

# hdWGCNA analysis multiomic\_stimulated\_sc

|            |                           |
|------------|---------------------------|
| Status     | Completed                 |
| Type       | code                      |
| Priority   | High 🔥                    |
| Project(s) |                           |
| Start Date | @August 20, 2022          |
| Due Date   | @October 27, 2022         |
| Assign     |                           |
| Date Added | @October 17, 2022 8:48 AM |
| Size       | sprint                    |
| URL        |                           |
| Comment    |                           |

[Description](#)

[Data](#)

[Metadata](#)

[Data locations](#)

[Methods](#)

[Tools](#)

[Results](#)

[Exploratory data analysis](#)

[NormalizeData](#) SC-α, SC-β, SC-δ, SC-EC cell analysis across all genes with 25 neighbors ( [allcells\\_ND\\_25neighbors](#) )

[NormalizeData](#) SC-α, SC-β, SC-EC (no δ) cell analysis across all genes with 25 neighbors ( [allcells\\_ND\\_25neighbors](#) )

[NormalizeData](#) SC-β cell analysis across all genes with 25 neighbors ( [betacells\\_ND\\_25neighbors](#) )

[NormalizeData](#) SC-β cell analysis across all genes with parameter exploration ( [betacells\\_ND\\_exploration](#) )

[Help](#)

[Next steps](#)

## Description

I want to perform a single cell weighted gene correlation network analysis on the RNA from the treated multiome samples of stem cell derived islets. Mei has already sent me a processed object that I will use to simplify the workflow. I'll read it in to Seurat and go from there.

## Data

### ▼ Metadata

### ▼ Data locations

Processed and raw data files sent by Mei: [/cellar/users/aklie/data/beta\\_cell\\_networks/multiomic\\_stimulated\\_sc](#)

Analysis code and results:

[/cellar/users/aklie/projects/beta\\_cell\\_networks/notebooks/multiomic\\_stimulated\\_sc/infer\\_cellular\\_programs/hdWGCNA](#)

**Note:** I think this organization makes a lot of sense for this tool, where each notebook generates results that have a corresponding results directory in the same folder. Could also have a separate results directory outside the notebooks but I think

this is ok for now.

## Methods

---

### ▼ Tools

- hd [WGCNA](#)
  - Network construction
    - [SetUpForWGCNA](#)
    - [MetacellsByGroup](#)
    - [NormalizeMetacells](#): wrapper around [NormalizeData](#) from Seurat
  - Basic visualization
  - Module trait correlation
  - Gene set enrichment
  - Network analysis
  - Motif analysis

## Results

---

### ▼ Exploratory data analysis

- See [0\\_EDA.ipynb](#)
- Basically just gives us a feel for all the data that's there

### ▼ [NormalizeData](#) SC- $\alpha$ , SC- $\beta$ , SC- $\delta$ , SC-EC cell analysis across all genes with 25 neighbors ( [allcells\\_ND\\_25neighbors](#) )

Active assay: [RNA](#)

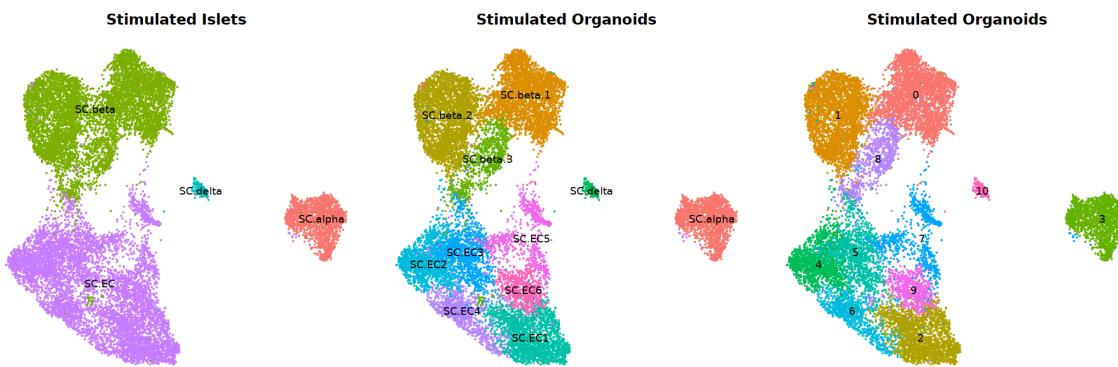
Cell types under consideration: [SC. \$\alpha\$](#) , [SC. \$\beta\$](#) , [SC. \$\delta\$](#) , [SC. EC](#)

Fraction of cells to keep a gene: [0.05](#)

Number of neighbors for metacell construction: [25](#)

### ▼ Network Construction

- Starting with [16342](#) cells across [28796](#) genes in the integrated multiome data

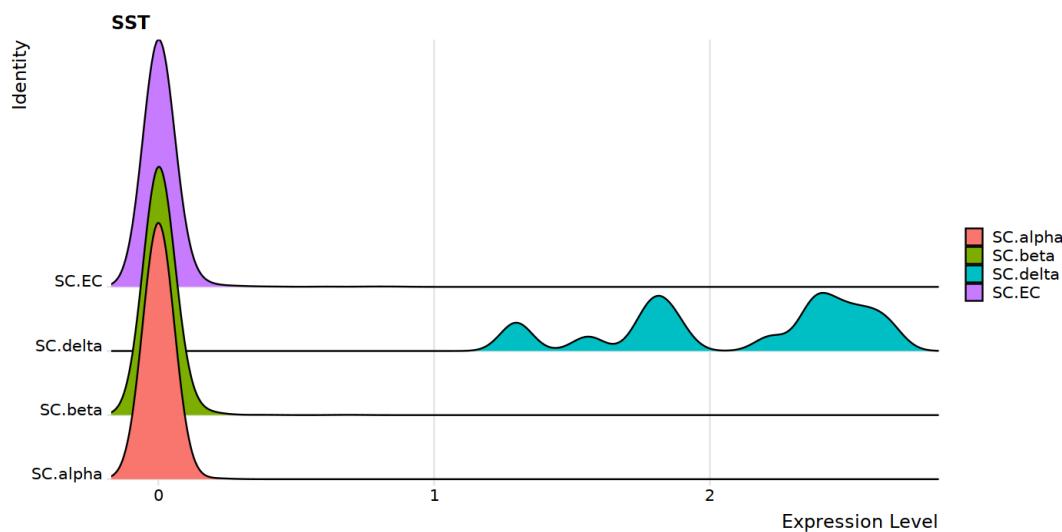
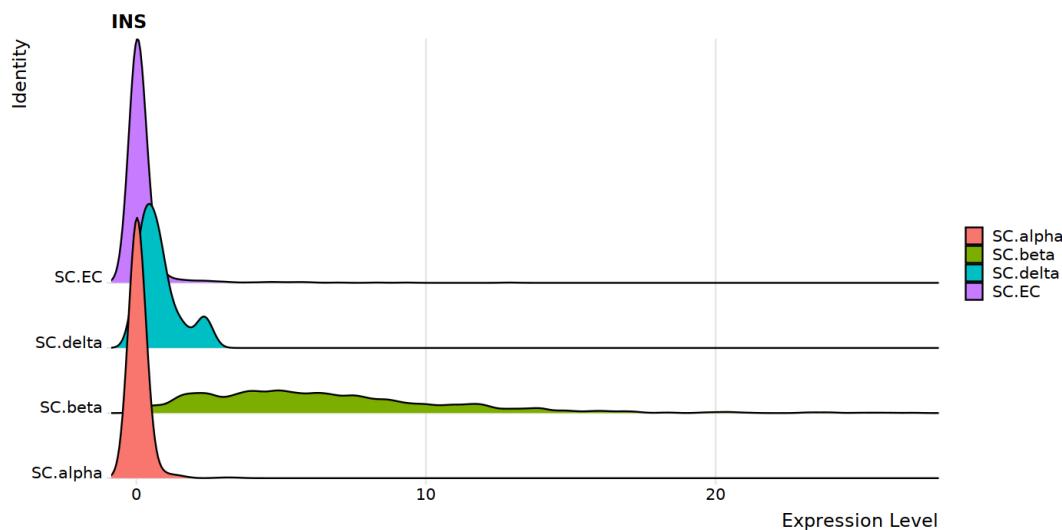


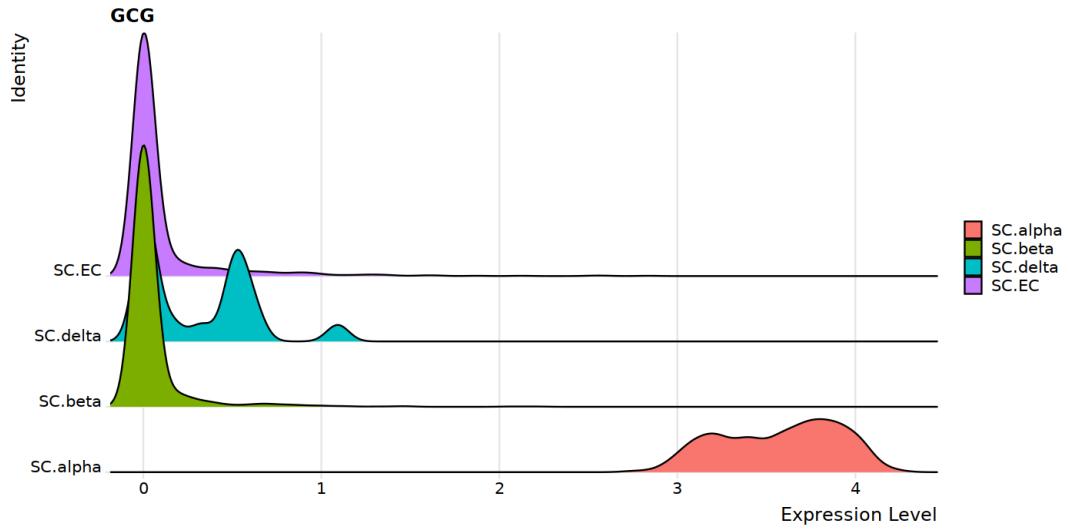
- Keeping only  $\alpha$ ,  $\beta$  and  $\delta$  cells from original processed Seurat [.rds](#)

| SC.alpha | SC.beta | SC.delta | SC.EC |
|----------|---------|----------|-------|
| 1821     | 7042    | 186      | 7293  |

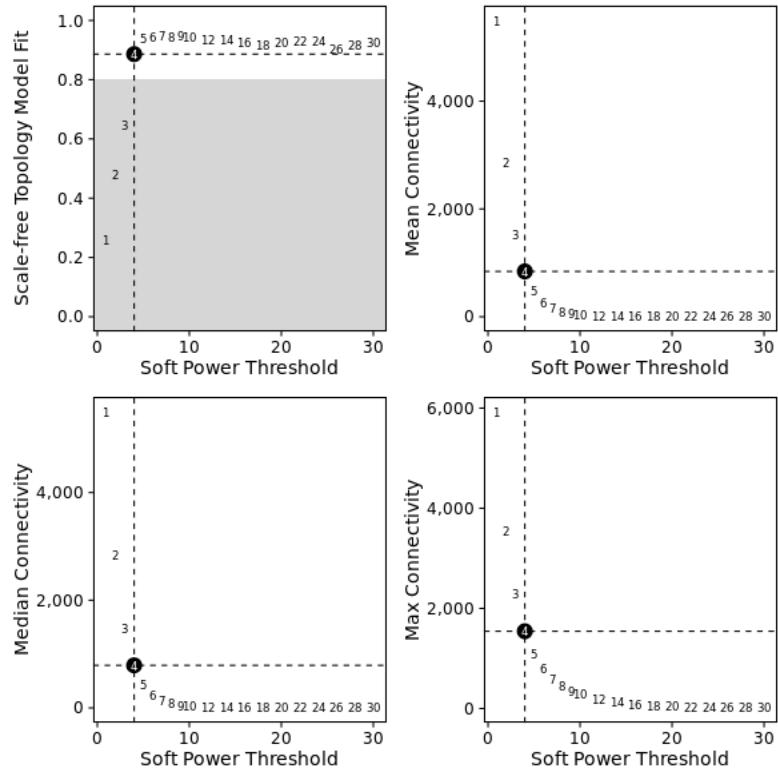
- Using a fraction of  $0.05$  based on the `counts` slot of the active assay in `SetupForWGCNA` function to keep only genes expressed in at least 5% of these cells
  - $10832$  genes remain after this filter
- Created metacells using top  $25$  nearest neighbors within a cell type (label from processed object). Used the `counts` to create these metacells in `RNA` assay and only allowed them to share  $10$  cells in common at most. Minimum cells by default is  $100$  and the max number of metacells to construct per cell type is  $1000$ . Can normalize this metacell object using Seurat's default function that uses a `scale.factor` of  $10000$  and by default takes the natural log with a pseudo-count
  - Have a total of  $2597$  metacells after this

| SC.alpha | SC.beta | SC.delta | SC.EC |
|----------|---------|----------|-------|
| 579      | 1000    | 18       | 1000  |



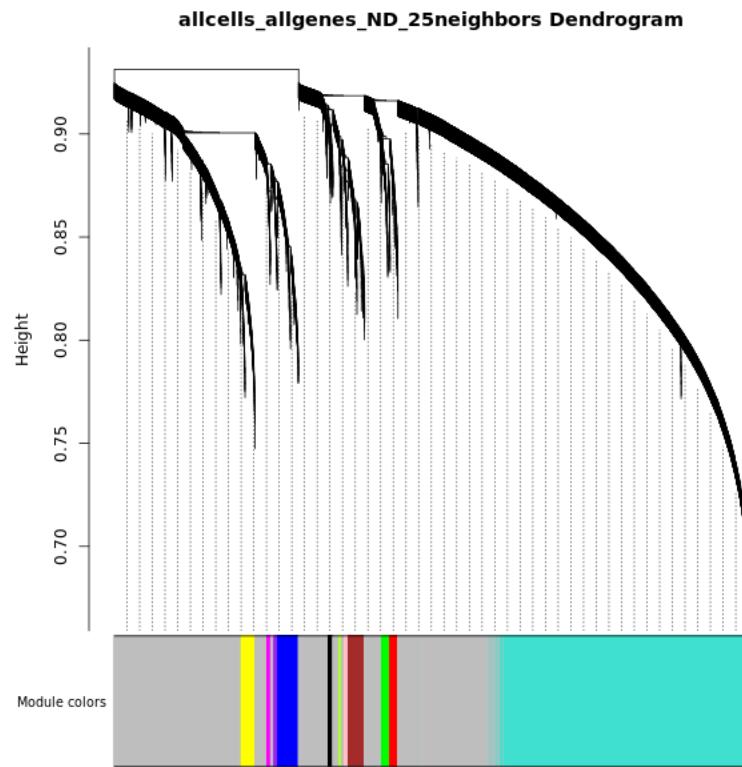


- Using the `SetDataExpr` function, selected the normalized metacell data from the `data` slot of the stored metacell Seurat object.
- Selected a soft-power threshold of `4` for downstream network construction
- 



|   | Power | SFT.R.sq  | slope     | truncated.R.sq | mean.k.   | median.k. | max.k.    |
|---|-------|-----------|-----------|----------------|-----------|-----------|-----------|
|   | <dbl> | <dbl>     | <dbl>     | <dbl>          | <dbl>     | <dbl>     | <dbl>     |
| 1 | 1     | 0.2595093 | 12.586806 | 0.9754613      | 5499.3142 | 5503.0547 | 5930.3070 |
| 2 | 2     | 0.4783162 | -7.875034 | 0.8913452      | 2868.1998 | 2828.7163 | 3550.7871 |
| 3 | 3     | 0.6486607 | -6.187506 | 0.8294011      | 1533.8584 | 1476.6128 | 2279.7623 |
| 4 | 4     | 0.8862293 | -5.303395 | 0.9541745      | 840.5963  | 790.1529  | 1543.7580 |
| 5 | 5     | 0.9397192 | -4.410018 | 0.9822279      | 472.1439  | 431.5324  | 1089.5089 |
| 6 | 6     | 0.9424749 | -3.782363 | 0.9816709      | 271.9335  | 239.1272  | 794.8577  |

- Using the normalized metacell expression matrix, performed WGCNA using soft power of `4`. By default this builds a signed network using Pearson correlation.

