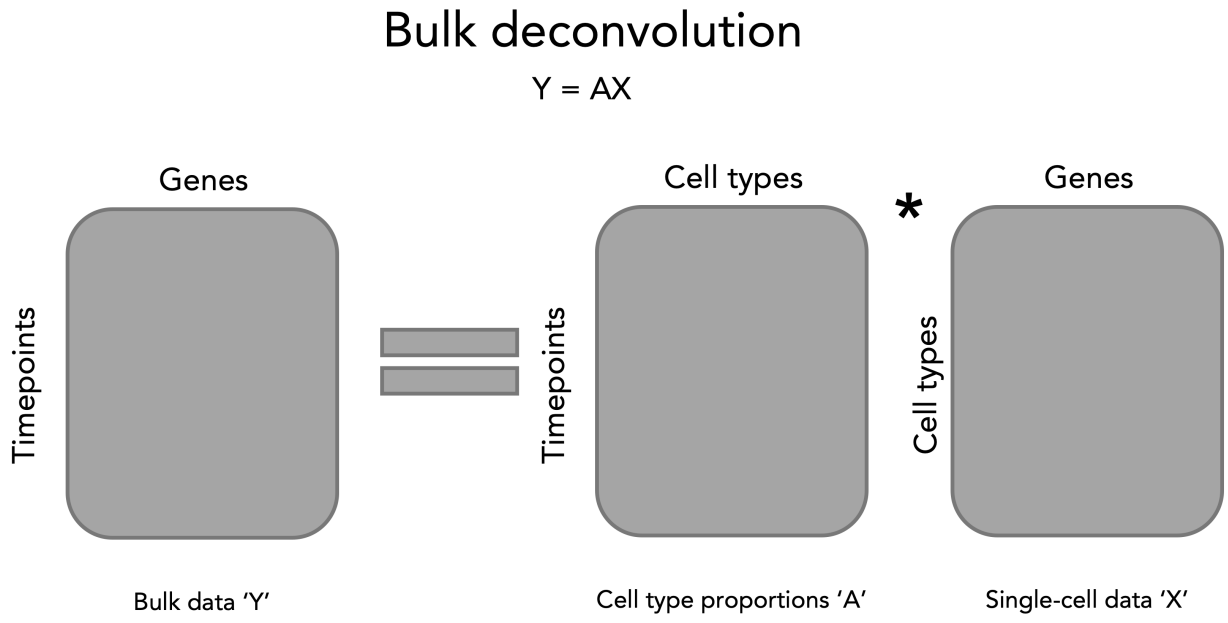


Method Development and Optimization

To investigate the ability to recover the scRNA data from the pseudo-bulk data, we start by clustering the scRNA data to identify the cell clusters followed by creating the following matrices:

- Matrix A of dimensions $timepointsXclusters$. (cell type proportions)
- Matrix X of dimensions $clustersXgenes$. (cluster-averaged single-cell RNA data)
- Matrix Y of dimensions $timepointsXgenes$. (pseudo-bulk RNA data)

We then attempt to re-create the single-cell matrix X data by computing $Y = AX'$.

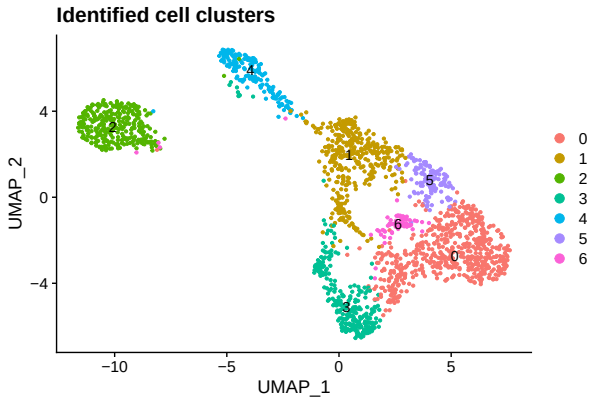


Identify cell clusters

The cell states in our dataset are identified in an unsupervised manner based on similarity of gene expression profiles. All cells from all timepoints are pooled together for this analysis.

For data pre-processing, we remove the genes with low expression counts, retaining genes with a minimum of 3 counts in at least 3 cells. This removed 1,240 genes (13% of all genes). On average, each cell expressed ~3,600 genes after processing. Seurat is then used to cluster the cells based on the 2,000 most variable genes.

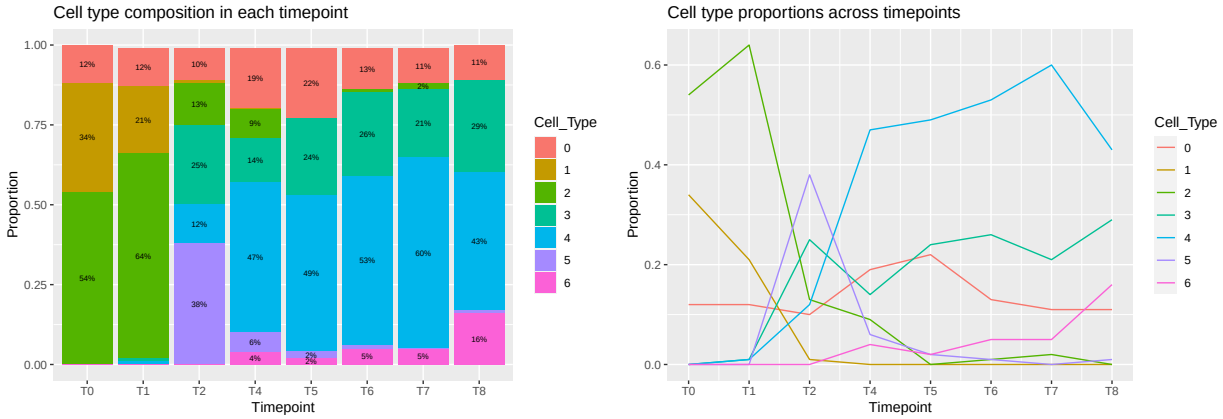
The initial model defines 7 cell clusters at a Seurat resolution of 0.4. We intentionally begin with a number of clusters less than the number of timepoints to avoid creating an undetermined problem. The below UMAP plot visualizes the identified cell clusters.



Changes in cell state composition over time

Since this dataset is investigating cells undergoing EMT, the cell population abundances are changing over time. The below figures visualize these proportion changes.

Note: this approach assumes the single-cell data accurately captures the sample heterogeneity. In practice, biased cell sampling upstream could lead to an inaccurate view of sample heterogeneity here.

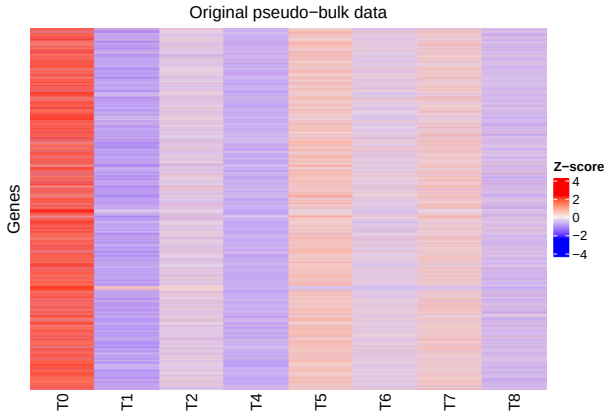


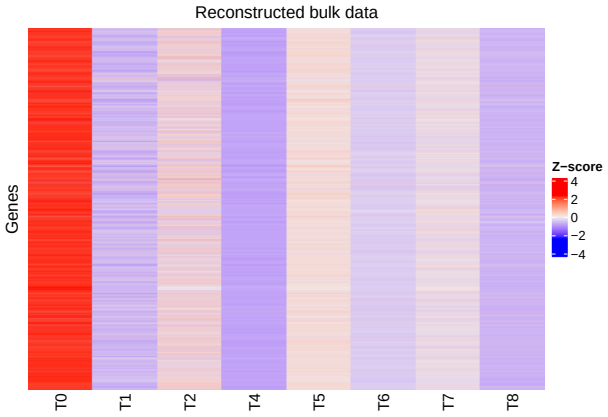
Re-construct pseudo-bulk data from single-cell data

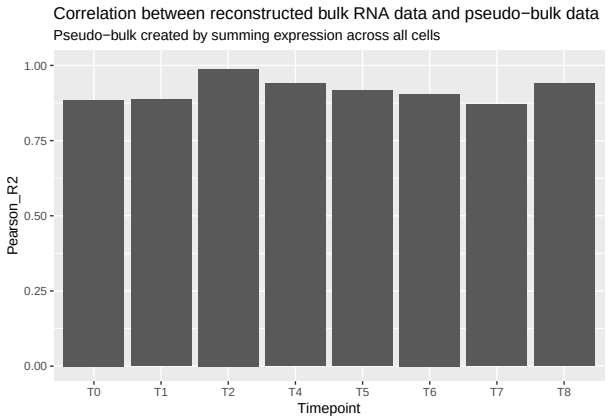
We now attempt to re-create the pseudo-bulk data (Y) from the single-cell data using the simple formula $Y' = AX$. The matrix X was constructed by averaging the gene expression of each cluster. The matrix A contains the number of cells from each cell cluster in each timepoint.

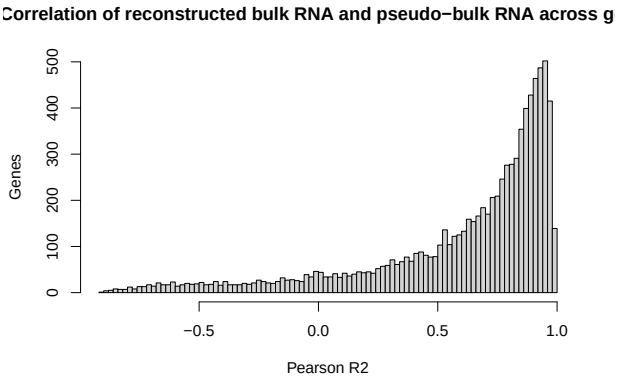
The below heatmaps show the pseudo-bulk data and the re-constructed data side-by-side, with the 8,528 genes in the same order. The heatmaps are scaled by row for visualization purposes. This reconstructed bulk

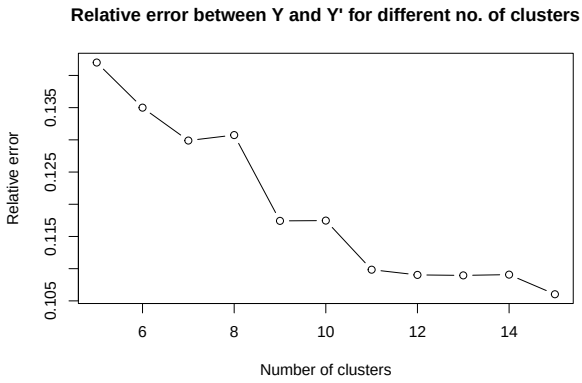
data is then compared to the pseudo-bulk data in each timepoint. Correlations were also computed across the cross-timepoint measurements for each gene. The distributions of these correlations are shown below.









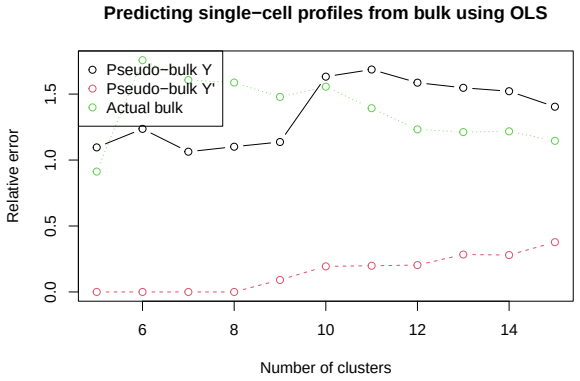


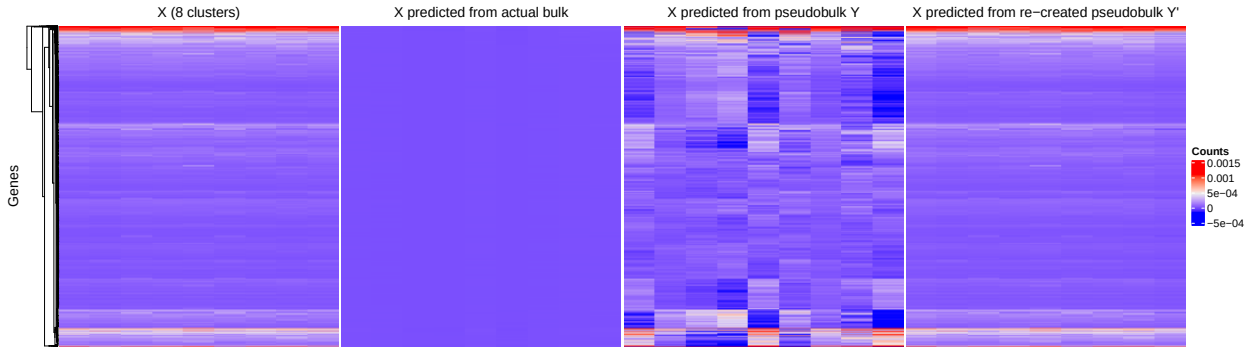
Re-construct single-cell data from pseudo-bulk data

In this section, we attempt to re-create the single-cell data from the pseudo-bulk data and the timepoint-specific cell cluster counts. Based on the formula $Y = AX$ outlined in previous sections, given the pseudobulk matrix Y and the timepoint-specific cell counts 'mixing' matrix A, we aim to compute X using the formula $X' = (A^T A)^{-1}(A^T Y)$, which is essentially the pseudo-inverse of A multiplied by the pseudo-bulk Y. Recall that the pseudo-bulk is computed by summing up the counts of individual cells in each timepoint.

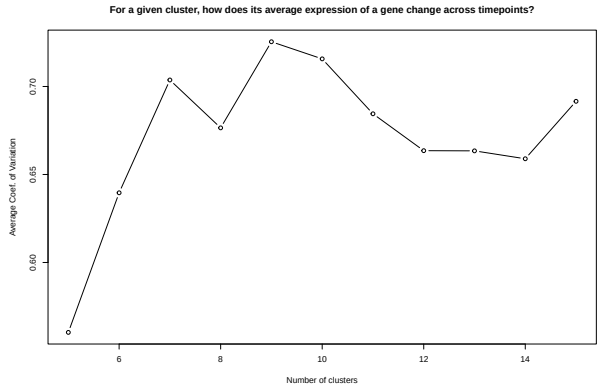
We vary the number of cell clusters by varying the *resolution* parameter in Seurat's clustering algorithm. We try this method for a number of clusters ranging between 5-15. For each given number of clusters, we solve the formula above to predict X'. We then compare this to the actual observed X from the scRNA-Seq data, which is the per-cluster average of gene counts. The errors are reported as relative RMAD (relative mean absolute deviation) using the formula $|X - X'|/X$, where $|X|$ is the absolute value of the difference. This error is computed for each gene and the final reported score is the average RMAD value, in other words: *on average, how different is a gene's predicted values compared to the true ones?*

This process above is repeated once using Y' as the bulk data, where $Y' = AX$ is computed first prior to computing X' as above, and once using the actual bulk data in this dataset. The resultant cluster-RMAD relationships are shown in the below plot.





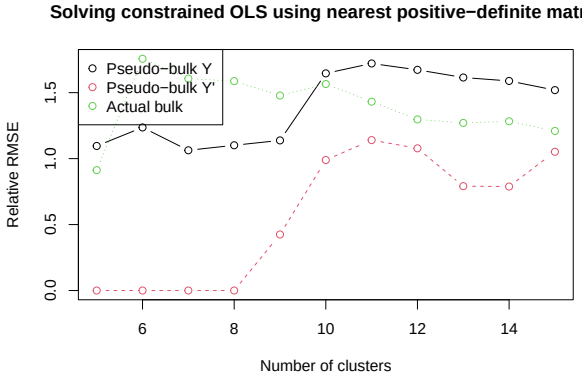
Variation of gene expression within cell type across timepoints

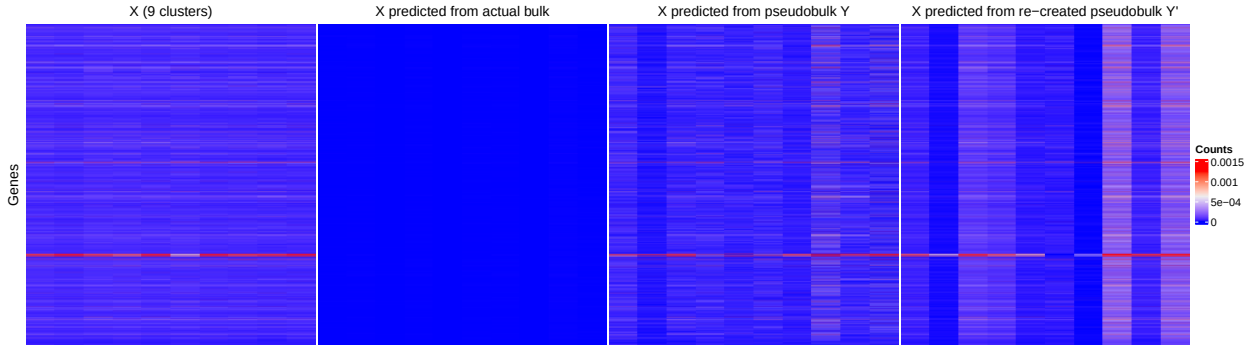


Constraining the solution to be non-negative

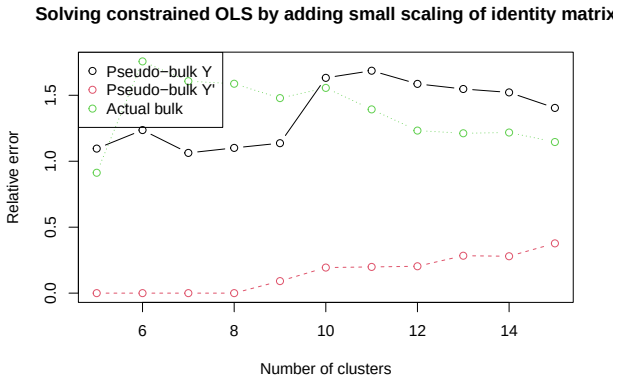
The Tsoucas et al publication proposed a strategy for RNA deconvolution based on the OLS formulation we used in this report, with the addition of constraining the solution to be non-negative since there are no possible negative gene counts, and increasing the weight of cell types with low average expression levels. Since we typically have more cell types than time points, our problem is underdetermined and as such there are multiple possible solutions. In this section, we follow their strategy for constraining the solution to be non-negative.