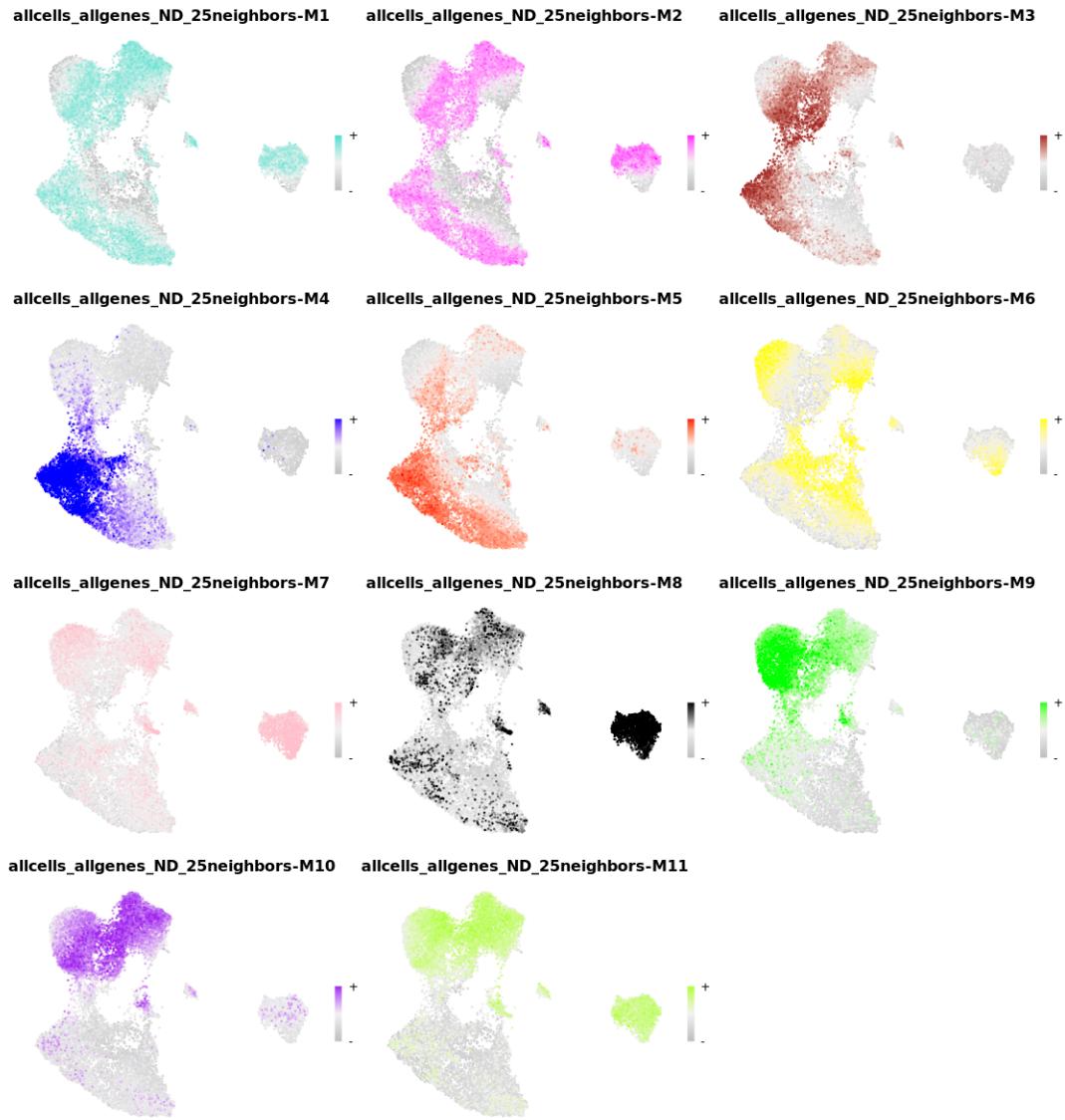


|        |       |      |       |       |             |      |       |
|--------|-------|------|-------|-------|-------------|------|-------|
| Module | black | blue | brown | green | greenyellow | grey | mager |
|--------|-------|------|-------|-------|-------------|------|-------|

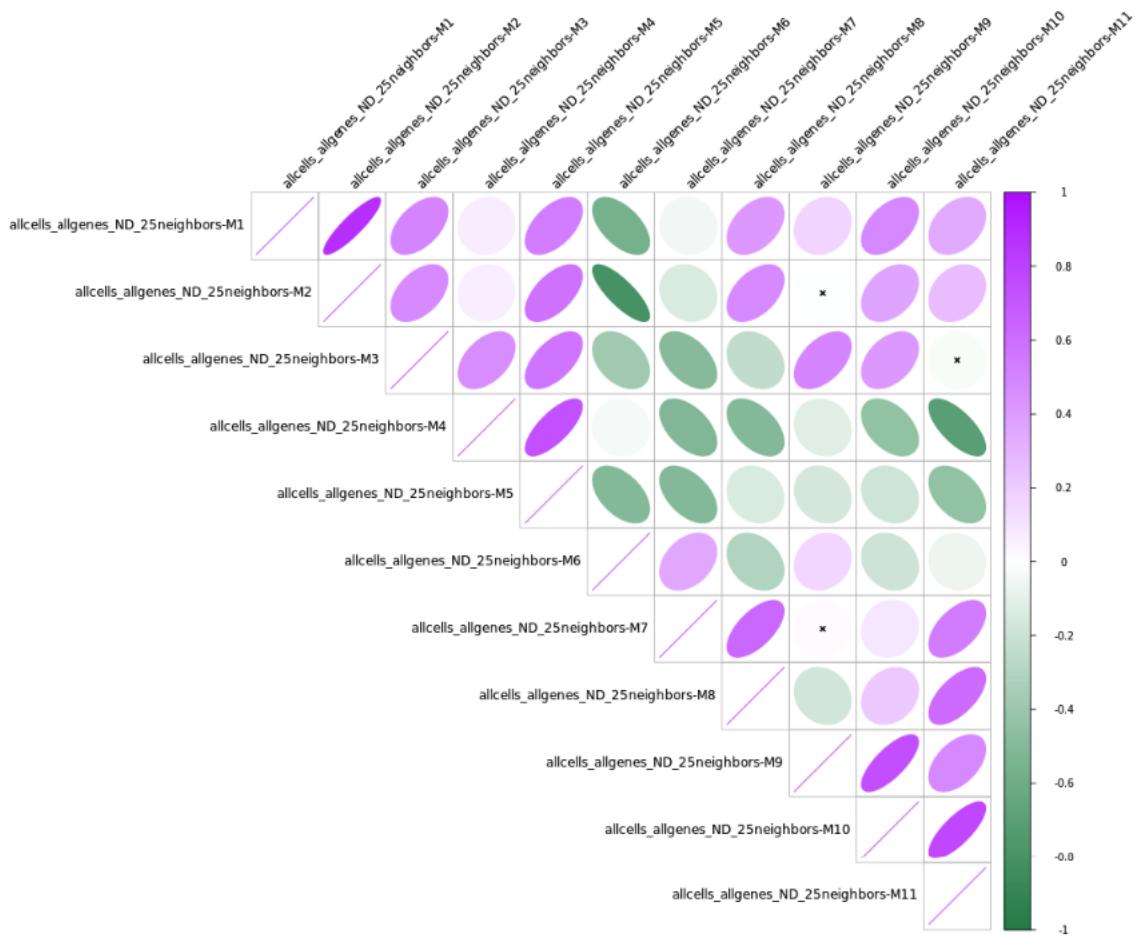
- Did not include a `group_by_vars` argument in the `ModuleEigengenes` function, which look like stops Harmony batch correction from running. The module eigengenes for each module are the 1st PC of the subsetted matrix
- Did not include `group_by` argument in the `ModuleConnectivity` function (which seems to make it take a long time —> ended up taking ~30 minutes). Computes a correlation for every gene in every module with its module eigengene. We still have 3000+ cells and 8000+ genes, so this is a heavy computation step

## ▼ Basic Visualization

Plotting the raw MEs (PCA yields a sample x PC matrix for each module, I can overlay these onto the previous UMAP plot:



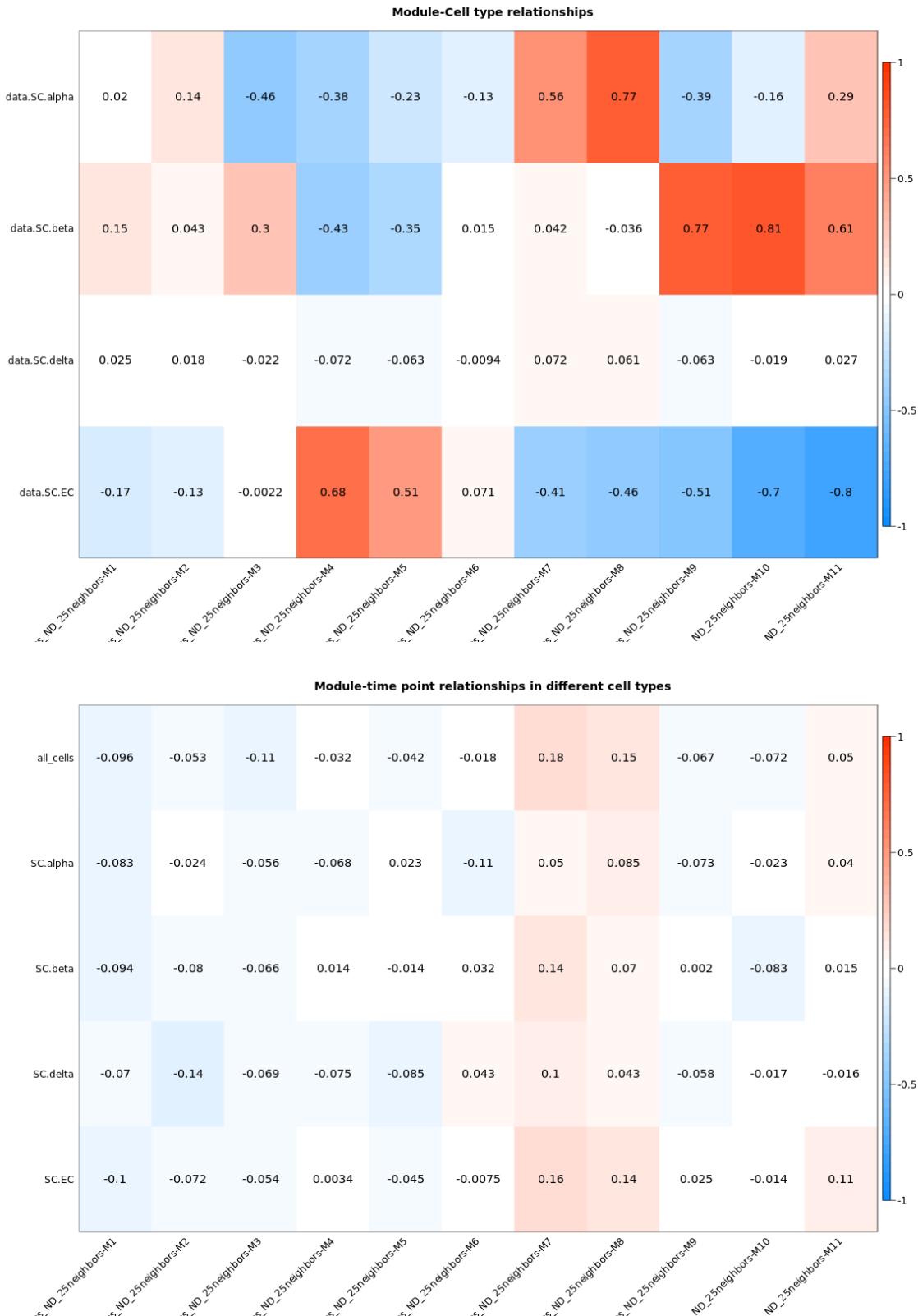
As a supplementary visualization, we can look at how correlated modules are with each other based on their MEs:



### ▼ Module trait correlation

I binarized each cell type in a 1 versus rest manner and calculated correlations with MEs across cells. For example, if the cell type is  $\alpha$  cells, each cell gets labeled a 1 if it is an  $\alpha$  cell and a 0 if it is some other cell type. I can then correlate this binary vector with ME vectors and get a module-trait correlation for  $\alpha$  cells across modules (last row of below matrix)

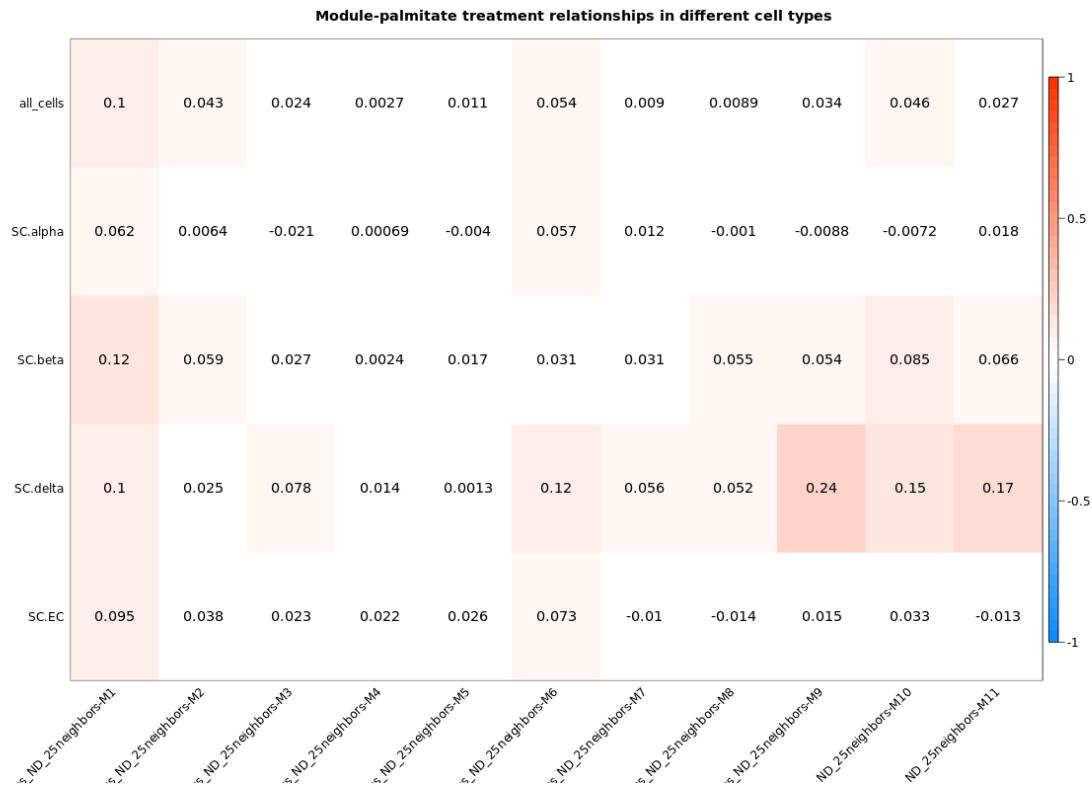
- Correlation of MEs with cell type



I can also look at correlations with treatments of the cells. To do this I binarized all “condition” metadata in a pairwise fashion, forcing untreated (`unt`) to come first and serve as 0. For example, if I am interested in the cytokine treatment (`cyt`), I label all

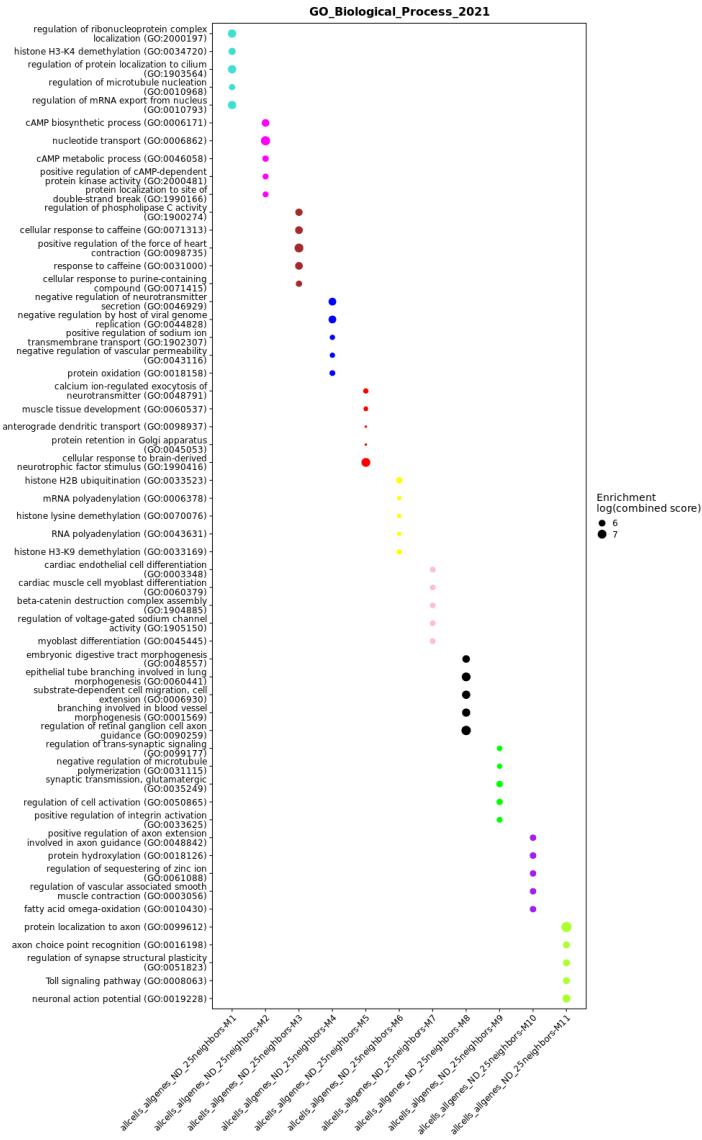
cells that are treated with cytokines as 1 and label all cells that are untreated as 0. The rest are not considered and I calculate correlations on this subset of cells.

#### Correlation of MEs with treatment



#### ▼ Gene set enrichment

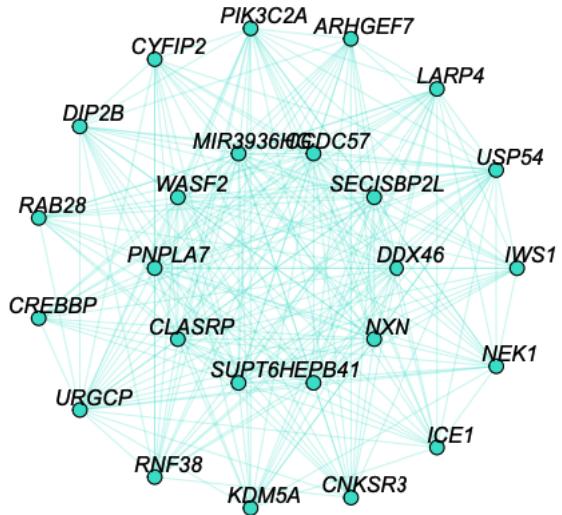
We can run gene set enrichment analysis with EnrichR for a given set of [available gene sets \(libraries\)](#) provided by the package. We can then get the top 5 for each module and plot them as a DotPlot (I also have per module barplots as supplementary but visualizing this way provides a better summary).



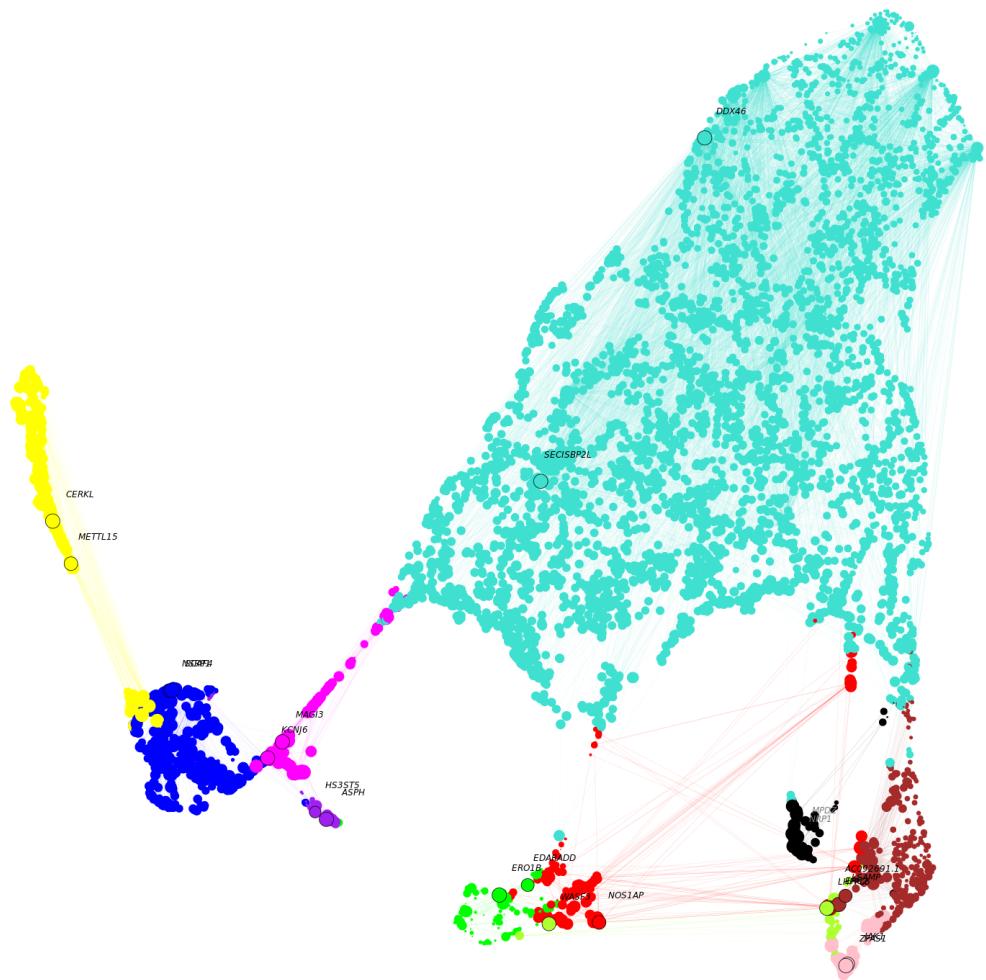
## ▼ Network analysis

We can visualize the “hub” genes of each module as a network with the top 10 on the inside and the top 15 on the outside. Edges represent the correlation strength for genes in this module. This really doesn't do a whole lot except for provide a visual way to look at module membership

### allcells\_allgenes\_ND\_25neighbors-M1

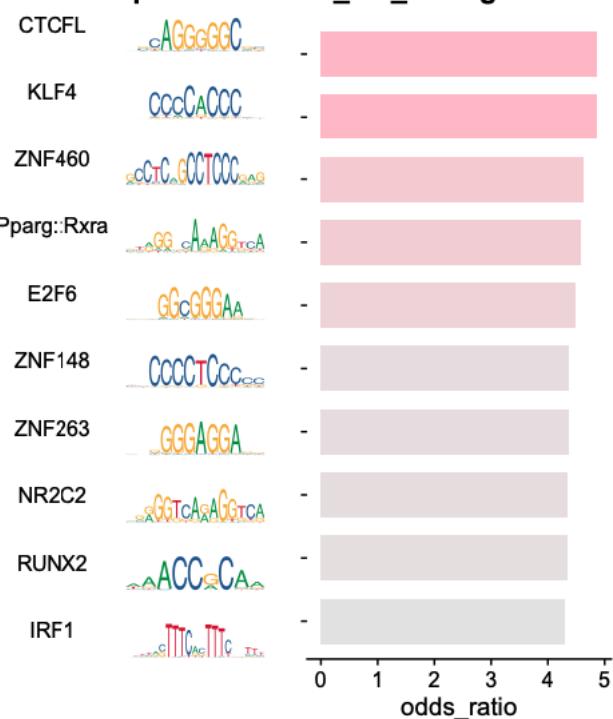


We can also try to combine the module networks into one plot (top) and generate a UMAP visualization of the TOM (bottom)

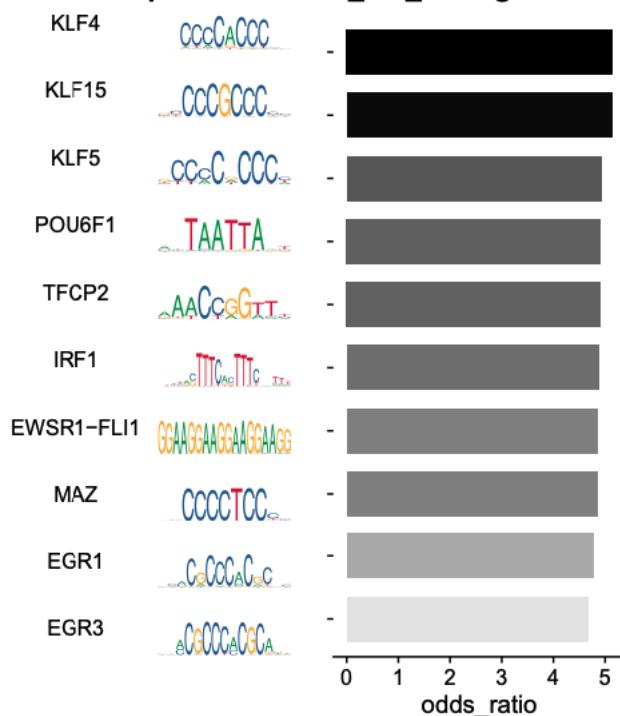


## ▼ Motif analysis

### Motif overlaps with allcells\_ND\_25neighbors-M1



### Motif overlaps with allcells\_ND\_25neighbors-M7



### ▼ NormalizeData SC- $\alpha$ , SC- $\beta$ , SC-EC (no $\delta$ ) cell analysis across all genes with 25 neighbors ( allcells\_ND\_25neighbors )

Active assay: RNA

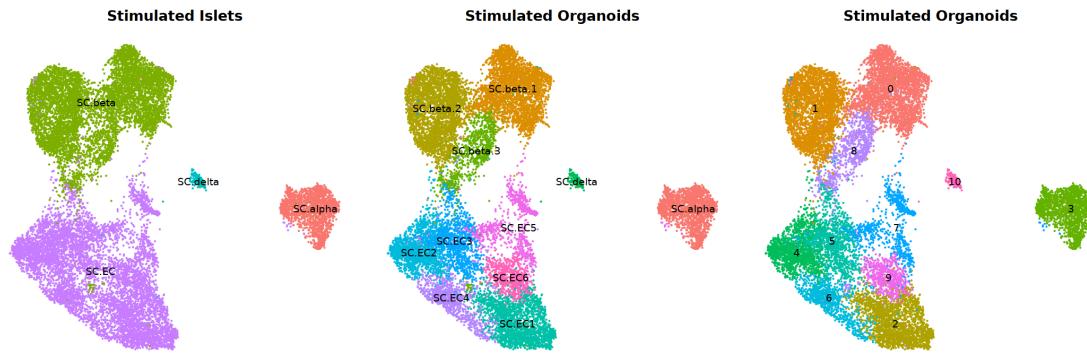
Cell types under consideration: `SC.α`, `SC.β`, `SC.EC` (in the last run, we only had 18 metacells created for the δ-cell type, so I removed them for this run)

Fraction of cells to keep a gene: `0.05`

Number of neighbors for metacell construction: `25`

## ▼ Network Construction

- Starting with `16342` cells across `28796` genes in the integrated multiome data

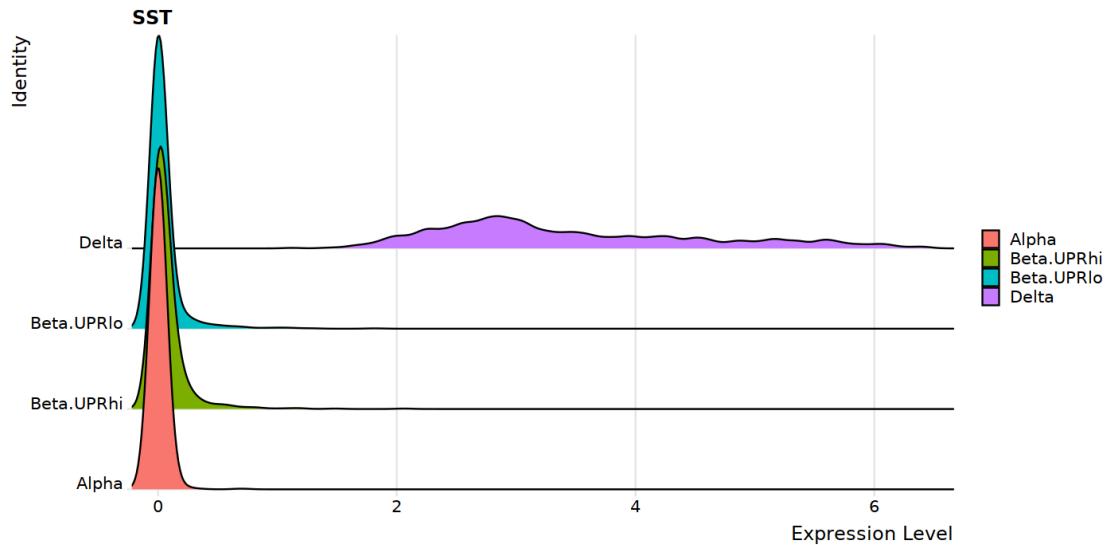


- Keeping only α, β and δ cells from original processed Seurat `.rds`

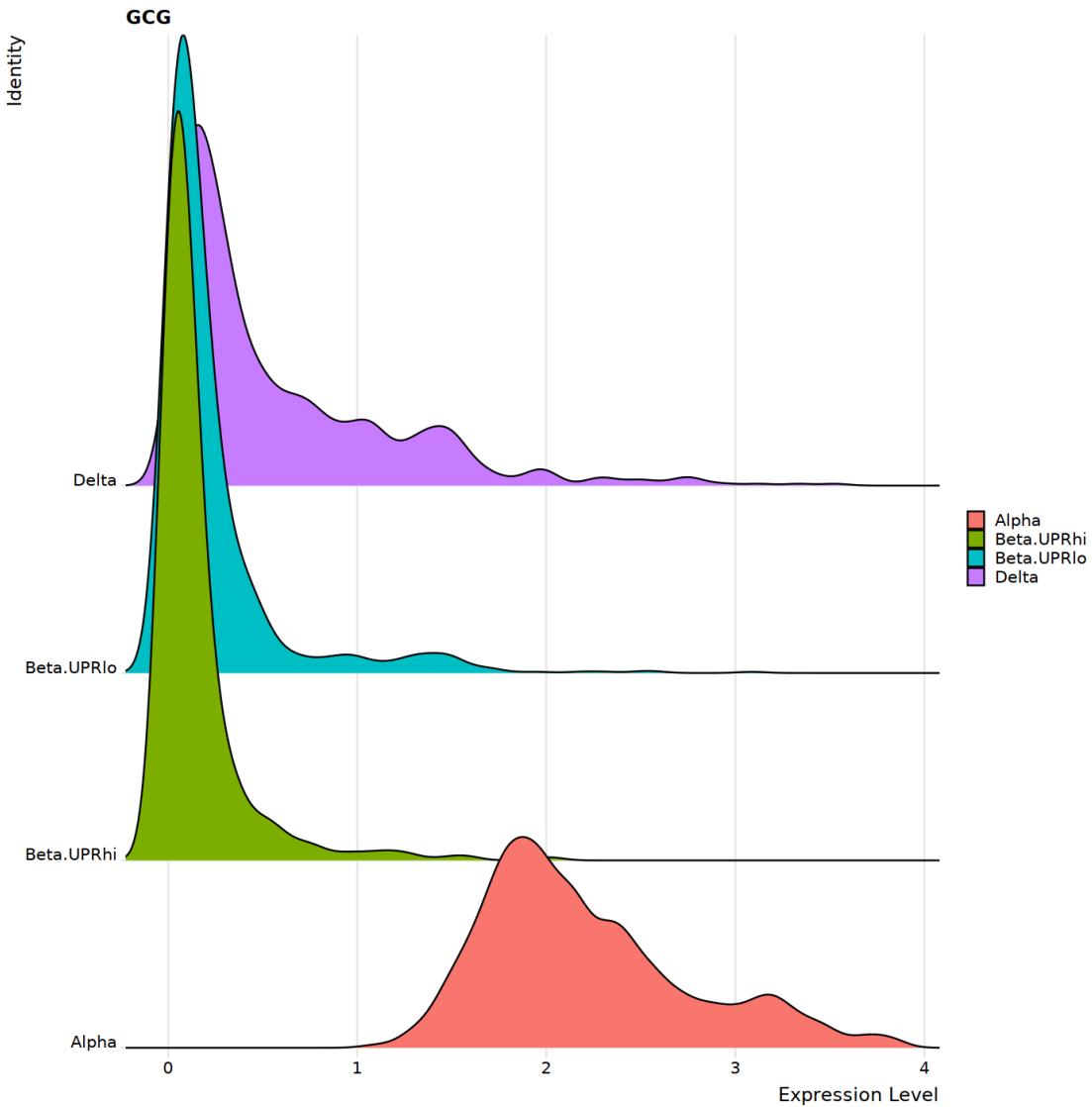
| SC.alpha          | SC.beta           | SC.delta         | SC.EC             |
|-------------------|-------------------|------------------|-------------------|
| <code>1821</code> | <code>7042</code> | <code>186</code> | <code>7293</code> |

- Using a fraction of `0.05` based on the `counts` slot of the active assay in `SetupForWGCNA` function to keep only genes expressed in at least 5% of these cells
  - `10832` genes remain after this filter
- Created metacells using top `25` nearest neighbors within a cell type (label from processed object). Used the `counts` to create these metacells in `RNA` assay and only allowed them to share `10` cells in common at most. Minimum cells by default is `100` and the max number of metacells to construct per cell type is `1000`. Can normalize this metacell object using Seurat's default function that uses a `scale.factor` of `10000` and by default takes the natural log with a pseudo-count
  - Have a total of `2597` cells after this

| SC.alpha         | SC.beta           | SC.delta        | SC.EC             |
|------------------|-------------------|-----------------|-------------------|
| <code>579</code> | <code>1000</code> | <code>18</code> | <code>1000</code> |

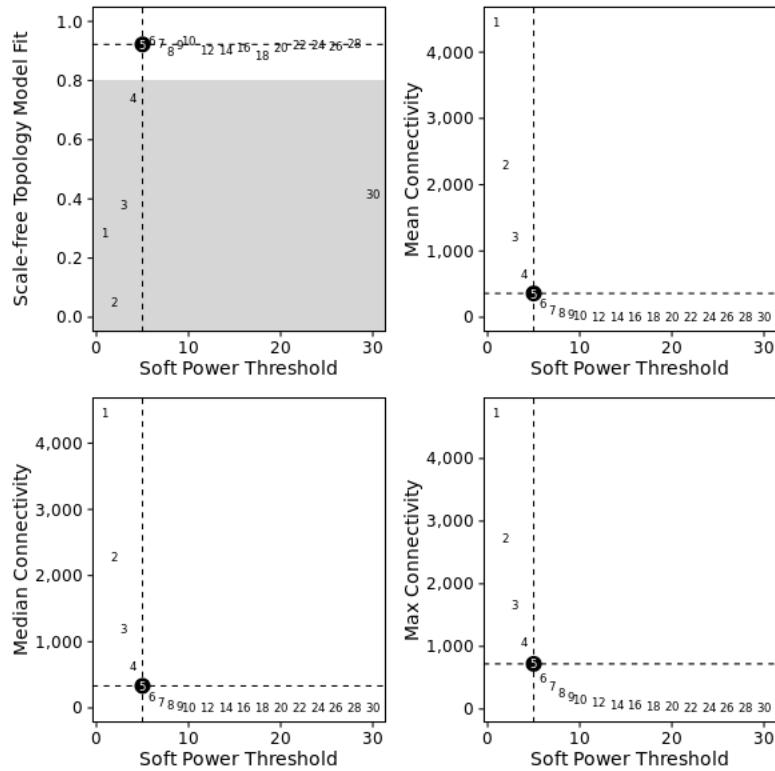


Example of normalized metacell expression distribution for  $\delta$ -cell marker



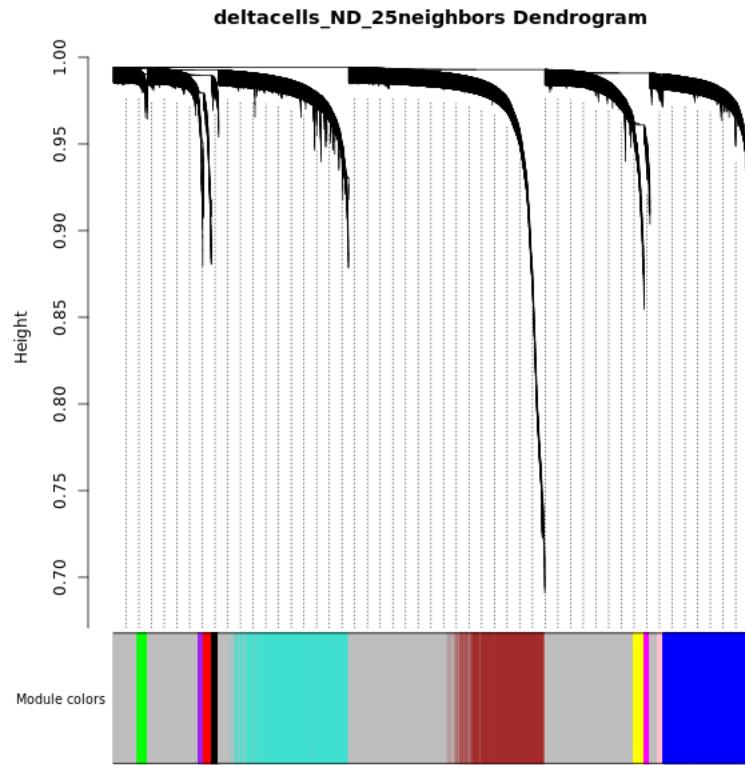
Example of normalized metacell expression distribution for  $\alpha$ -cell marker