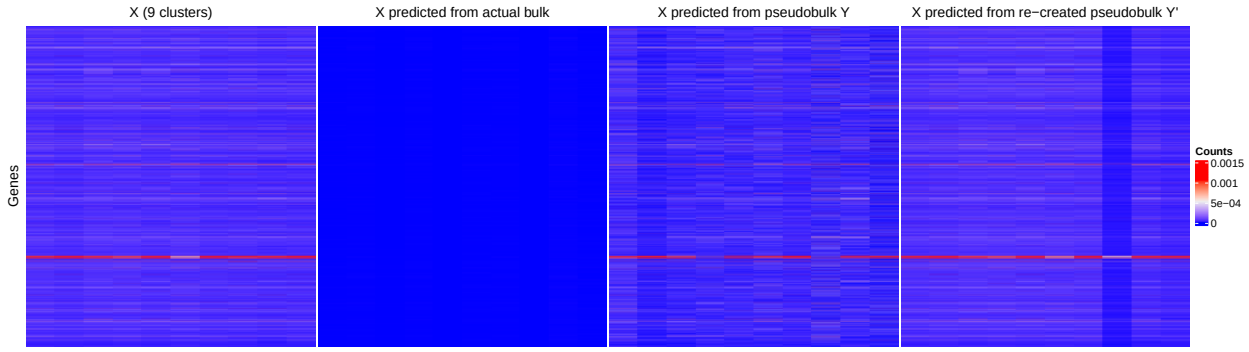
We solve $\hat{X} = min_{x>=0}||AX - Y||^2$ which algebraically is the same as solving $\hat{X} = min_{x>=0}(-2Y^TAX + X^TA^TAX)$. The R function *solve.QP* from the *quadprog* package is used to solve this equation one gene at a time to estimate the expression profile at the cell-type level. This algorithm, however, expects a positive definite matrix. The matrix A^TA is positive definite when A has more rows than columns, i.e. when the problem is a least-squares problem. When we have more clusters than timepoints then A has more columns than rows, and so the matrix is not invertible, and by extension not positive definite. As a workaround, we use the function *nearPD* to find the nearest positive matrix instead.





As an alternative to the *nearPD* function, we add a small scaling of the identity matrix (10^{-9}) to A^TA, which will make it positive definite and as such allow *solve.QP* to work. The result is shown below.





Predicting single-cell profiles using ridge regression

The condition number of $A^T A$ for the case where we have 8 single-cell clusters, i.e. as many clusters as there are timepoints, is noticeably high (15,828) compared to a smaller number of clusters which suggests that it is a near-singular matrix. This would explain the relatively high error in estimating the single-cell profiles. In an attempt to address this, we use ridge regression to estimate the single-cell profiles instead.

Ridge regression is similar to linear regression but with the addition of a regularization term λ. A range of values for λ are tested to decide on the optimal one. The R package *glmnet* is used for this task. We solve the problem by fitting a ridge regression model to each gene. The full procedure is outlined below:

Input: Number of clusters N, Single-cell expression matrix SC

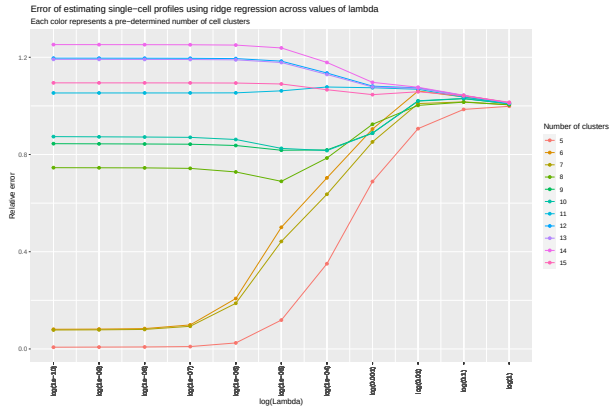
Procedure:

1. Cluster single-cell matrix SC *into* N clusters using *Seurat*
2. Construct cluster * gene matrix $\boldsymbol{X}$ by computing the average expression of each gene in each cluster
3. Construct timepoint * cluster mixing matrix $\boldsymbol{A}$ by counting the number of cells in each cluster in each timepoint
4. Construct timepoint * gene pseudobulk matrix $\boldsymbol{Y}$ using the formula $Y = AX$
5. Define $LambdaSet = \{10^{-8}, 10^{-7}, 10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}\}$
6. For each column of Y, denoted as y :
7. Fit ridge regression model *Beta* using the function `Beta = glmnet(x = A, y = y, lambda = LambdaSet, alpha = 0)`

8. For each column in X, denoted as x :

- For each lambda L :

1. Compute x' = column L of $Beta$
2. Compute error in predicting x as $||x - x'||^2$
3. Accumulate error for each lambda as the sum of errors across genes
4. Select lambda with minimum accumulated error

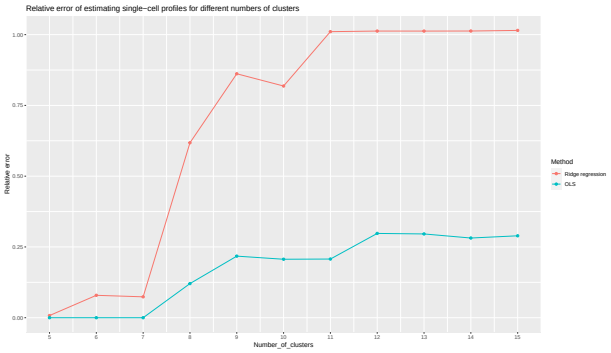


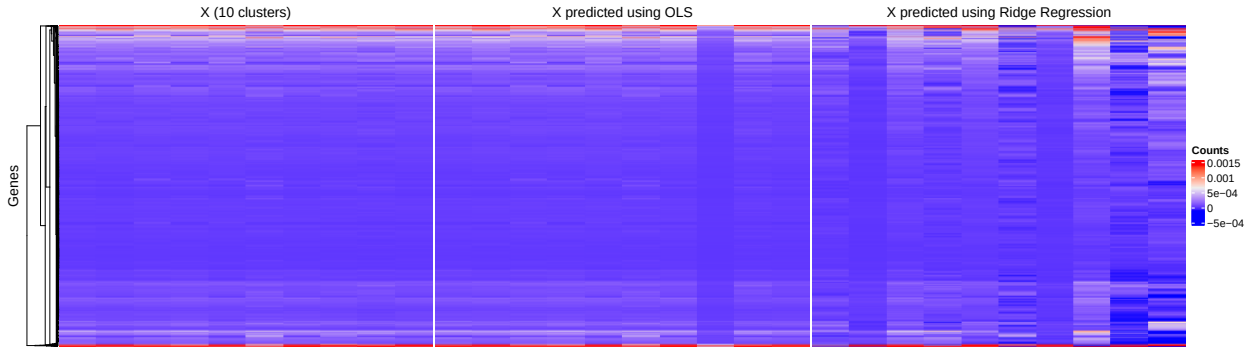
For each number of clusters, we select the lambda which led to the minimal sum squared error when comparing the predicted single-cell profiles to the actual ones. Those selected lambdas are shown below.

Table 2: Ridge regression - Lambda values that minimize error

Number of clusters	Best lambda
5	1e-10
6	1e-10
7	1e-10
8	1e-05
9	1e-05
10	1e-04
11	1
12	1
13	1
14	1
15	1

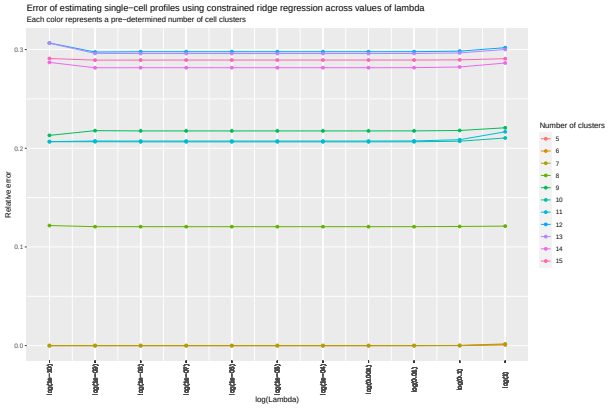
Comparing ridge regression to OLS The below plots show a comparison of using ridge regression and non-negative constrained OLS for predicting the single-cell profiles from the bulk data and the timepoint-specific proportions.





Ridge regression with non-negative constraint

The results above do not show an improvement in the single-cell predictions when using ridge regression. In this section, we explore the addition of a non-negative constraint to the ridge regression model, similar to how we applied it for OLS earlier in this report. To achieve this, we solve the equation $\hat{X} = min_{x>=0}(-2Y^T AX + X^T A^T AX)$ after adding a ridge penalty λ to the diagonal of the matrix $A^T A$. The R function *solve.QP* from the *quadprog* package is used to solve this equation one gene at a time to estimate the expression profile at the cell-type level. To decide on the optimal value for the parameter λ, we test a range of values between 10^{-10} and 1 for each number of clusters. For each value of λ, we sum the errors in estimating each gene's single-cell profile as a measure of the accuracy of the predicted single-cell profiles. The λ that leads to the minimal error is selected as the optimal value.