- Using the `SetDataExpr` function, selected the normalized metacell data from the `data` slot of the stored metacell Seurat object.
- Selected a soft-power threshold of `5` for downstream network construction



| Power | SFT.R.sq   | slope     | truncated.R.sq | mean.k.   | median.k. | max.k.    |
|-------|------------|-----------|----------------|-----------|-----------|-----------|
| <dbl> | <dbl>      | <dbl>     | <dbl>          | <dbl>     | <dbl>     | <dbl>     |
| 1     | 0.28380288 | 18.929000 | 0.9764219      | 4464.0449 | 4469.5653 | 4741.5444 |
| 2     | 0.05257345 | -3.759070 | 0.8533002      | 2309.7087 | 2291.4523 | 2737.7290 |
| 3     | 0.38226184 | -6.369994 | 0.7158130      | 1218.2877 | 1190.1127 | 1676.4506 |
| 4     | 0.73991759 | -6.055871 | 0.8634186      | 655.2120  | 626.0709  | 1076.8130 |
| 5     | 0.92217560 | -5.502160 | 0.9815870      | 359.6580  | 333.4544  | 721.3339  |
| 6     | 0.93276135 | -4.593437 | 0.9843583      | 201.8194  | 179.3330  | 502.3328  |

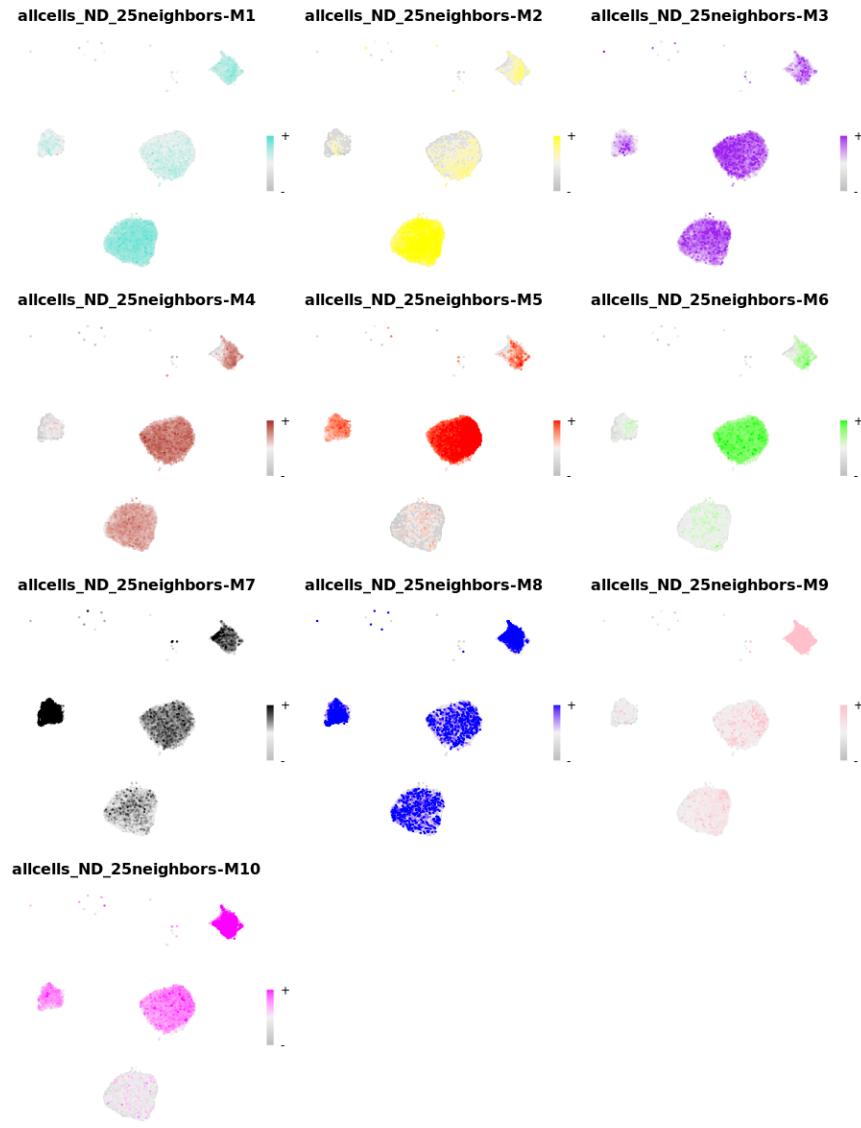
- Using the normalized metacell expression matrix, performed WGCNA using soft power of 5. By default this builds a signed network using Pearson correlation.



- Did not include a `group_by_vars` argument in the `ModuleEigengenes` function, which look like stops Harmony batch correction from running. The module eigengenes for each module are the 1st PC of the subsetted matrix
- Did not include `group_by` argument in the `ModuleConnectivity` function (which seems to make it take a long time —> ended up taking ~30 minutes). Computes a correlation for every gene in every module with its module eigengene. We still have 3000+ cells and 8000+ genes, so this is a heavy computation step

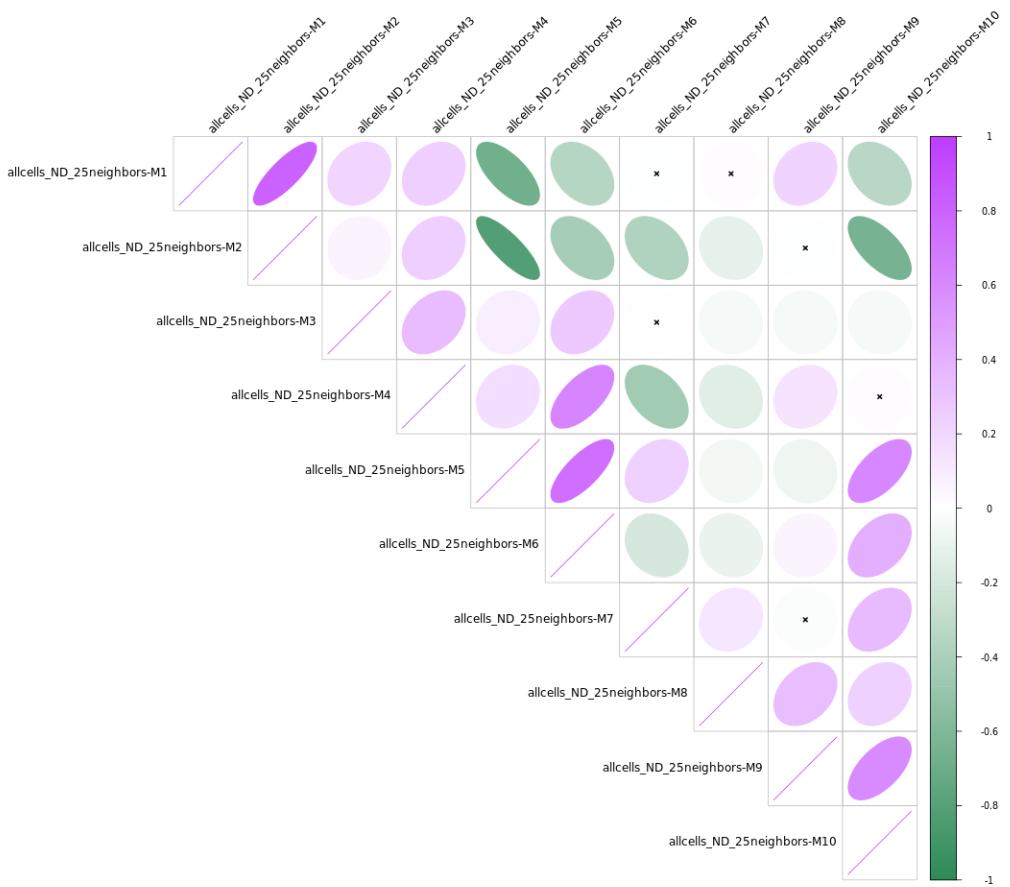
## ▼ Basic Visualization

Plotting the raw MEs (PCA yields a sample x PC matrix for each module, I can overlay these onto the previous UMAP plot:



It looks like I have a few cell type specific modules. M1, M2, M3 seem to be high in Alpha cells. M4, M5, M6 are highest in the Beta.UPR.lo cells. M7 is high in the Beta.UPR.hi cells. M8, M9, M10 are high in the Delta cells

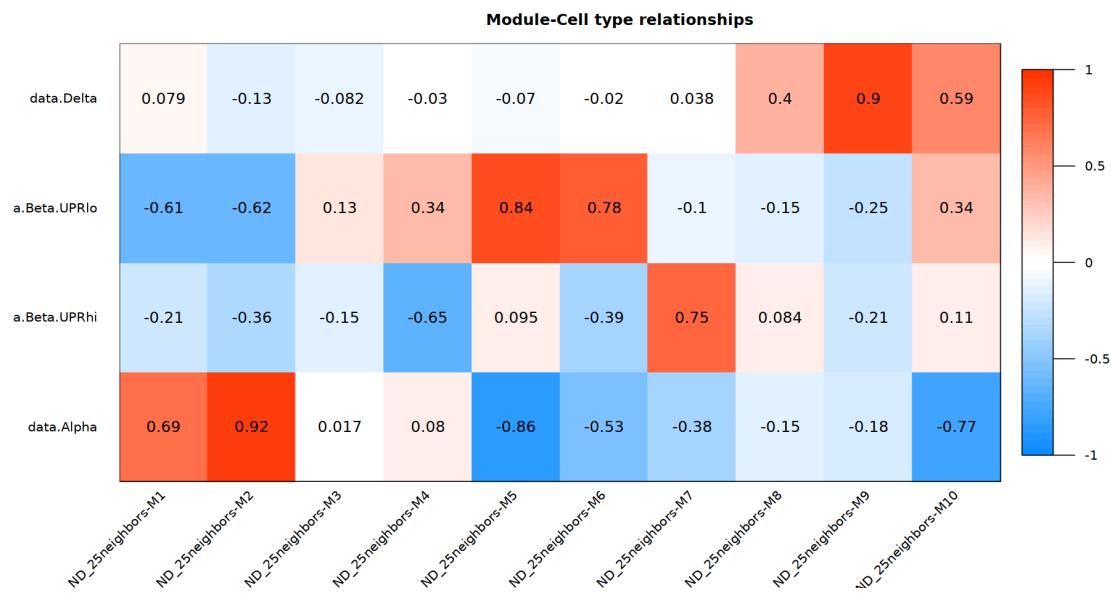
As a supplementary visualization, we can look at how correlated modules are with each other based on their MEs:



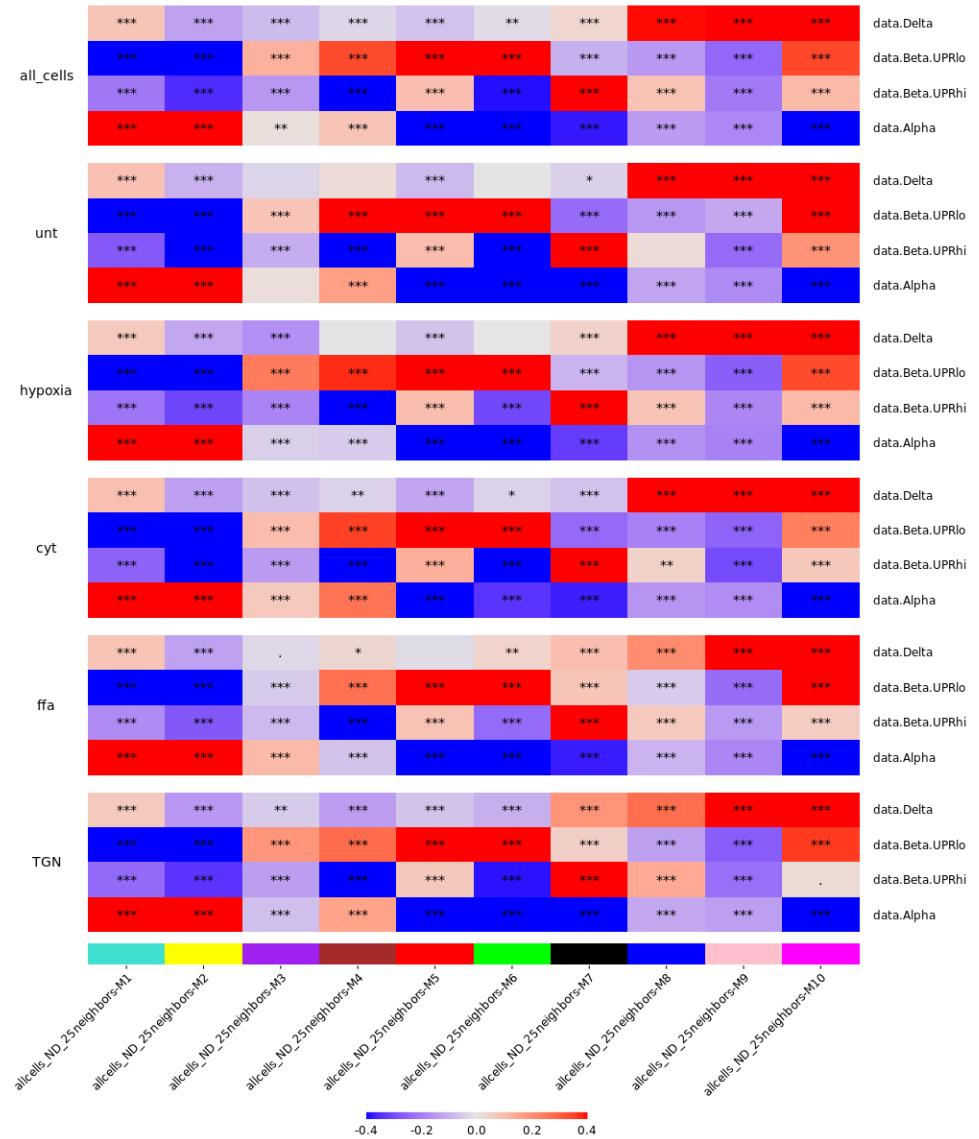
### ▼ Module trait correlation

I binarized each cell type in a 1 versus rest manner and calculated correlations with MEs across cells. For example, if the cell type is  $\alpha$  cells, each cell gets labeled a 1 if it is an  $\alpha$  cell and a 0 if it is some other cell type. I can then correlate this binary vector with ME vectors and get a module-trait correlation for  $\alpha$  cells across modules (last row of below matrix)

- Correlation of MEs with cell type



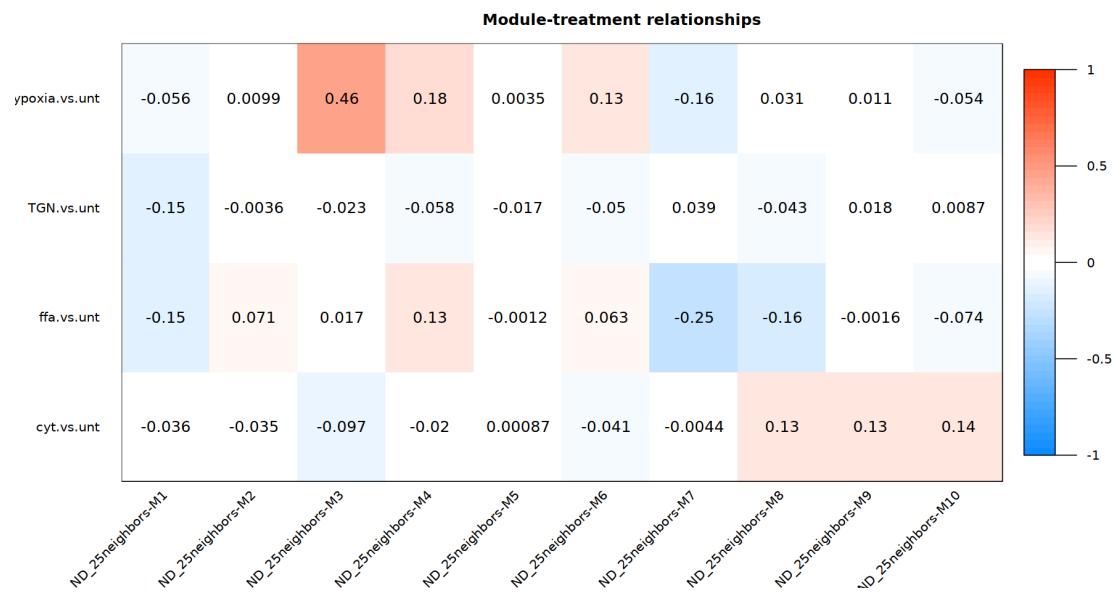
Across all cells. Tracks with the above UMAP visualization, where M1 and M2 look alpha cell specific, M4-M6 look Beta cells with low UPR specific, M7 looks specific to beta cells with UPR high and M8-M10 look delta cell specific.



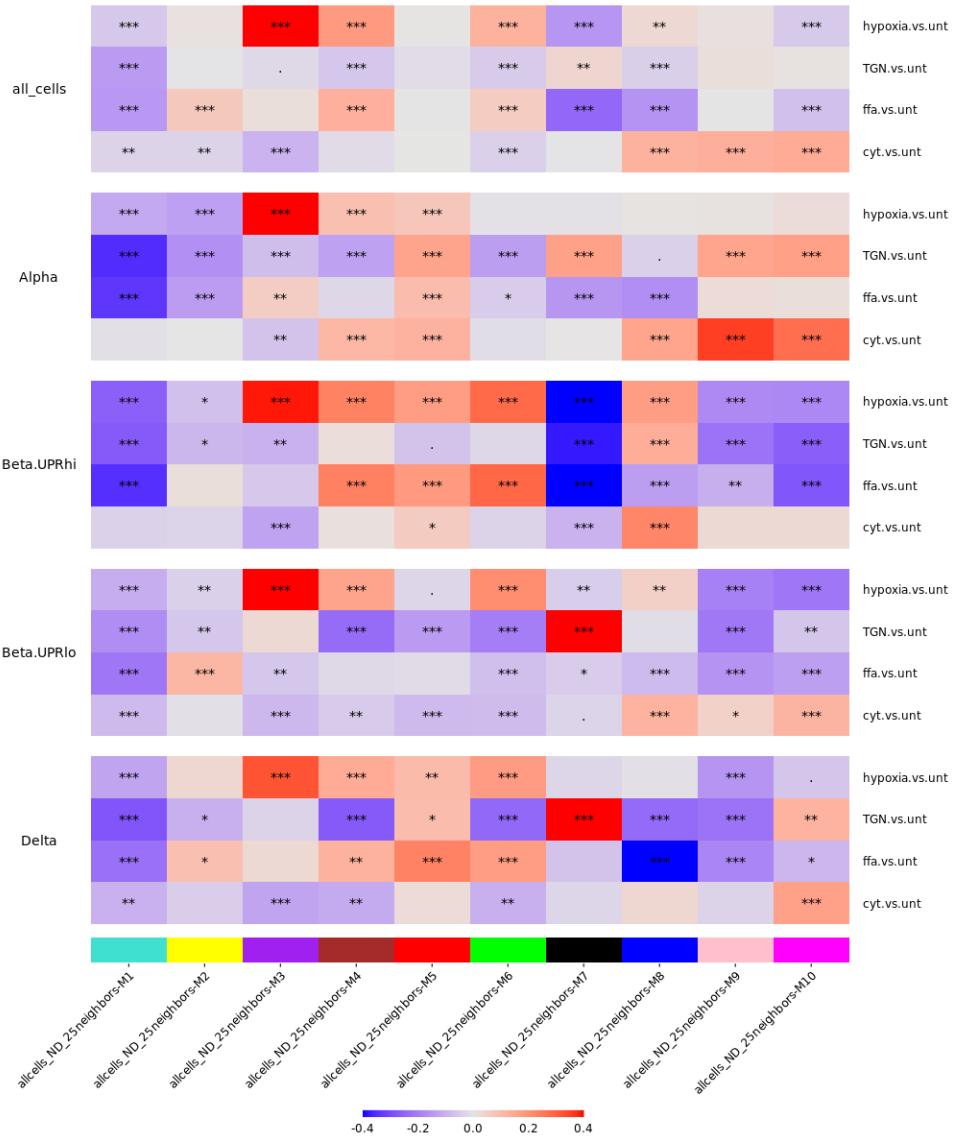
Across all cells and broken up into treatments. The first set of rows are the same as the above plot. I looked briefly to see if any correlations changed substantially depending on the treatment. Most of the changes seem subtle and the majority of signal is tied to the cell type.

I can also look at correlations with treatments of the cells. To do this I binarized all “condition” metadata in a pairwise fashion, forcing untreated (`unt`) to come first and serve as 0. For example, if I am interested in the cytokine treatment (`cyt`), I label all cells that are treated with cytokines as 1 and label all cells that are untreated as 0. The rest are not considered and I calculate correlations on this subset of cells.

#### Correlation of MEs with treatment



Across all cells. Here we see that hypoxia seems to be most associated with M3, which didn't have a strong signal for any cell type (see above). There are some weaker correlations for other treatments, interestingly, cytokine treatment is slightly positively correlated with M8-M10, which were delta cell modules



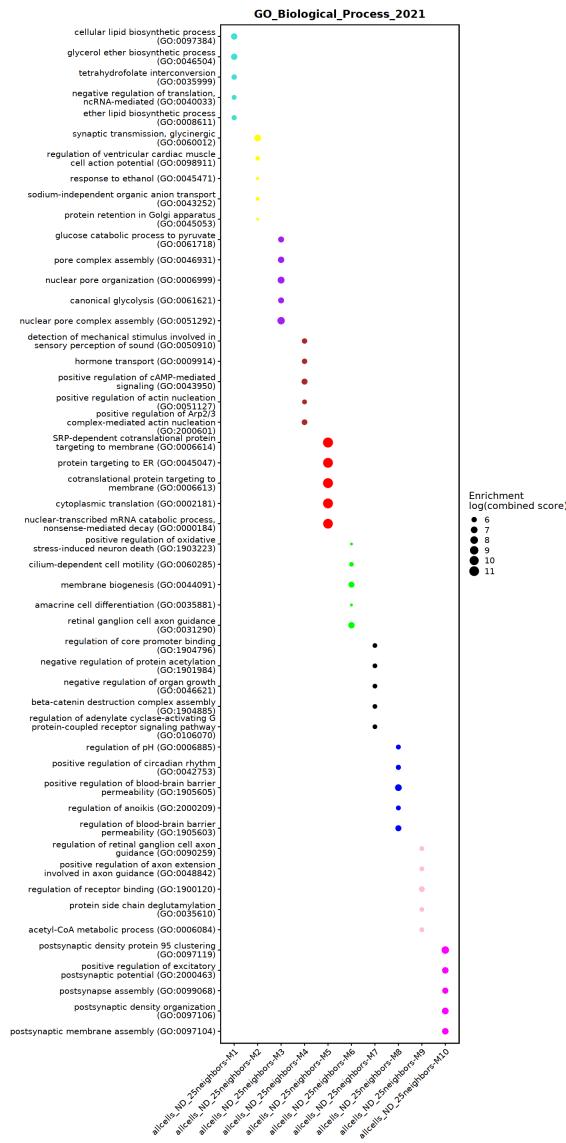
Across all cells broken up by cell type. Been working on teasing apart this. It does look like M7 (black) that was associated with UPRhi beta cells is important for distinguishing Beta.UPRhi from the rest

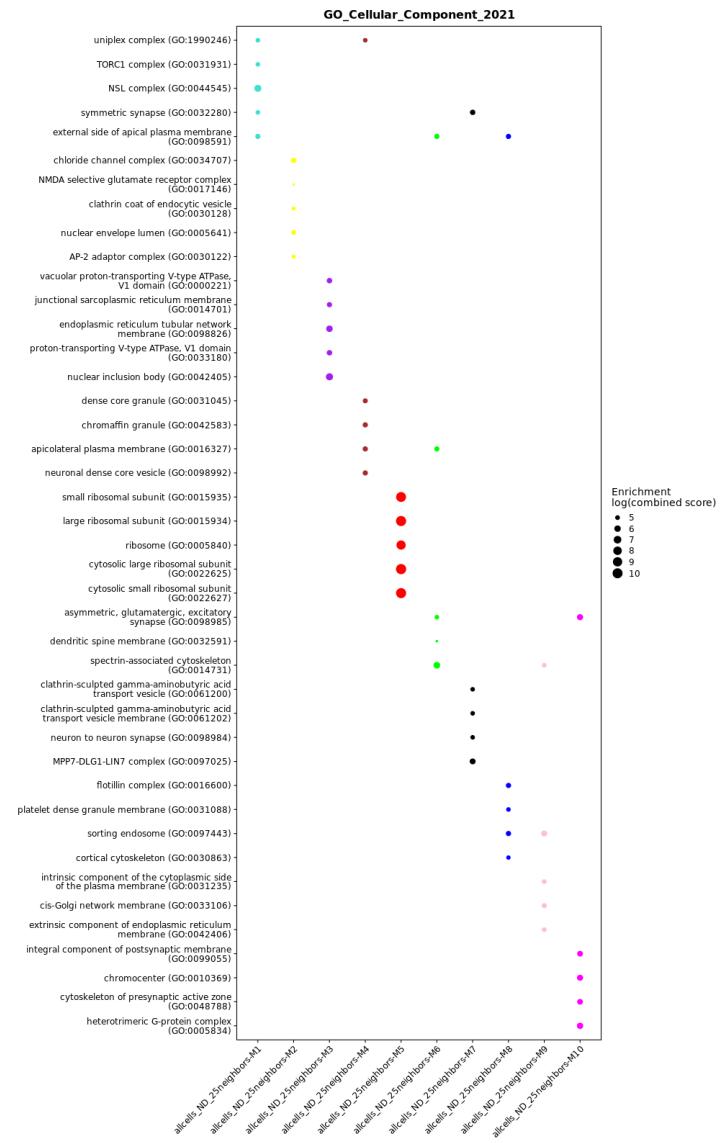
Proportion of treatments in each cell type (how the different treatments are divided up for a given cell type)

|            | cyt   | ffa   | hypoxia | TGN   | unt   |
|------------|-------|-------|---------|-------|-------|
| Alpha      | 0.170 | 0.220 | 0.330   | 0.170 | 0.110 |
| Beta.UPRhi | 0.320 | 0.100 | 0.250   | 0.160 | 0.180 |
| Beta.UPRlo | 0.170 | 0.210 | 0.340   | 0.170 | 0.110 |
| Delta      | 0.270 | 0.160 | 0.330   | 0.170 | 0.077 |

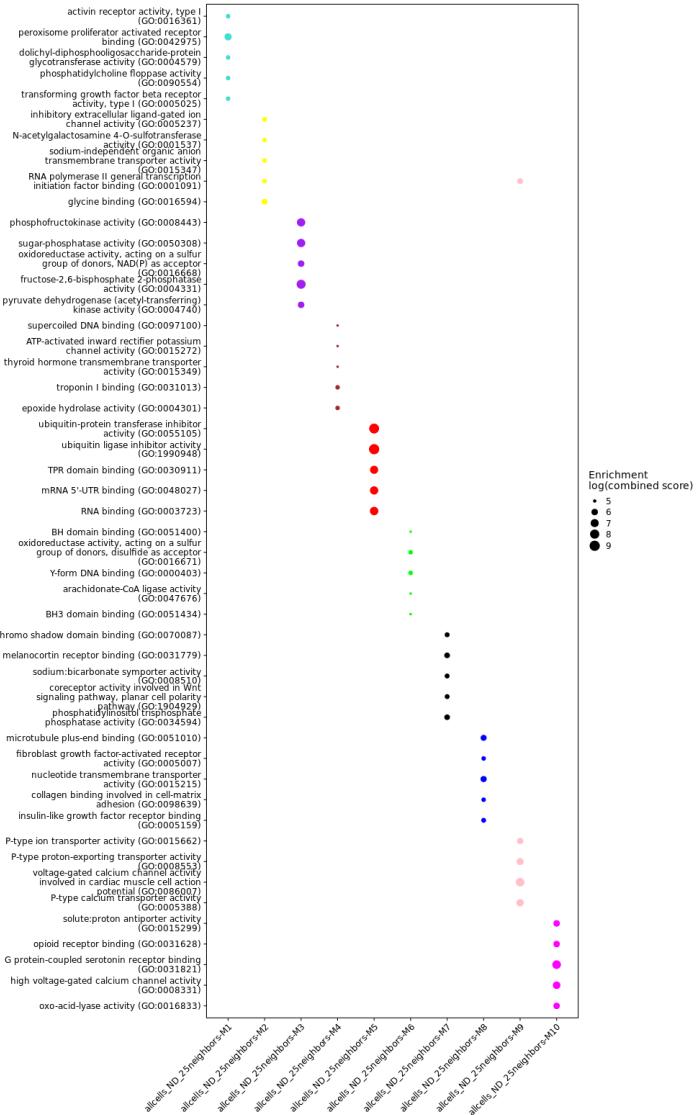
## ▼ Gene set enrichment

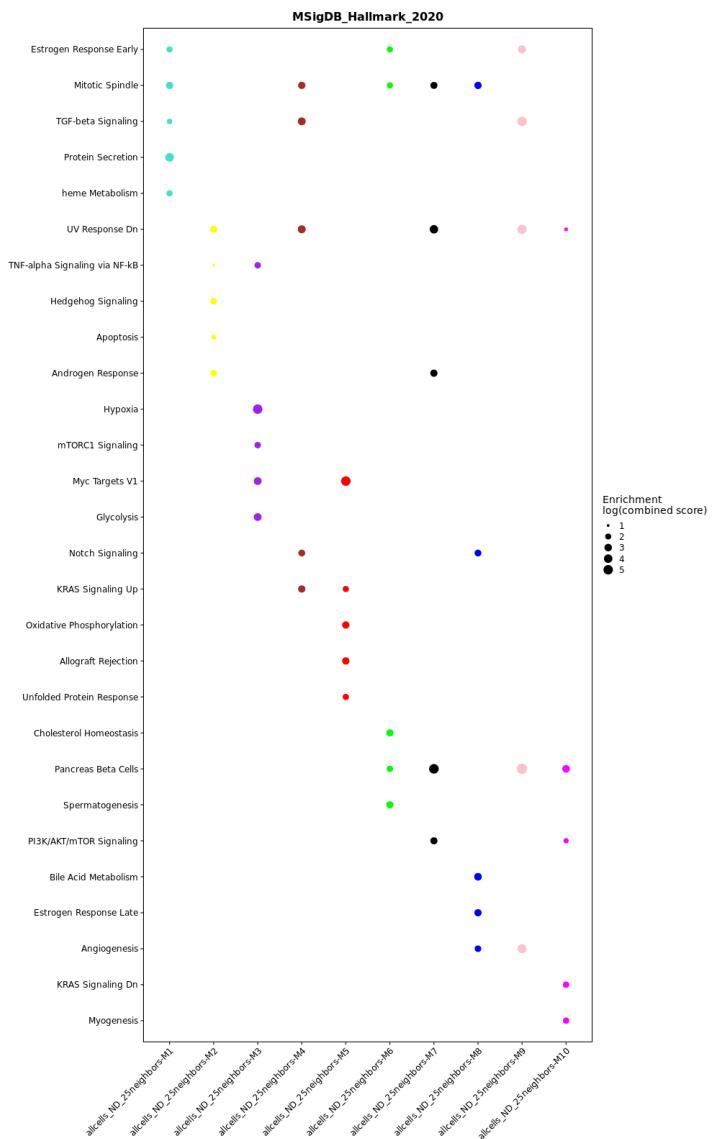
We can run gene set enrichment analysis with EnrichR for a given set of [available gene sets \(libraries\)](#) provided by the package. We can then get the top 5 for each module and plot them as a DotPlot (I also have per module barplots as supplementary but visualizing this way provides a better summary).



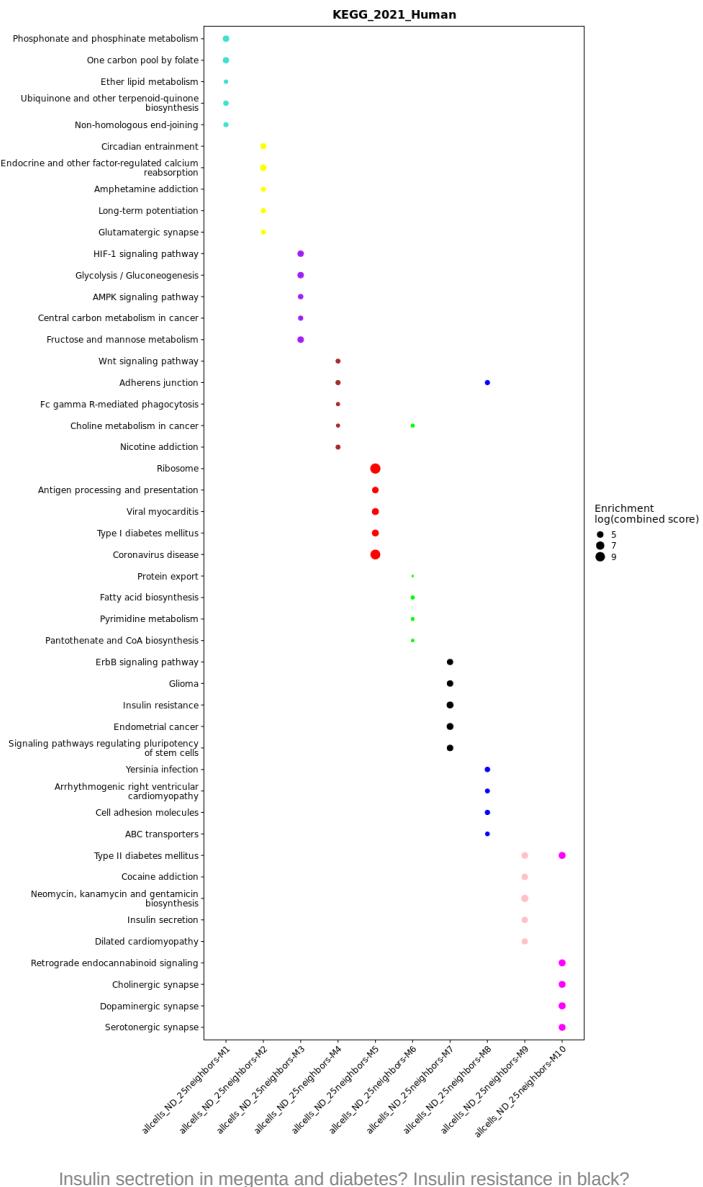


### GO\_Molecular\_Function\_2021





Pretty clear that M3 is a hypoxia specific gene set

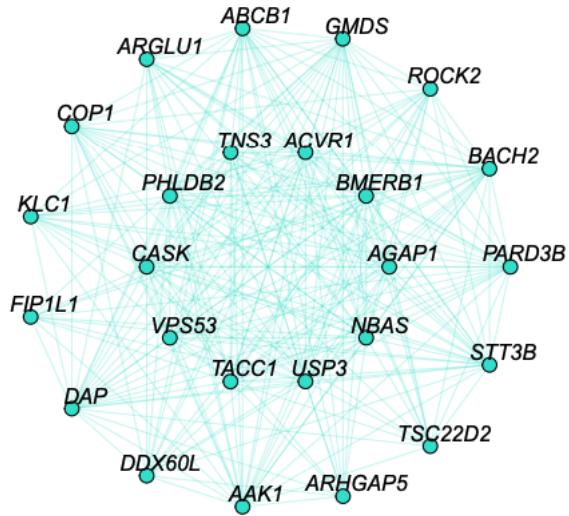


## ▼ Network analysis

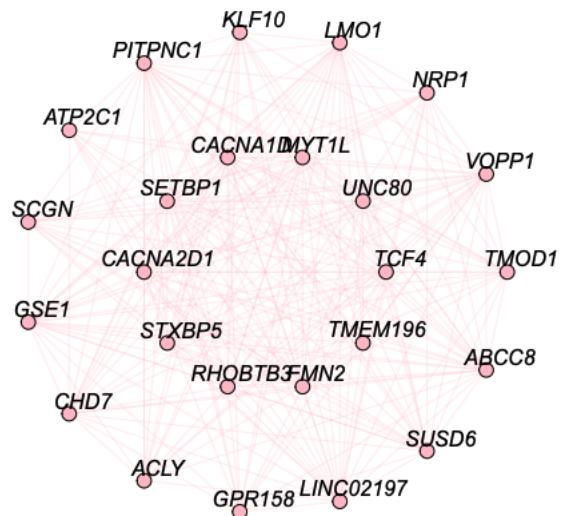
We can visualize the “hub” genes of each module as a network with the top 10 on the inside and the top 15 on the outside. Edges represent the correlation strength for genes in this module. This really doesn’t do a whole lot except for provide a visual way to look at module membership