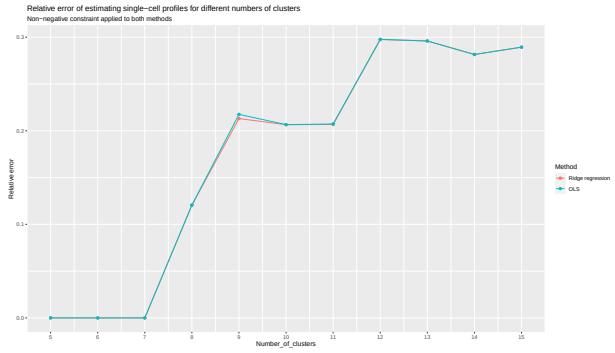


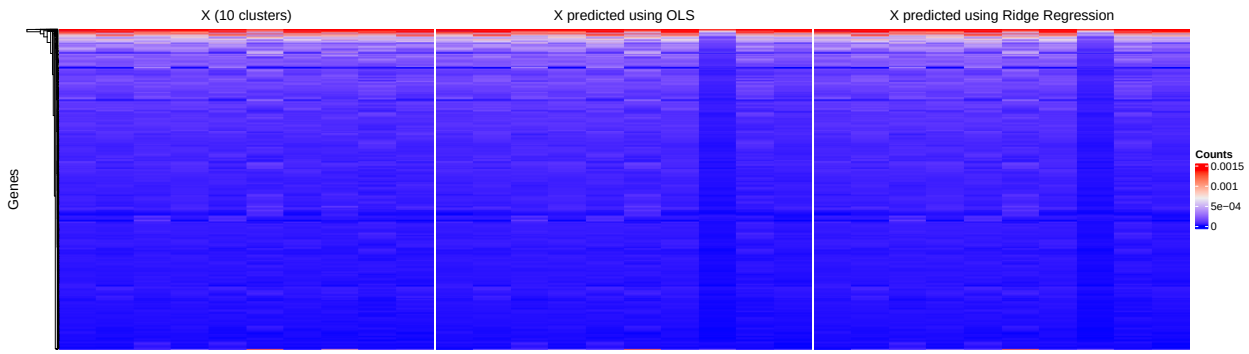
For each number of clusters, we select the lambda which led to the minimal sum squared error when comparing the predicted single-cell profiles to the actual ones. Those selected lambdas are shown below.

Table 3: Constrained ridge regression - Lambda values that minimize error

Number of clusters	Best lambda
5	1e-10
6	1e-10
7	1e-10
8	1e-07
9	1e-10
10	1e-07
11	1e-10
12	1e-09
13	1e-06
14	1e-09
15	1e-09

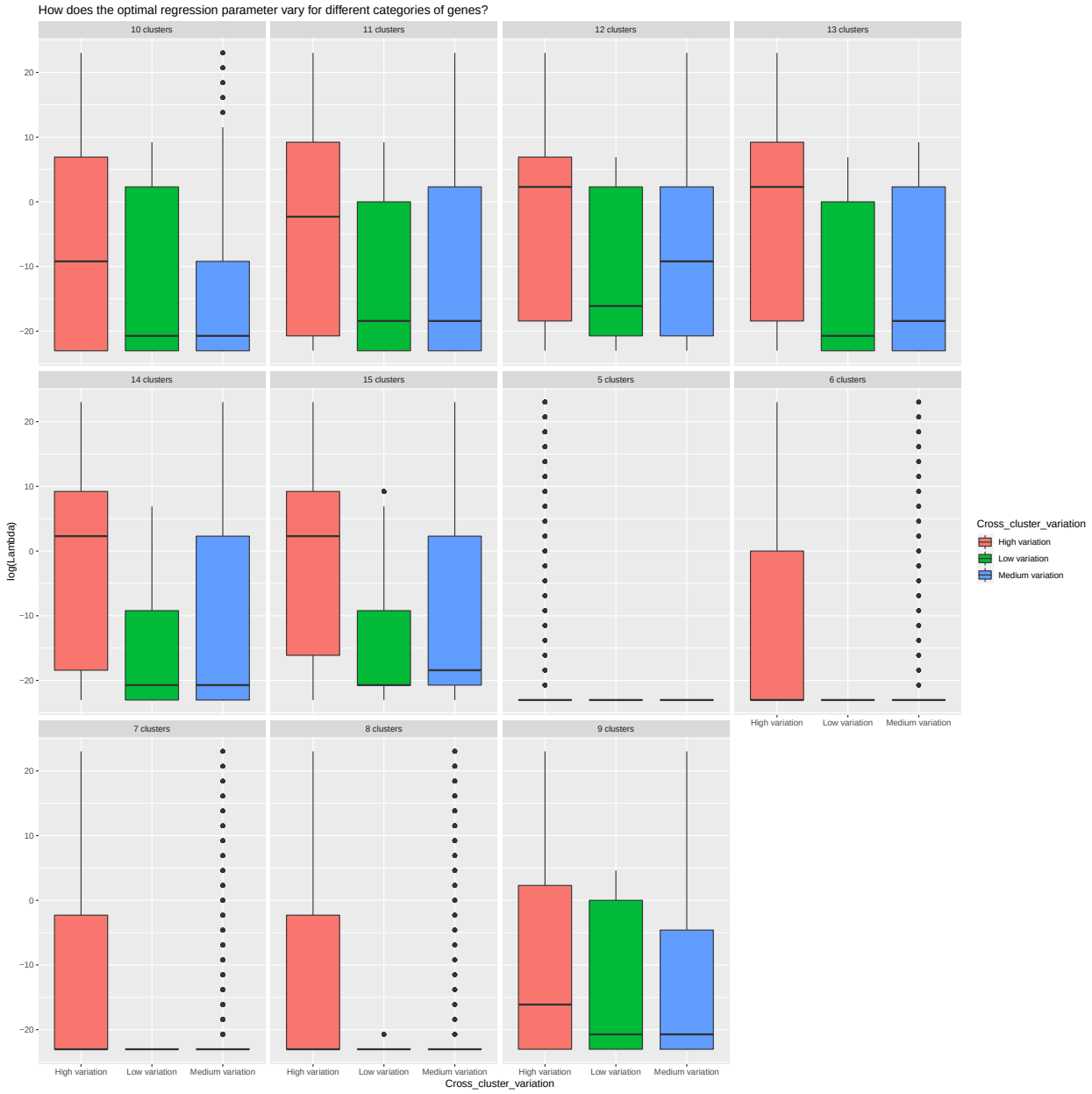
After selecting the optimal λ for each number of clusters, we compare the results of using non-negative constrained ridge regression and OLS on predicting the single-cell profiles. The errors are reported as relative RMAD (relative mean absolute deviation) using the formula $|X - X'|/X$, where $|X|$ is the absolute value of the difference. This error is computed for each gene and the final reported score is the average RMAD value, in other words: *on average, how different is a gene's predicted values compared to the true ones?* The resultant relative RMAD values for each of the methods are shown below, along with an example predicted matrix.





How does the optimal regression parameter vary for different categories of genes?

We group the genes according to how variable their expression is across clusters and compute and compare the optimal regression parameter for each of the groups. The expression variability is computed as the coefficient of variation (standard deviation / mean) of the gene's expression across clusters.