@Adam Klie TODO: Add the rest of the hub gene plots here

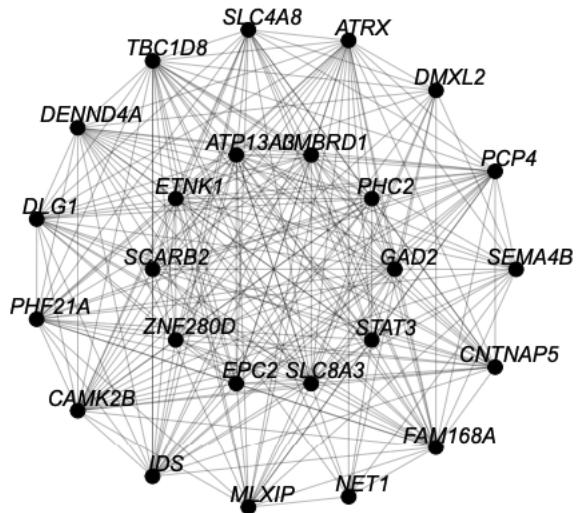
allcells\_ND\_25neighbors-M1



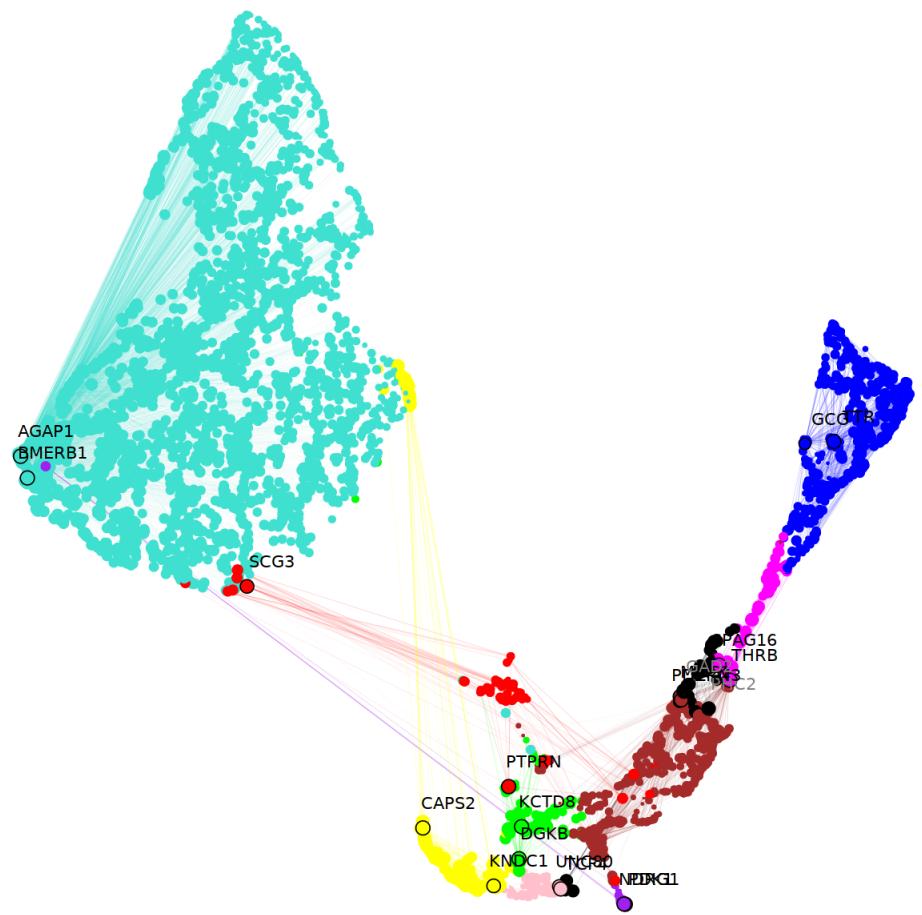
allcells\_ND\_25neighbors-M9



allcells\_ND\_25neighbors-M7



We can also try to combine the module networks into one plot (top) and generate a UMAP visualization of the TOM (bottom)

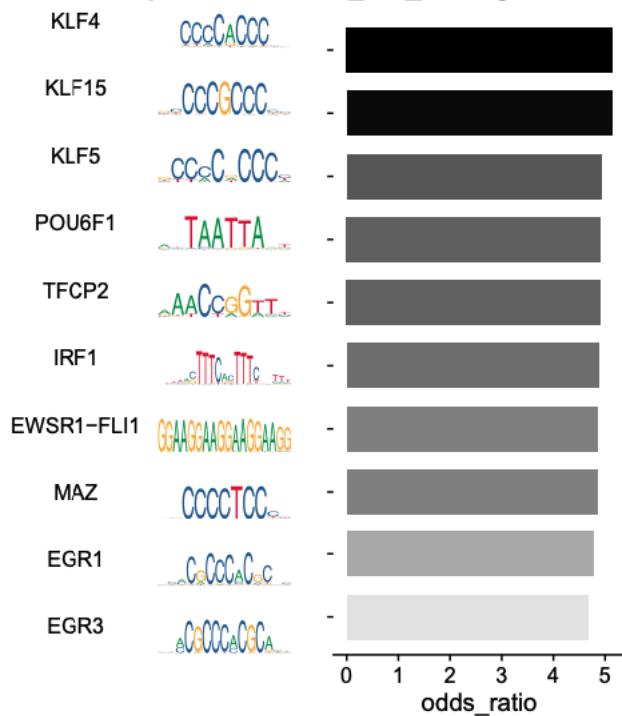


▼ Motif analysis

### Motif overlaps with allcells\_ND\_25neighbors-M1



### Motif overlaps with allcells\_ND\_25neighbors-M7



### ▼ `NormalizeData` SC- $\beta$ cell analysis across all genes with 25 neighbors (`betacells_ND_25neighbors`)

Active assay: `RNA`

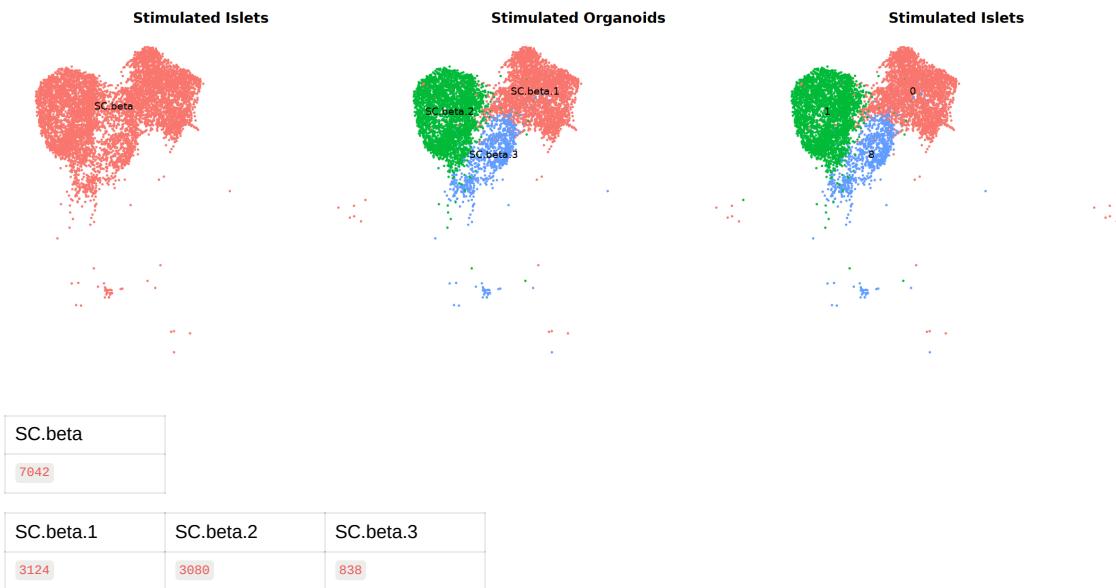
Cell types under consideration: `SC.β`

Fraction of cells to keep a gene: `0.05`

Number of neighbors for metacell construction: `25`

## ▼ Network Construction

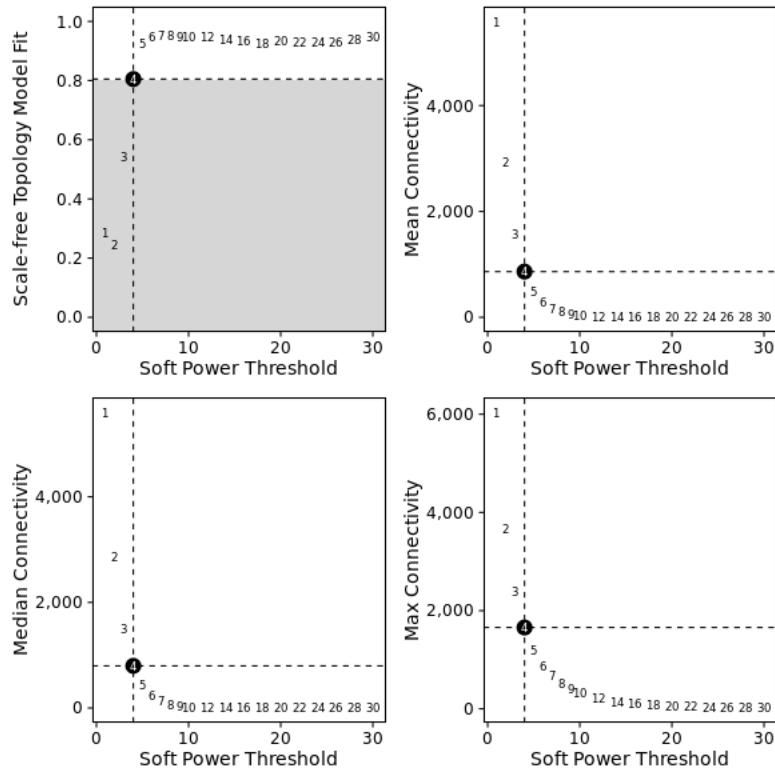
- Starting with `16342` cells across `36601` genes in the integrated multiome data
- Keeping only  $\beta$  cells from original processed Seurat `.rds`



- Using a fraction of `0.05` based on the `counts` slot of the active assay in `SetupForWGCNA` function to keep only genes expressed in at least 5% of these cells
  - `11003` genes remain after this filter
- Created metacells using top `25` nearest neighbors **that can go across subclusters**. Used the `counts` to create these metacells in the `RNA` assay and only allowed them to share `10` cells in common at most. Minimum cells by default is `100` and the max number of metacells to construct per cell type is `1000`. Can normalize this metacell object using Seurat's default function that uses a `scale.factor` of `10000` and by default takes the natural log with a pseudo-count
  - Have a total of `2580` cells after this

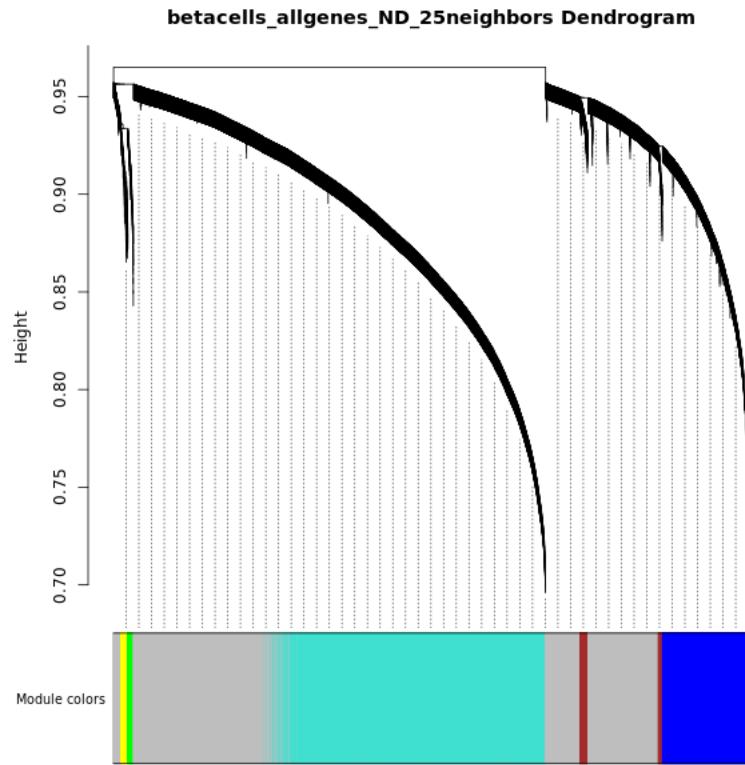
|      |
|------|
| SC.β |
| 2580 |

- Using the `SetDataExpr` function, selected the normalized metacell data from the `data` slot of the stored metacell Seurat object.
- Selected a soft-power threshold of `5` for downstream network construction



| A data.frame: 6 x 7 |          |           |                |           |           |           |
|---------------------|----------|-----------|----------------|-----------|-----------|-----------|
| Power               | SFT.R.sq | slope     | truncated.R.sq | mean.k.   | median.k. | max.k.    |
| <dbl>               | <dbl>    | <dbl>     | <dbl>          | <dbl>     | <dbl>     | <dbl>     |
| 1                   | 1        | 0.2881371 | 10.968859      | 0.9517153 | 5590.6868 | 5603.3172 |
| 2                   | 2        | 0.2452424 | -5.008839      | 0.8479542 | 2922.7231 | 2875.2103 |
| 3                   | 3        | 0.5450047 | -4.935994      | 0.7817115 | 1568.4551 | 1495.4946 |
| 4                   | 4        | 0.8050741 | -4.358672      | 0.9016088 | 863.3839  | 797.7157  |
| 5                   | 5        | 0.9268446 | -3.850557      | 0.9753876 | 487.5594  | 431.7712  |
| 6                   | 6        | 0.9467369 | -3.339103      | 0.9850075 | 282.5936  | 237.5245  |

- Using the normalized metacell expression matrix, performed WGCNA using soft power of 5. By default this builds a signed network using Pearson correlation.

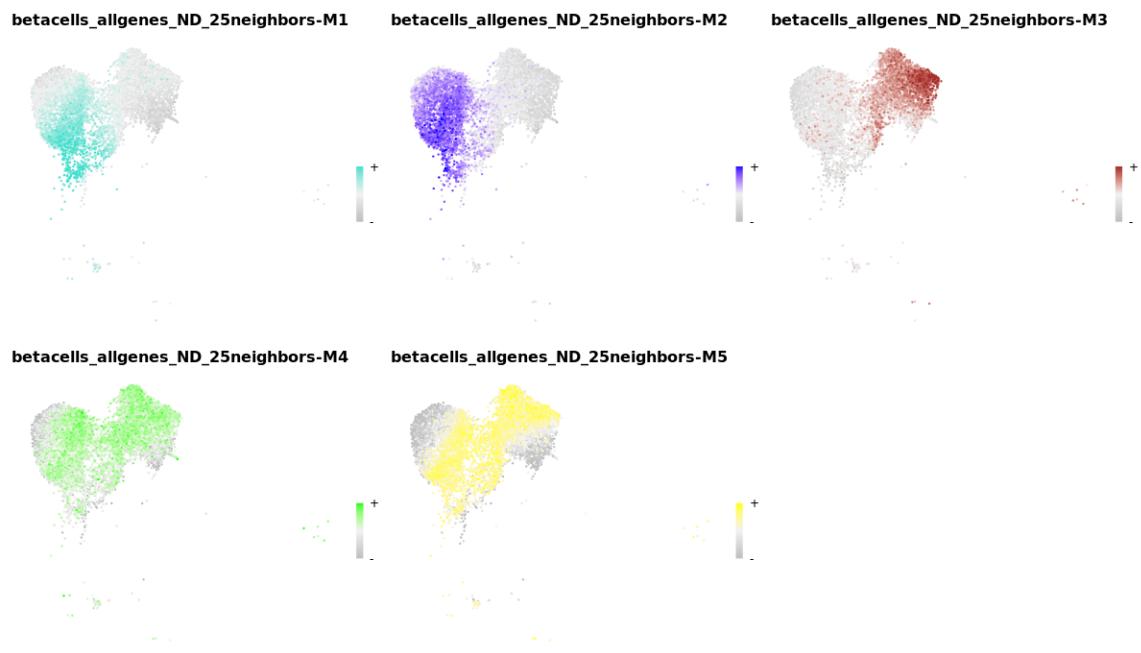


| Module    | blue | brown | green | grey | turquoise | yellow |
|-----------|------|-------|-------|------|-----------|--------|
| Num Genes | 1489 | 210   | 99    | 4453 | 4636      | 116    |

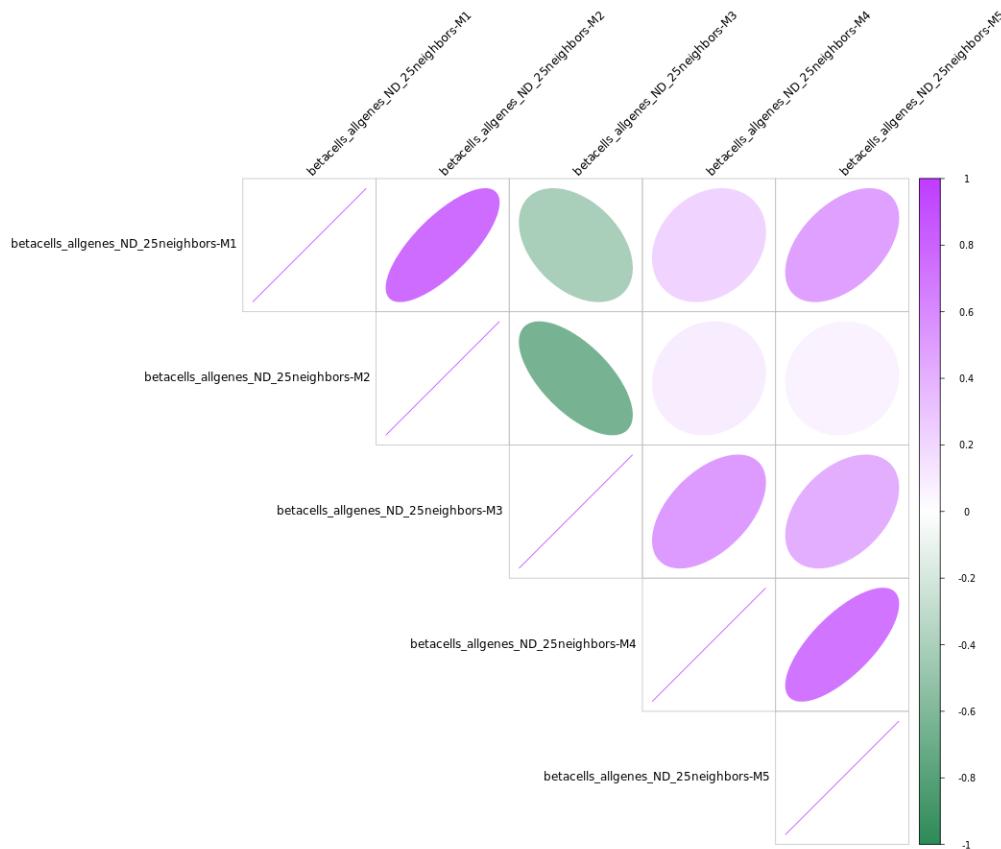
- Did not include a `group_by_vars` argument in the `ModuleEigengenes` function, which look like it stops Harmony batch correction from running. The module eigengenes for each module are the 1st PC of the subsetted matrix
- Did not include `group_by` argument in the `ModuleConnectivity` function (which seems to make it take a long time —> ended up taking ~30 minutes). Computes a correlation for every gene in every module with its module eigengene.