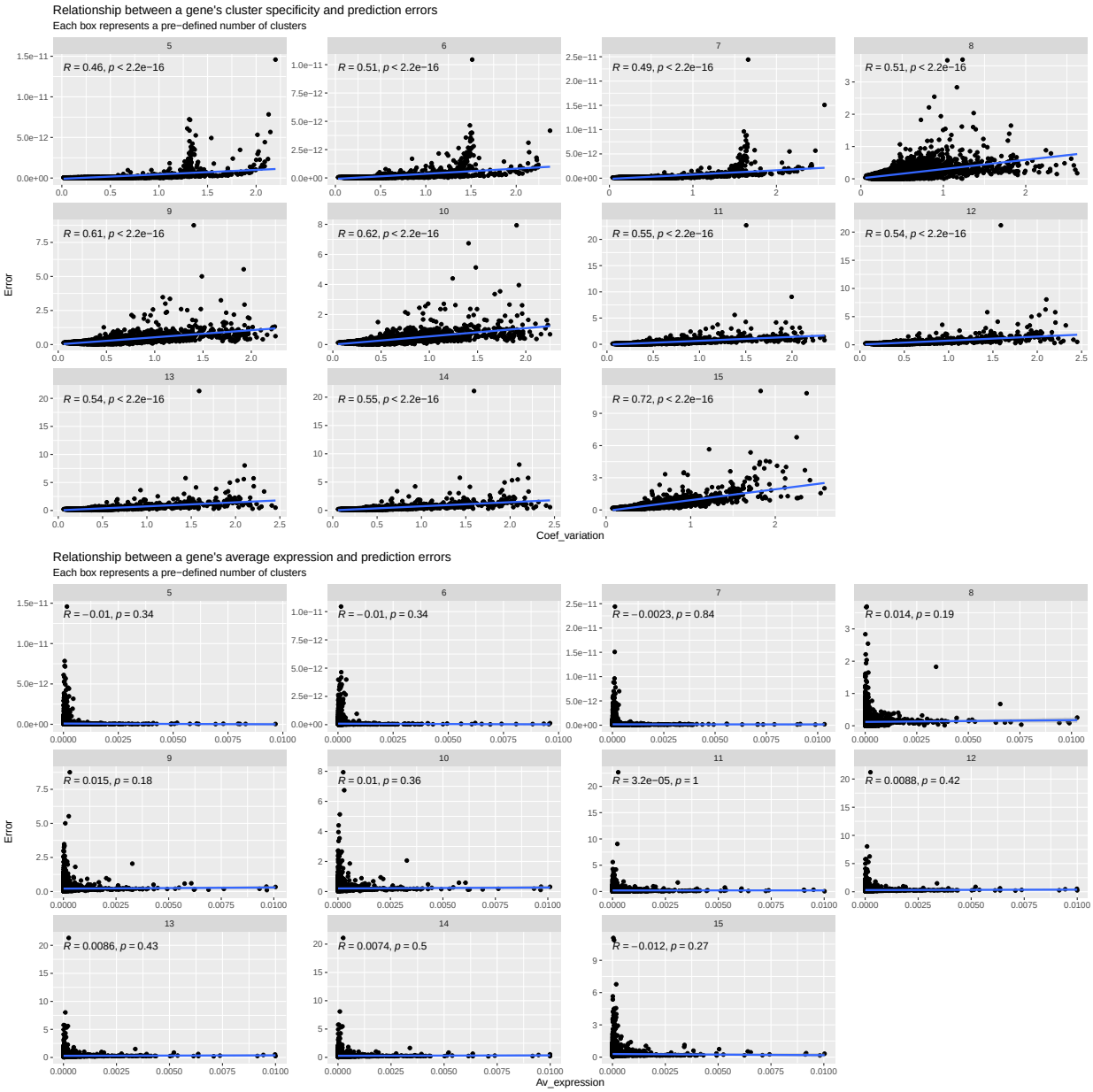
Distribution of errors across genes

This section explores whether a ***partial prediction*** might be more feasible, that is whether the single-cell predictions might be more accurate for a subset of the genes, such as differential markers, by comparing the distributions of prediction errors for individual genes as they relate to cell-type expression specificity.

To explore the relationship between properties of the genes and their corresponding predicted values, for each gene we compute the following measures:

- ***Expression specificity:*** This metric looks at the relative specificity of a given gene's expression to the clusters. The coefficient of variation of the gene's cluster-specific expression values is computed as the standard deviation of the per-cluster expression values divided by the mean.
- ***Average expression:*** This metric is concerned with the relative abundance of each gene's transcript, and is simply computed as the mean expression of the gene across clusters.

Each of the above measures are then compared to the error in predicting the gene's expression value using regression.

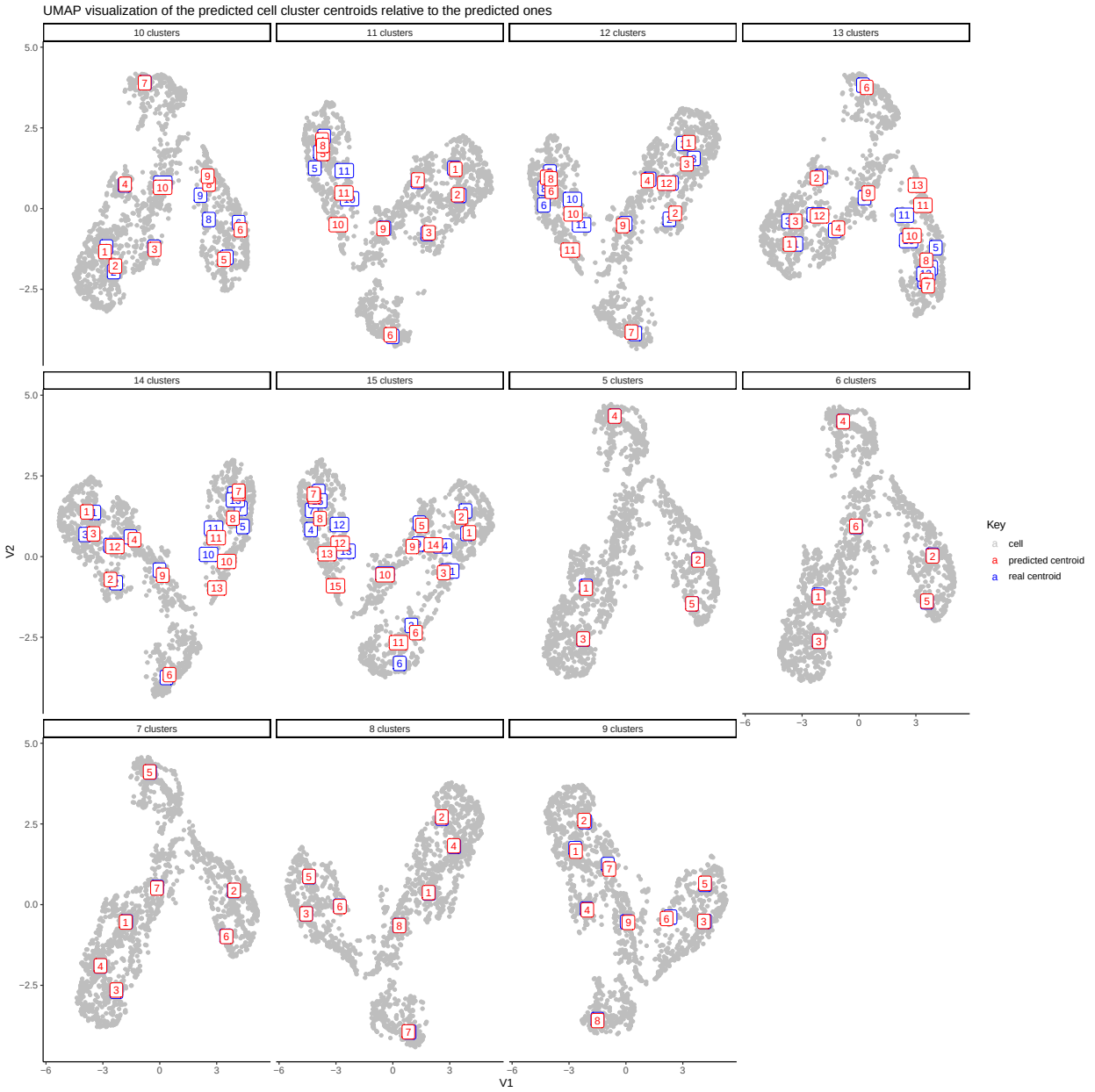


Evaluation of single-cell predictions

The below sections examine the accuracy of the single-cell predictions made using ridge regression in the previous section. Note: the bulk matrix that is deconvoluted in this section is the reconstructed pseudo-bulk

matrix ($Y = AX$).

Are our predicted cluster centroids close to the real ones? The cluster centroid is each cluster's average gene expression profile. Visualizing the two sets of centroids (real & predicted) side-by-side on a *UMAP* will help us answer this question. For each number of clusters between 5 and 15, we cluster the data using *Seurat* and construct the matrix X which contains the average gene expression by cluster. We then use the ridge regression approach defined in the previous section to construct the matrix X'. Finally, we perform a *UMAP* projection using the R package uwot on the original gene expression matrix but with the addition of the real and predicted cluster centroids, i.e. the rows of X and X'. The *UMAP*s below show the projections for each number of clusters.



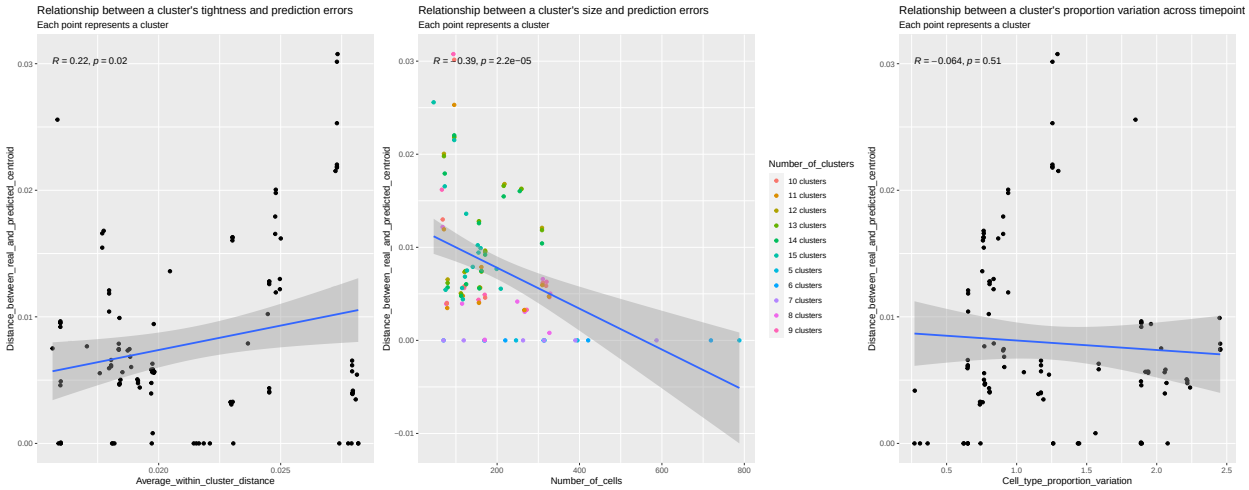
How similar is the average cell to it's real centroid compared to the predicted one? To determine if our predictions are reasonable approximations of the real data, we also compare the predictions to the

real data by computing the Euclidean distance between individual cells' measurements and their real cluster average as opposed to the distance between our predictions and the same cluster average.

More specifically, for a given number of clusters, we cluster the data using *Seurat*, followed by iterating over each of the ~1,900 cells in the original single-cell expression matrix and computing the Euclidean distance between it's profile and that of its cluster's centroid. Next, for each cluster we compute the average distance of it's cells to the centroid, as well as the distance between our predicted centroid and the corresponding cluster centroid. By comparing these two distances side-by-side, we can determine whether our predicted centroids fall within the correct intra-cluster range.

Which clusters are easier to predict? The accuracy of recapturing the cell cluster profiles varied by cluster. In this section, we are interested in examining the mathematical properties of the cell clusters derived from the scRNA-Seq data that influence the quality of our predictions. We first compute the correlation between how 'tightly-knit' a cluster is, i.e. average within-cluster distance to the centroid, and the error in

predicting the profiles. We also correlate the prediction error with the number of cells in each cluster and the variation of each cell type's proportion across timepoints. The error in this section is taken as the Euclidean distance between the cluster's predicted centroid and the actual centroid.



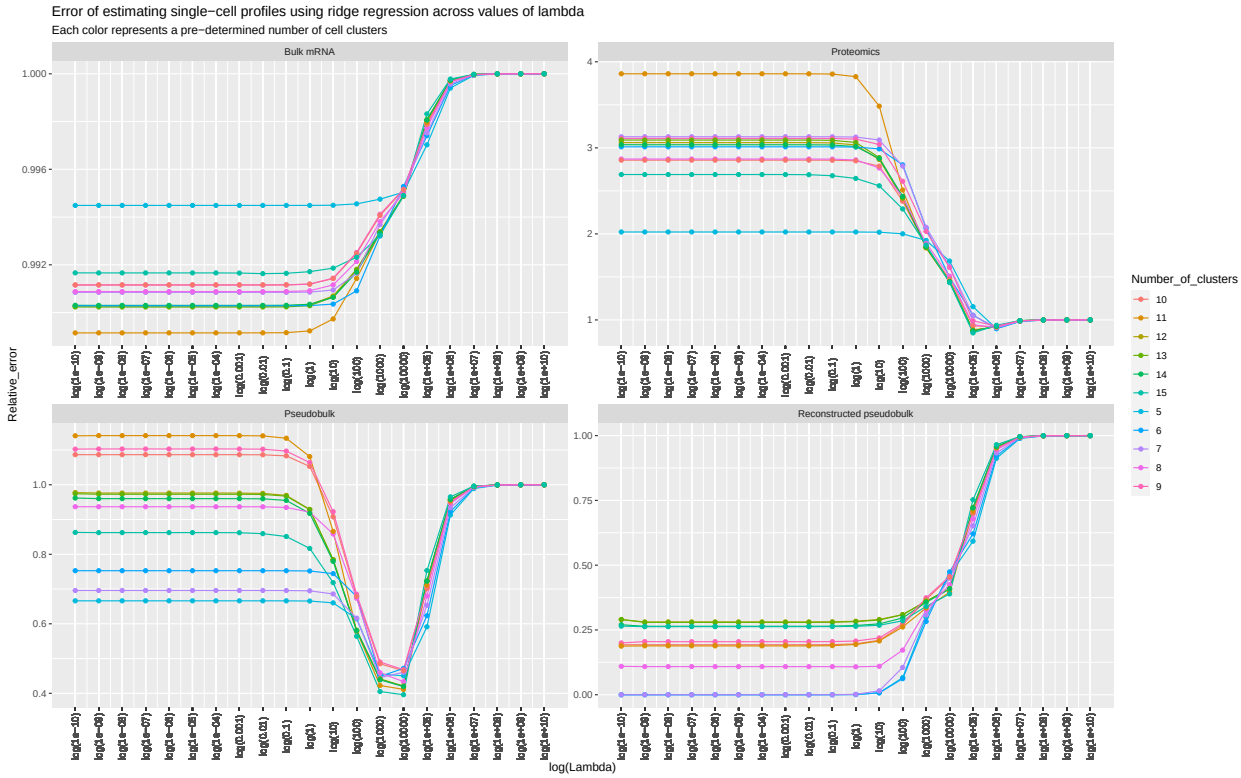
Algorithm Results

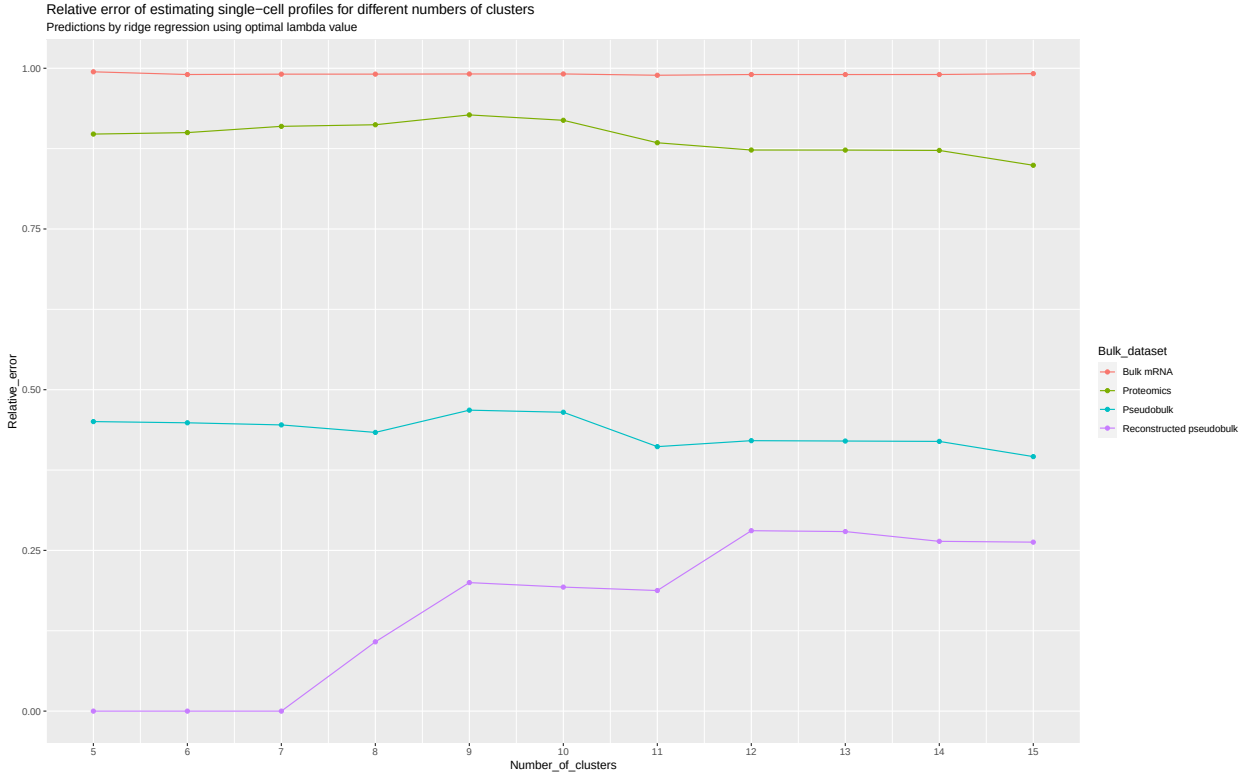
Prediction accuracy

The previous sections in this report optimized the deconvolution algorithm by using only the single-cell RNA-Seq data. This section applies the resultant deconvolution algorithm to the following 4 bulk datasets:

1) ***Pseudo-bulk***: A pseudo-bulk RNA dataset is created for each timepoint by summing the gene counts of all cells within the timepoint.
2) ***Reconstructed Pseudo-bulk***: The reconstructed pseudo-bulk is created by matrix multiplication of the average cell cluster expression of each gene and the number of cells in each cluster each timepoint, i.e. $Y = AX$.
3) ***Bulk mRNA***: Bulk mRNA data from a microarray experiment.
4) ***Proteomics***: Bulk proteomics data from a mass spectrometry experiment. To make the scale of the data comparable to that of the RNA data, the proteomics intensities are divided by the ratio of the sum of the proteomics intensities to the sum of the reconstructed pseudobulk counts.