## ▼ Basic Visualization

Plotting the raw MEs (PCA yields a sample x PC matrix for each module, I can overlay these onto the previous UMAP plot:



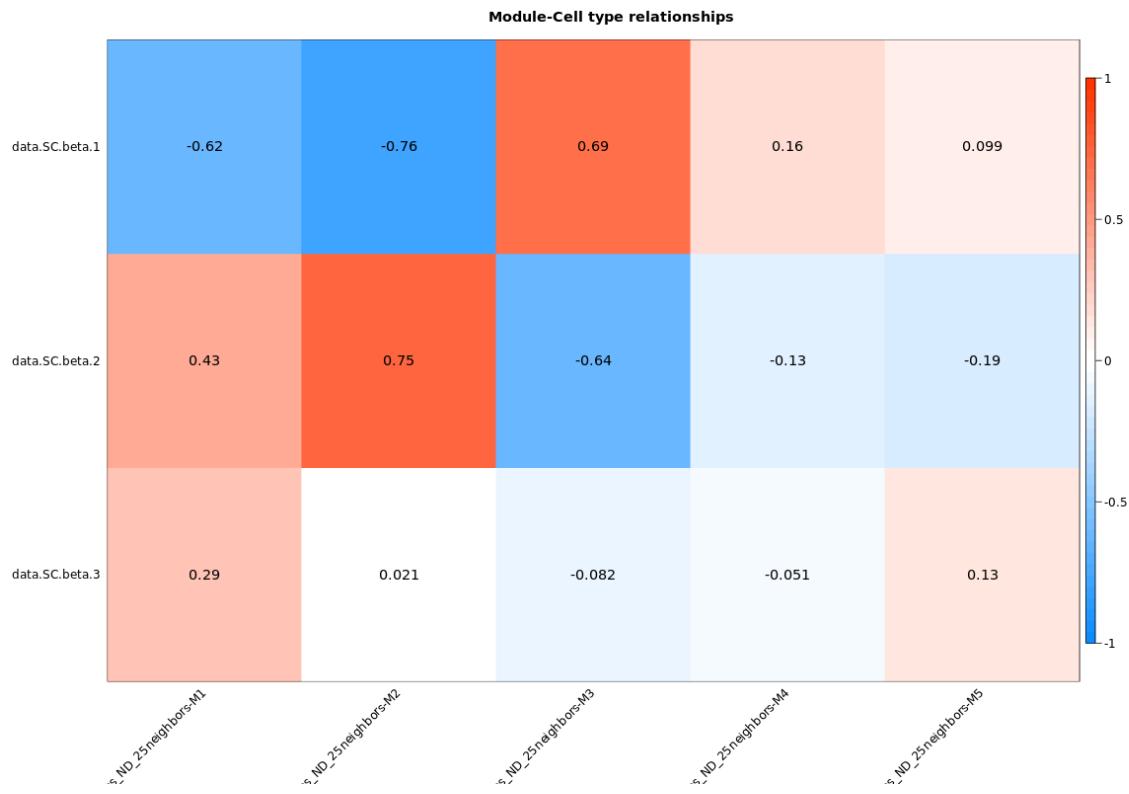
As a supplementary visualization, we can look at how correlated modules are with each other based on their MEs:

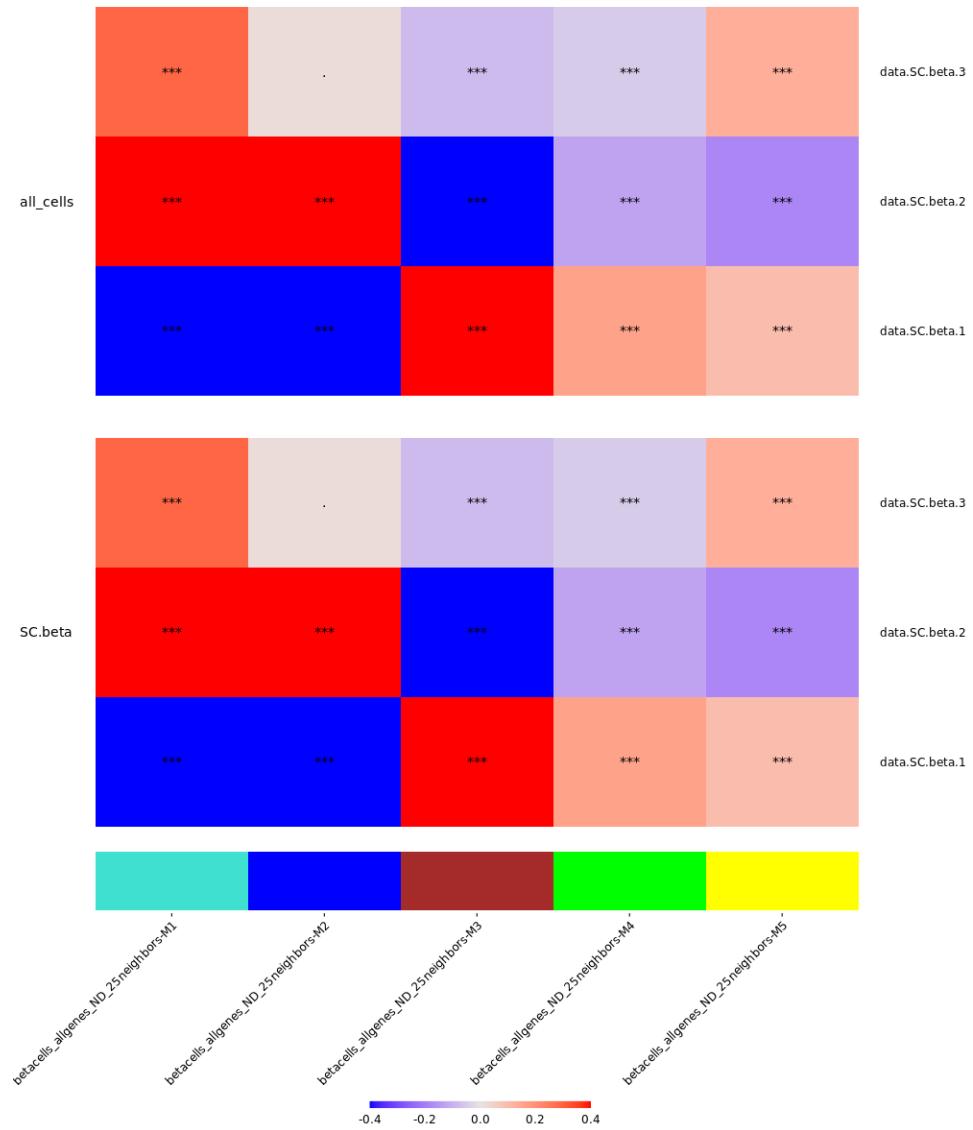


### ▼ Module trait correlation

I binarized each cell type in a 1 versus rest manner and calculated correlations with MEs across cells. For example, if the cell type is  $\alpha$  cells, each cell gets labeled a 1 if it is an  $\alpha$  cell and a 0 if it is some other cell type. I can then correlate this binary vector with ME vectors and get a module-trait correlation for  $\alpha$  cells across modules (last row of below matrix)

- Correlation of MEs with cell type

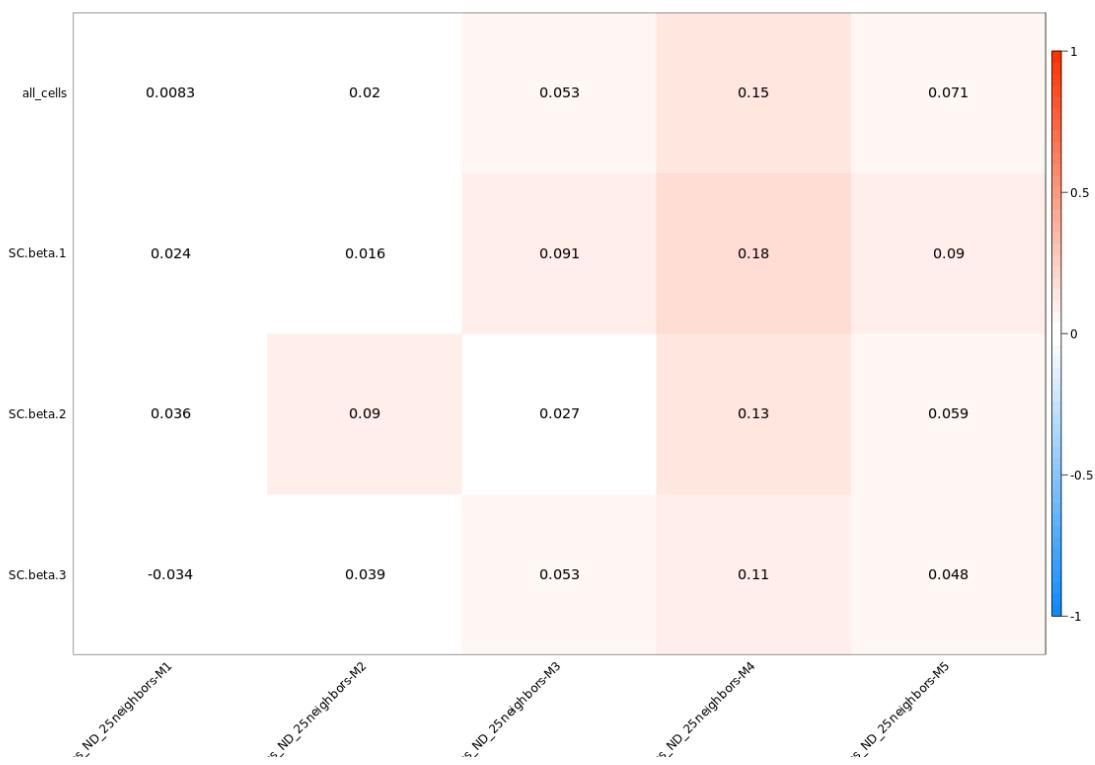




I can also look at correlations with treatments of the cells. To do this I binarized all “condition” metadata in a pairwise fashion, forcing untreated ( `unt` ) to come first and serve as 0. For example, if I am interested in the cytokine treatment ( `cyt` ), I label all cells that are treated with cytokines as 1 and label all cells that are untreated as 0. The rest are not considered and I calculate correlations on this subset of cells.

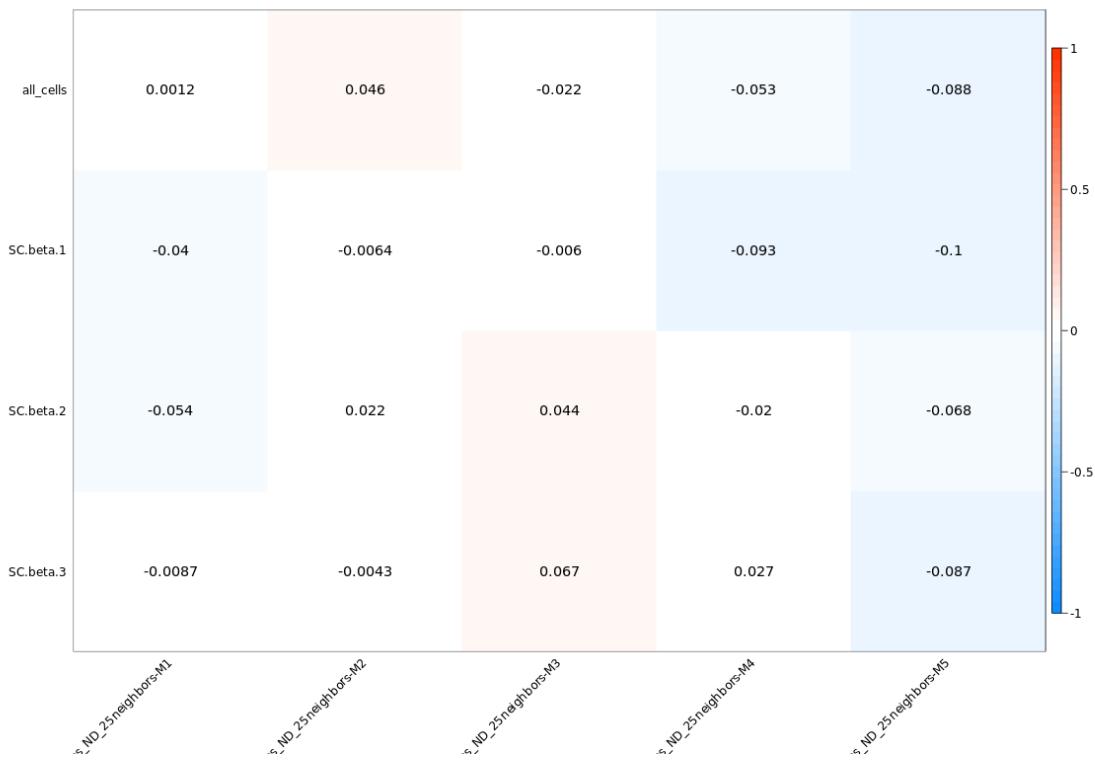
#### Correlation of MEs with treatment

**Module-palmitate treatment relationships in different cell types**



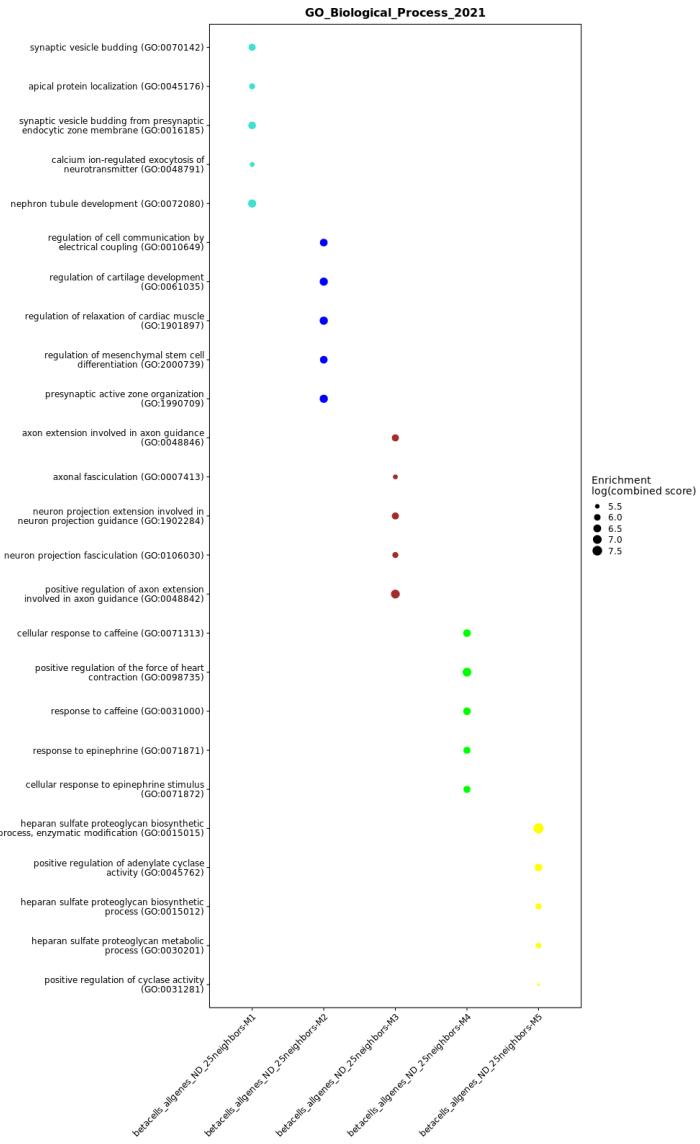
Correlation of MEs with timepoint

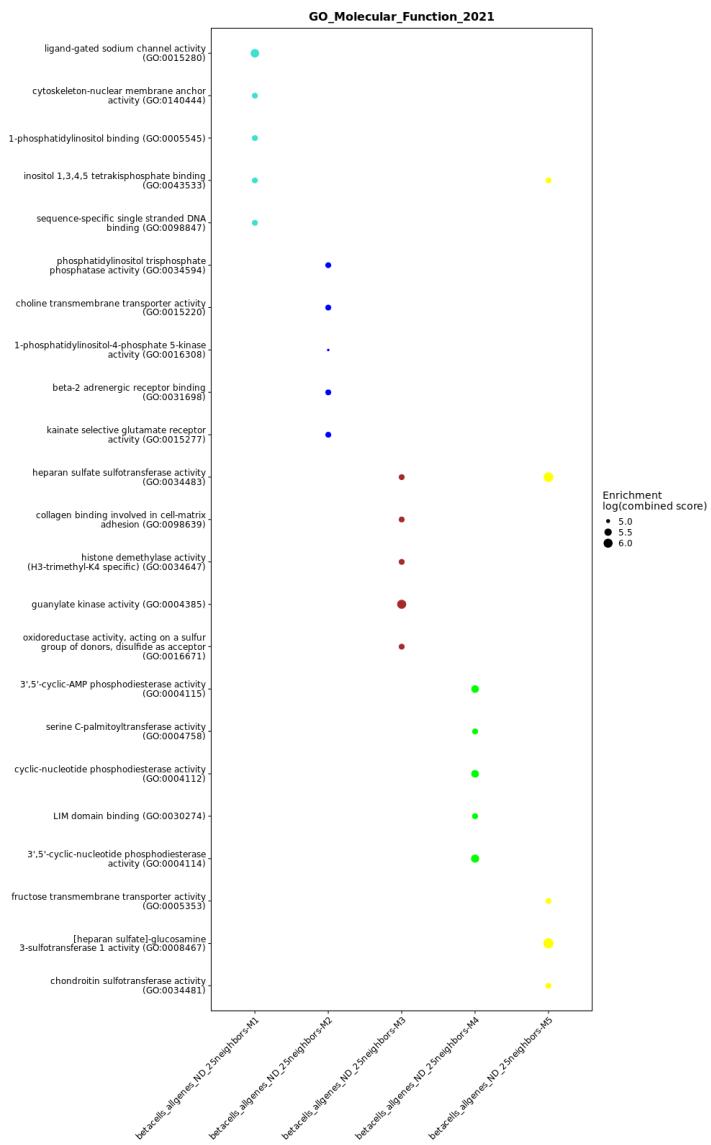
**Module-time point relationships in different cell types**