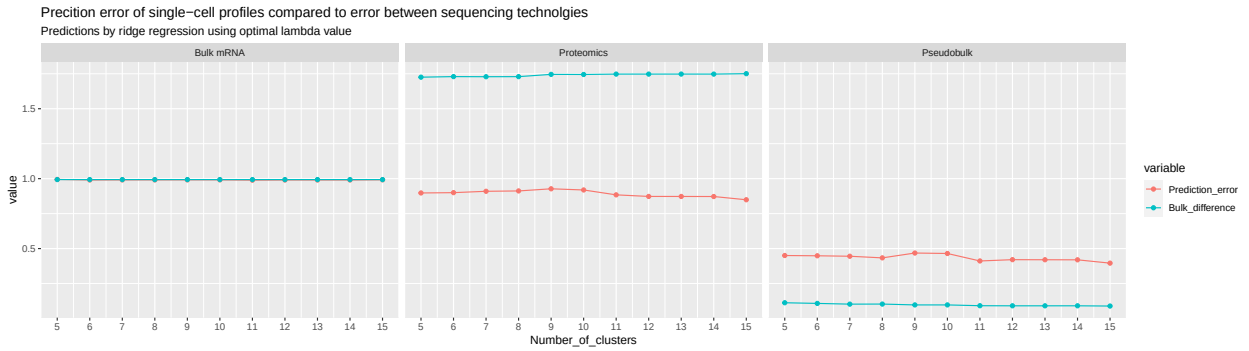
For each of the above datasets, we apply the deconvolution algorithm and compute the prediction error as the relative error between our predicted cell-cluster matrix X' and the one obtained from the Seurat clustering of the scRNA data X. Since the identified genes differ across the three technologies, we retain only the ~5,000 genes common to all datasets for this comparison.


Error of estimating single−cell profiles using ridge regression across values of lambda
Each color represents a pre−determined number of cell clusters
Bulk mRNA
Proteomics
Pseudobulk
Reconstructed pseudobulk
Relative_error
log(Lambda)
Number_of_clusters
10
11
12
13
14
15
5
6
7
8
9


Relative error of estimating single−cell profiles for different numbers of clusters
Predictions by ridge regression using optimal lambda value
Relative_error
Number_of_clusters
Bulk_dataset
Bulk mRNA
Proteomics
Pseudobulk
Reconstructed pseudobulk

Accounting for differences in experimental technologies

The errors computed in the previous section compared the predicted single-cell profiles to those derived from scRNA-Seq without accounting for the fact that the bulk mRNA and proteomics data were obtained from different technologies, namely microarray and mass spectrometry, which introduce their own technical noise. Indeed, even the pseudobulk that was reconstructed from the scRNA-Seq data after clustering differs from the actual summed-up measurements from the raw data. To account for these differences, we find the relative error between each of the bulk datasets and the "reconstructed" pseudobulk that was used to optimize the algorithm. By plotting that difference on the same plot as the prediction errors, we can place the prediction errors within the context of the underlying experimental differences. For example, in the case of proteomics it would correspond to adjusting the prediction error based on the discrepancy between the protein and RNA levels.

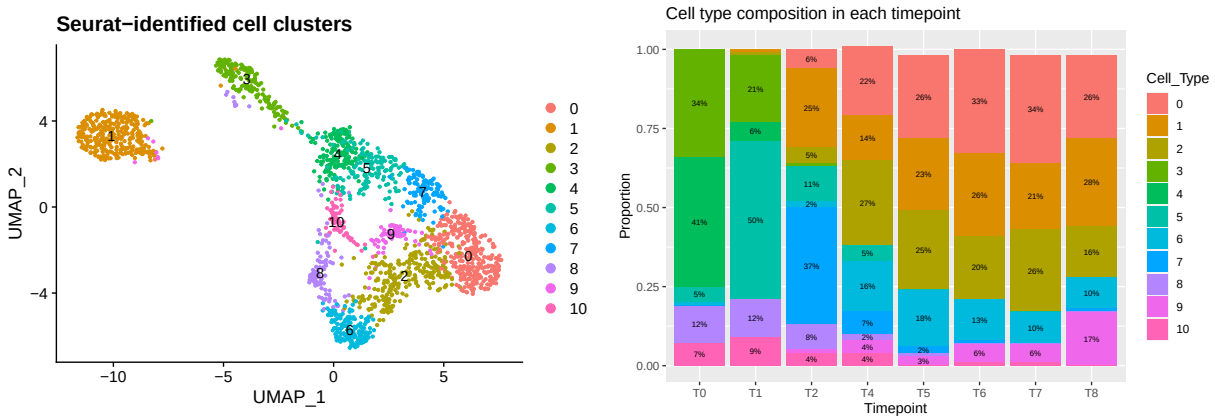


Deconvolution of proteomics data

In this section of the report, we focus on deconvoluting the bulk proteomics data with 11 Seurat-defined clusters. We furthermore focus on a set of 80 hallmark genes associated with EMT from the MSigDB database.

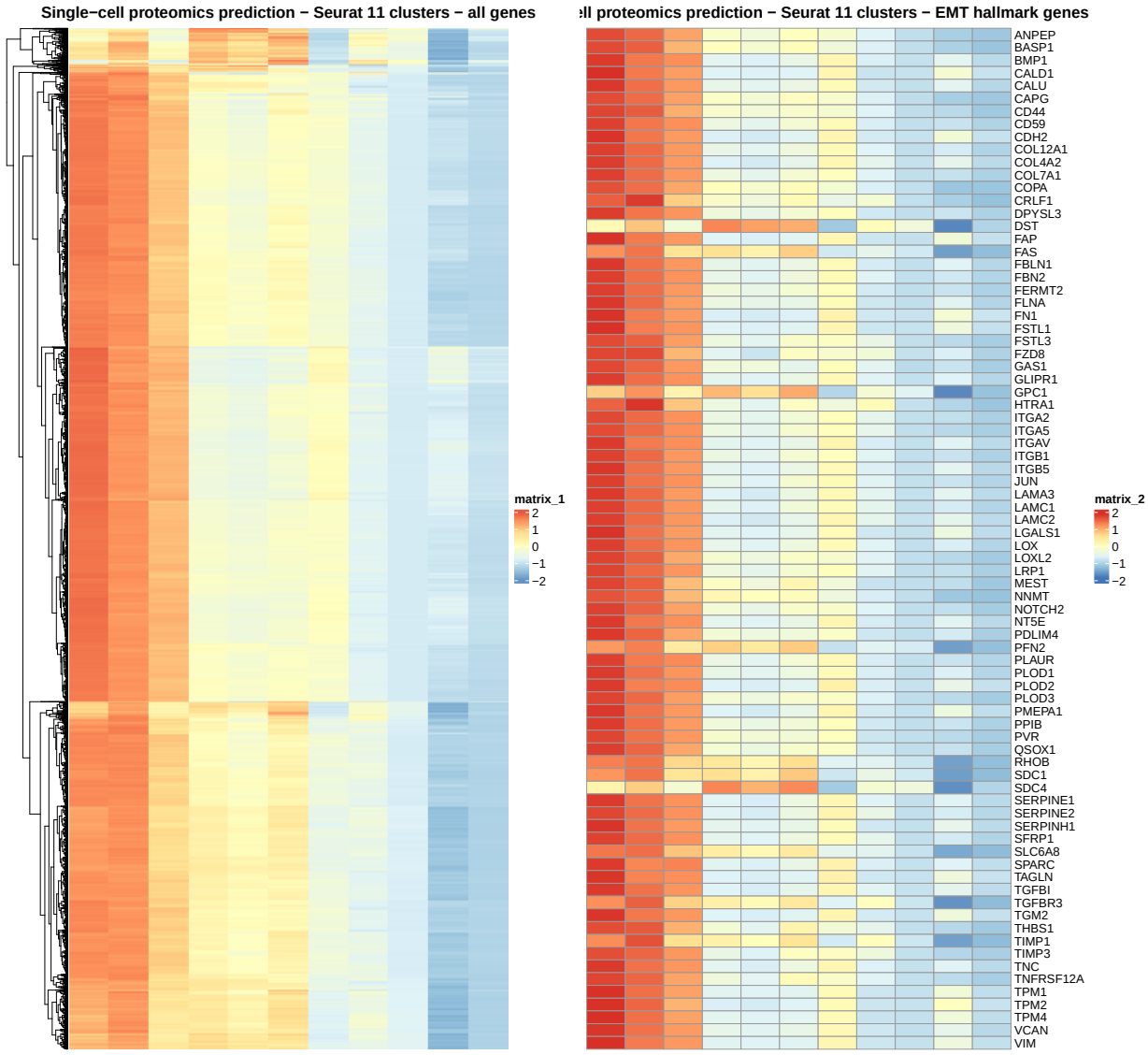
Seurat clustering results

```
## Modularity Optimizer version 1.3.0 by Ludo Waltman and Nees Jan van Eck
##
## Number of nodes: 1913
## Number of edges: 65152
##
## Running Louvain algorithm...
## Maximum modularity in 10 random starts: 0.7942
## Number of communities: 11
## Elapsed time: 0 seconds
```



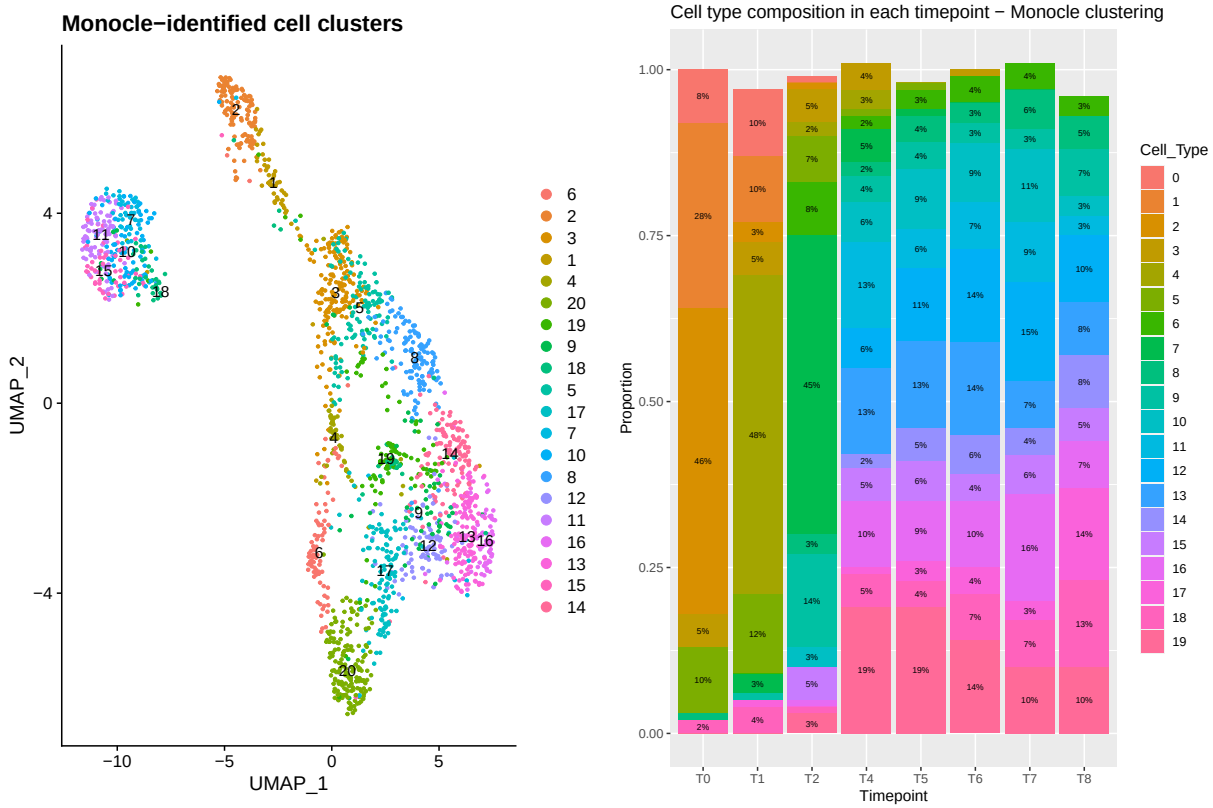
Single-cell proteomics predictions

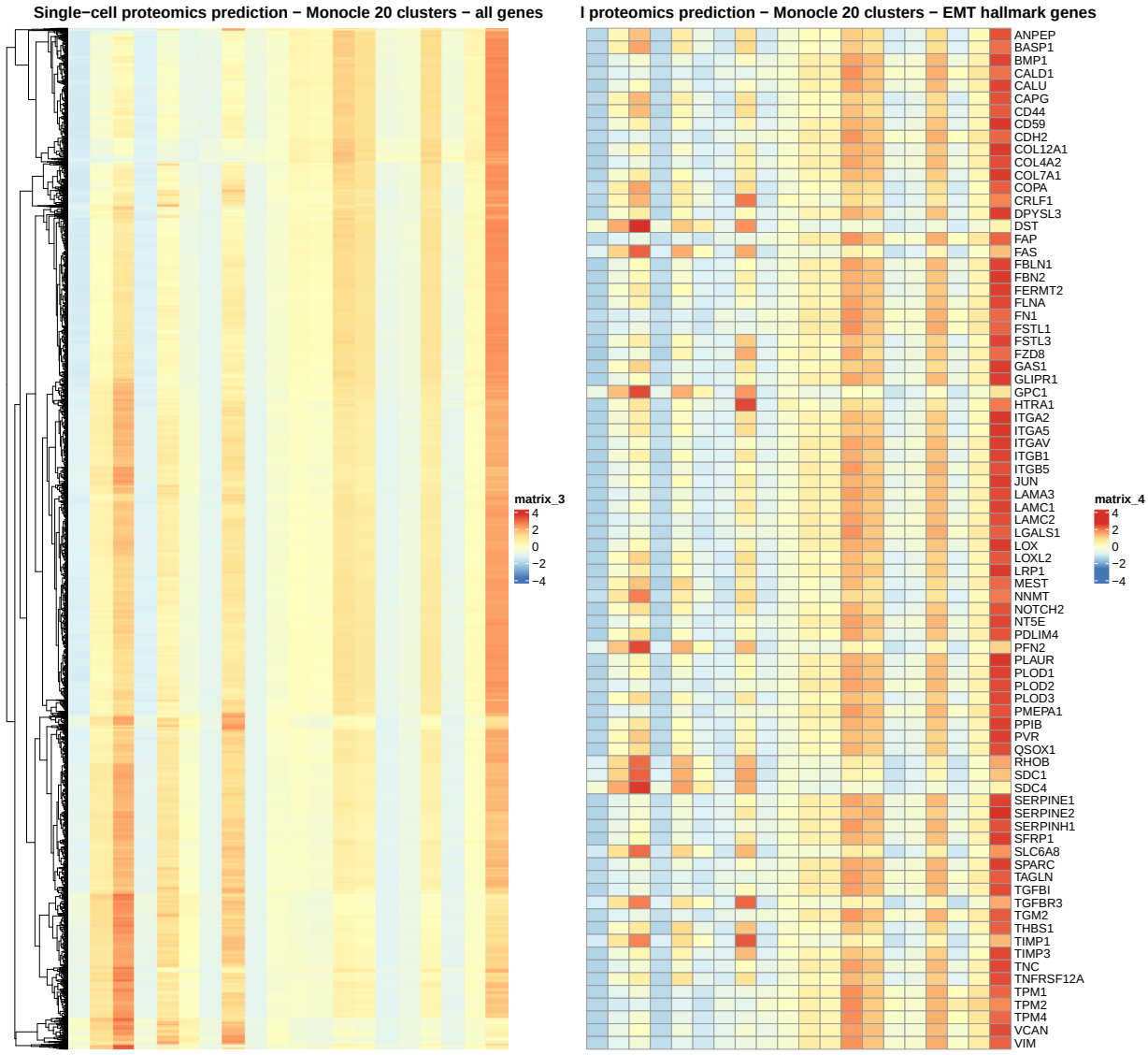
```
## Modularity Optimizer version 1.3.0 by Ludo Waltman and Nees Jan van Eck
##
## Number of nodes: 1913
## Number of edges: 65152
##
## Running Louvain algorithm...
## Maximum modularity in 10 random starts: 0.7942
## Number of communities: 11
## Elapsed time: 0 seconds
```


Single−cell proteomics prediction − Seurat 11 clusters − all genes
matrix_1
2
1
0
−1
−2
ll proteomics prediction − Seurat 11 clusters − EMT hallmark genes
ANPEP
BASP1
BMP1
CALD1
CALU
CAPG
CD44
CD59
CDH2
COL12A1
COL4A2
COL7A1
COPA
CRLF1
DPYSL3
DST
FAP
FAS
FBLN1
FBN2
FERMT2
FLNA
FN1
FSTL1
FSTL3
FZD8
GAS1
GLIPR1
GPC1
HTRA1
ITGA2
ITGA5
ITGAV
ITGB1
ITGB5
JUN
LAMA3
LAMC1
LAMC2
LGALS1
LOX
LOXL2
LRP1
MEST
NNMT
NOTCH2
NT5E
PDLIM4
PFN2
PLAUR
PLOD1
PLOD2
PLOD3
PMEPA1
PPIB
PVR
QSOX1
RHOB
SDC1
SDC4
SERPINE1
SERPINE2
SERPINH1
SFRP1
SLC6A8
SPARC
TAGLN
TGFBI
TGFBR3
TGM2
THBS1
TIMP1
TIMP3
TNC
TNFRSF12A
TPM1
TPM2
TPM4
VCAN
VIM
matrix_2
2
1
0
−1
−2

Alternative clustering

Here, we examine an alternative clustering of the data, which clustered the scRNA data into 20 clusters using the Monocle3 algorithm.




Single−cell proteomics prediction − Monocle 20 clusters − all genes
matrix_3
4
2
0
−2
−4
l proteomics prediction − Monocle 20 clusters − EMT hallmark genes
ANPEP
BASP1
BMP1
CALD1
CALU
CAPG
CD44
CD59
CDH2
COL12A1
COL4A2
COL7A1
COPA
CRLF1
DPYSL3
DST
FAP
FAS
FBLN1
FBN2
FERMT2
FLNA
FN1
FSTL1
FSTL3
FZD8
GAS1
GLIPR1
GPC1
HTRA1
ITGA2
ITGA5
ITGAV
ITGB1
ITGB5
JUN
LAMA3
LAMC1
LAMC2
LGALS1
LOX
LOXL2
LRP1
MEST
NNMT
NOTCH2
NT5E
PDLIM4
PFN2
PLAUR
PLOD1
PLOD2
PLOD3
PMEPA1
PPIB
PVR
QSOX1
RHOB
SDC1
SDC4
SERPINE1
SERPINE2
SERPINH1
SFRP1
SLC6A8
SPARC
TAGLN
TGFBI
TGFBR3
TGM2
THBS1
TIMP1
TIMP3
TNC
TNFRSF12A
TPM1
TPM2
TPM4
VCAN
VIM
matrix_4
4
2
0
−2
−4