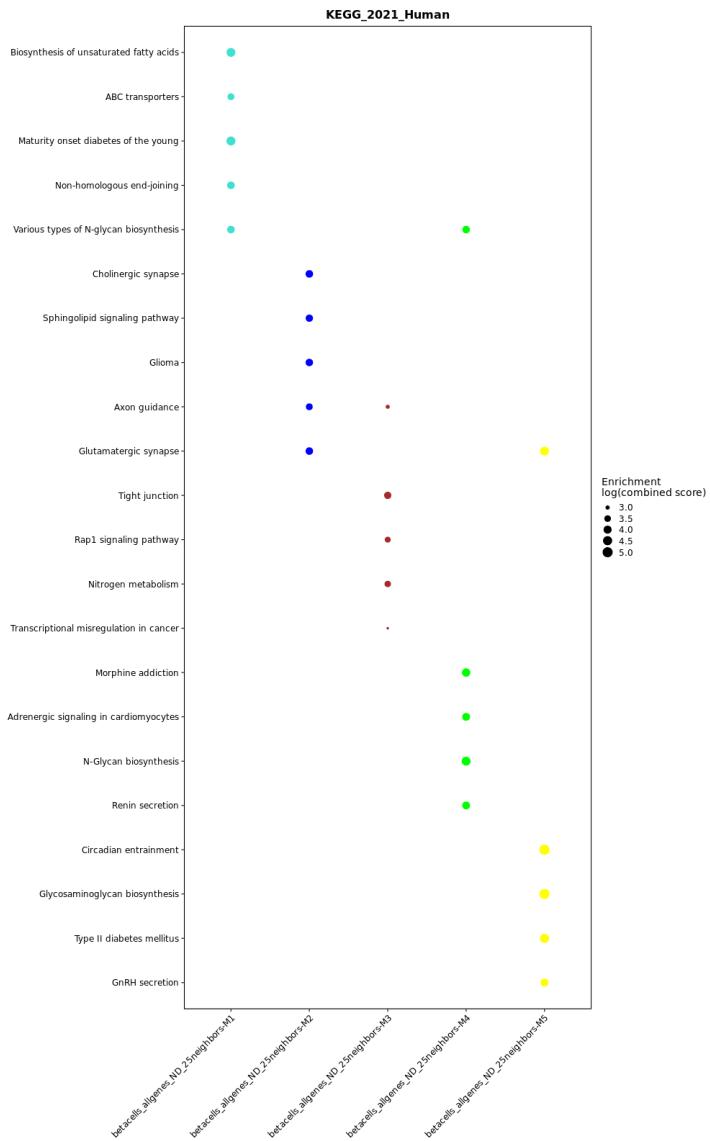


## ▼ Gene set enrichment

We can run gene set enrichment analysis with EnrichR for a given set of [available gene sets \(libraries\)](#) provided by the package. We can then get the top 5 for each module and plot them as a DotPlot (I also have per module barplots as supplementary but visualizing this way provides a better summary).



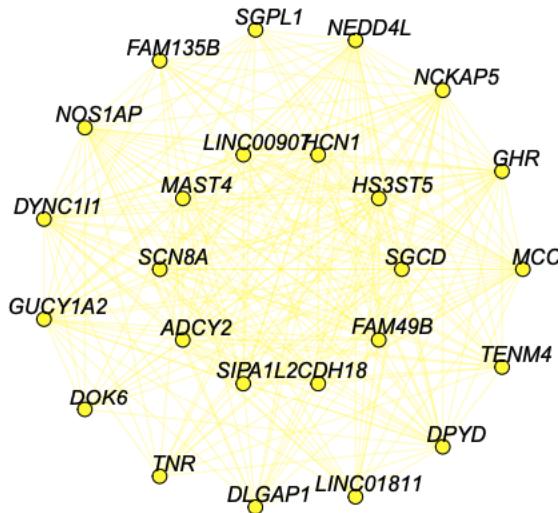




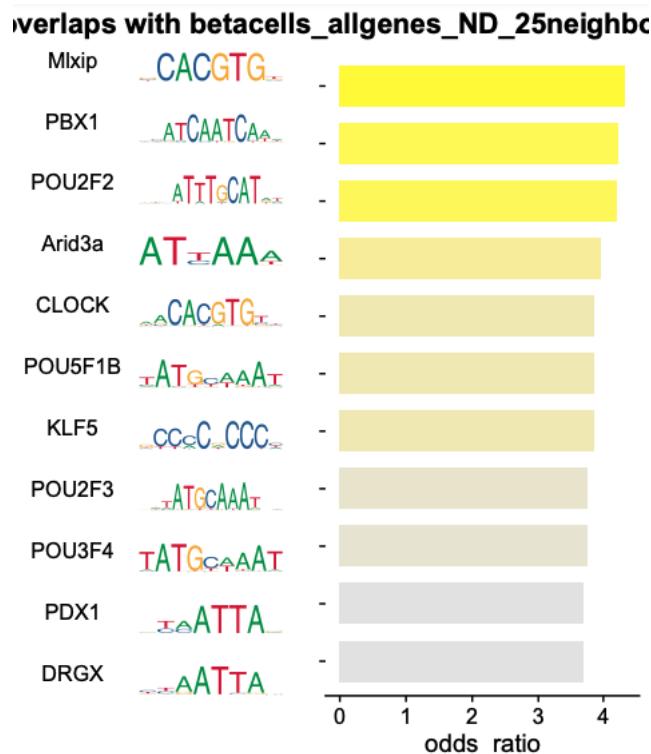
## ▼ Network analysis

We can visualize the “hub” genes of each module as a network with the top 10 on the inside and the top 15 on the outside. Edges represent the correlation strength for genes in this module. This really doesn’t do a whole lot except for provide a visual way to look at module membership\

### betacells\_allgenes\_ND\_25neighbors-M5



#### ▼ Motif analysis



#### ▼ `NormalizeData` SC- $\beta$ cell analysis across all genes with parameter exploration (`betacells_ND_exploration`)

Active assay: `RNA`

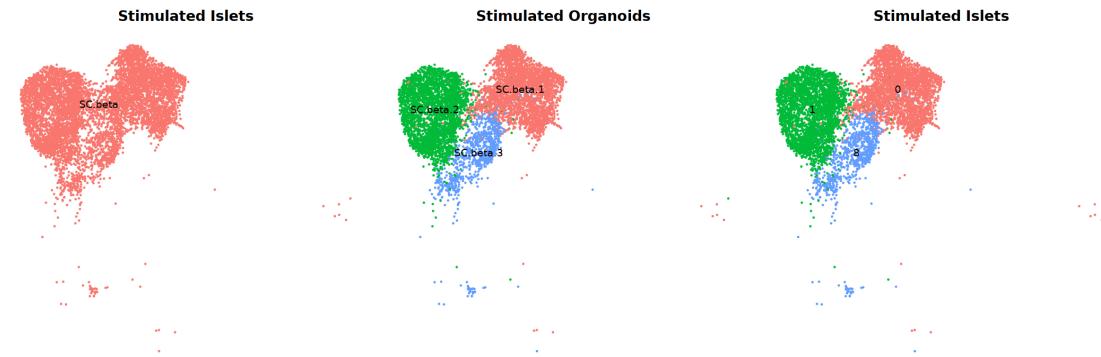
Cell types under consideration: `sc.β`

Fraction of cells to keep a gene: `0.05`

Number of neighbors for metacell construction: 25

## ▼ Network Construction

- Starting with 16342 cells across 36601 genes in the integrated multiome data
- Keeping only  $\alpha$ ,  $\beta$  and  $\delta$  cells from original processed Seurat .rds



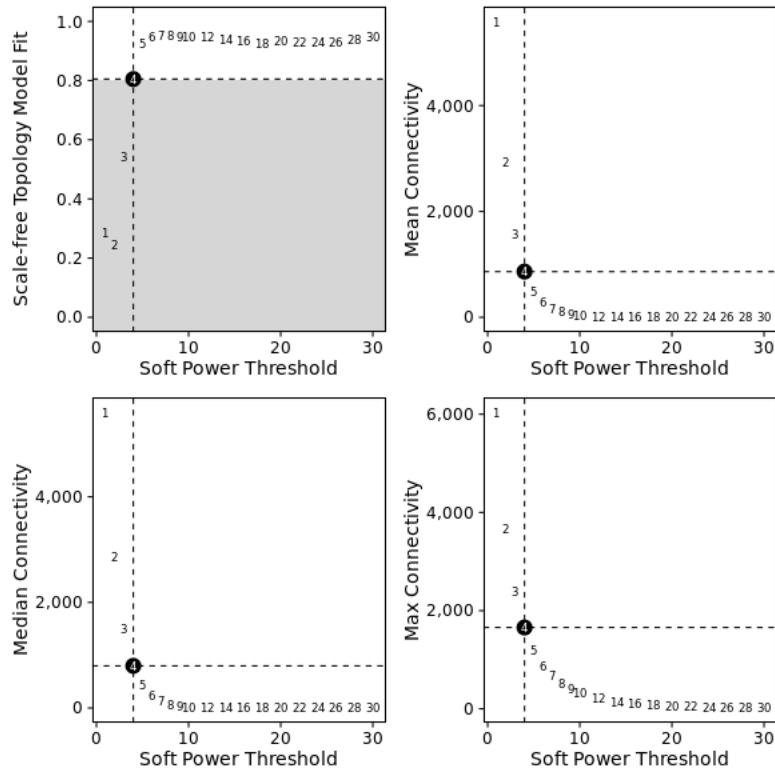
|         |
|---------|
| SC.beta |
| 7042    |

| SC.beta.1 | SC.beta.2 | SC.beta.3 |
|-----------|-----------|-----------|
| 3123      | 3080      | 838       |

- Using a fraction of 0.05 based on the counts slot of the active assay in `SetupForWGCNA` function to keep only genes expressed in at least 5% of these cells
  - 11003 genes remain after this filter
- Created metacells using top 25 nearest neighbors within a cell type (label from processed object). Used the counts to create these metacells in RNA assay and only allowed them to share 10 cells in common at most. Minimum cells by default is 100 and the max number of metacells to construct per cell type is 1000. Can normalize this metacell object using Seurat's default function that uses a `scale.factor` of 10000 and by default takes the natural log with a pseudo-count
  - Have a total of 2580 cells after this

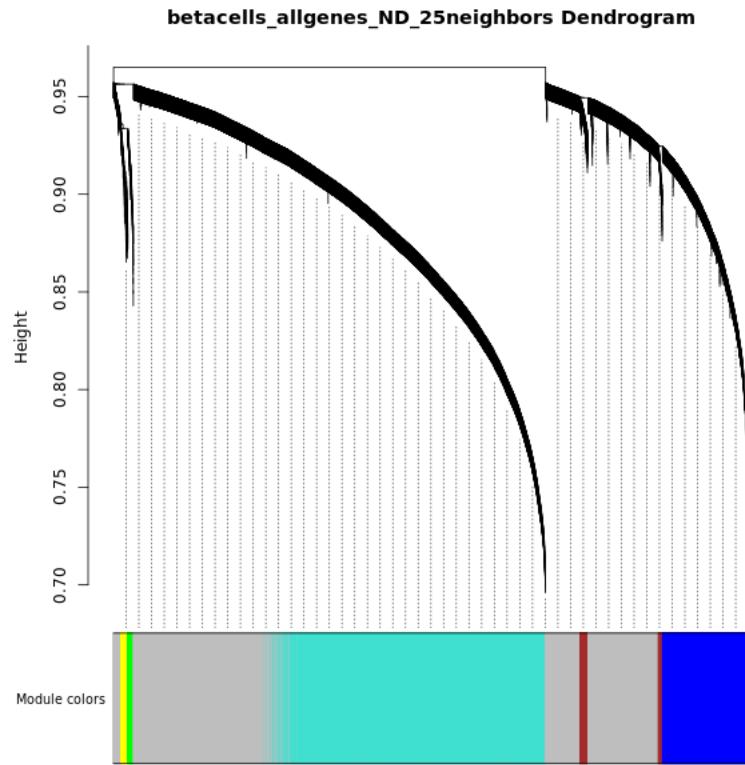
|         |
|---------|
| SC.beta |
| 2580    |

- Using the `SetDataExpr` function, selected the normalized metacell data from the data slot of the stored metacell Seurat object.
- Selected a soft-power threshold of 5 for downstream network construction



| A data.frame: 6 x 7 |          |           |                |           |           |           |
|---------------------|----------|-----------|----------------|-----------|-----------|-----------|
| Power               | SFT.R.sq | slope     | truncated.R.sq | mean.k.   | median.k. | max.k.    |
| <dbl>               | <dbl>    | <dbl>     | <dbl>          | <dbl>     | <dbl>     | <dbl>     |
| 1                   | 1        | 0.2881371 | 10.968859      | 0.9517153 | 5590.6868 | 5603.3172 |
| 2                   | 2        | 0.2452424 | -5.008839      | 0.8479542 | 2922.7231 | 2875.2103 |
| 3                   | 3        | 0.5450047 | -4.935994      | 0.7817115 | 1568.4551 | 1495.4946 |
| 4                   | 4        | 0.8050741 | -4.358672      | 0.9016088 | 863.3839  | 797.7157  |
| 5                   | 5        | 0.9268446 | -3.850557      | 0.9753876 | 487.5594  | 431.7712  |
| 6                   | 6        | 0.9467369 | -3.339103      | 0.9850075 | 282.5936  | 237.5245  |

- Using the normalized metacell expression matrix, performed WGCNA using soft power of 5. By default this builds a signed network using Pearson correlation.

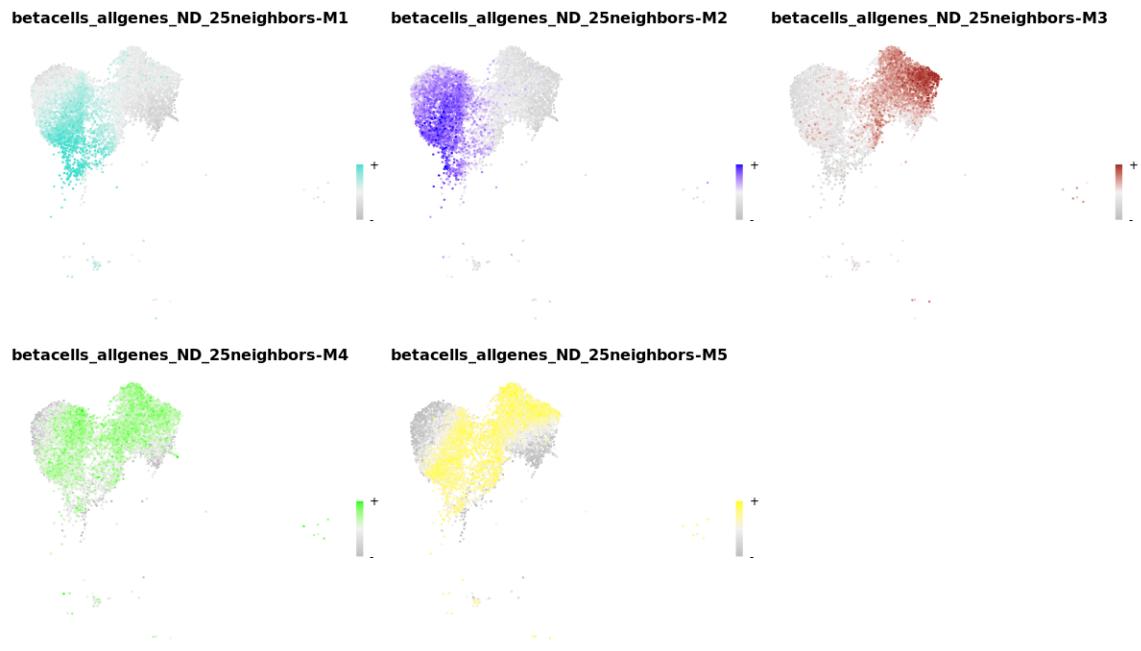


| black  | blue | brown | green | grey | magenta   | pink   | purple |
|--------|------|-------|-------|------|-----------|--------|--------|
| 74     | 501  | 381   | 101   | 4536 | 53        | 71     | 52     |
| Module | blue | brown | green | grey | turquoise | yellow |        |
| Count  | 1489 | 210   | 99    | 4453 | 4636      | 116    |        |

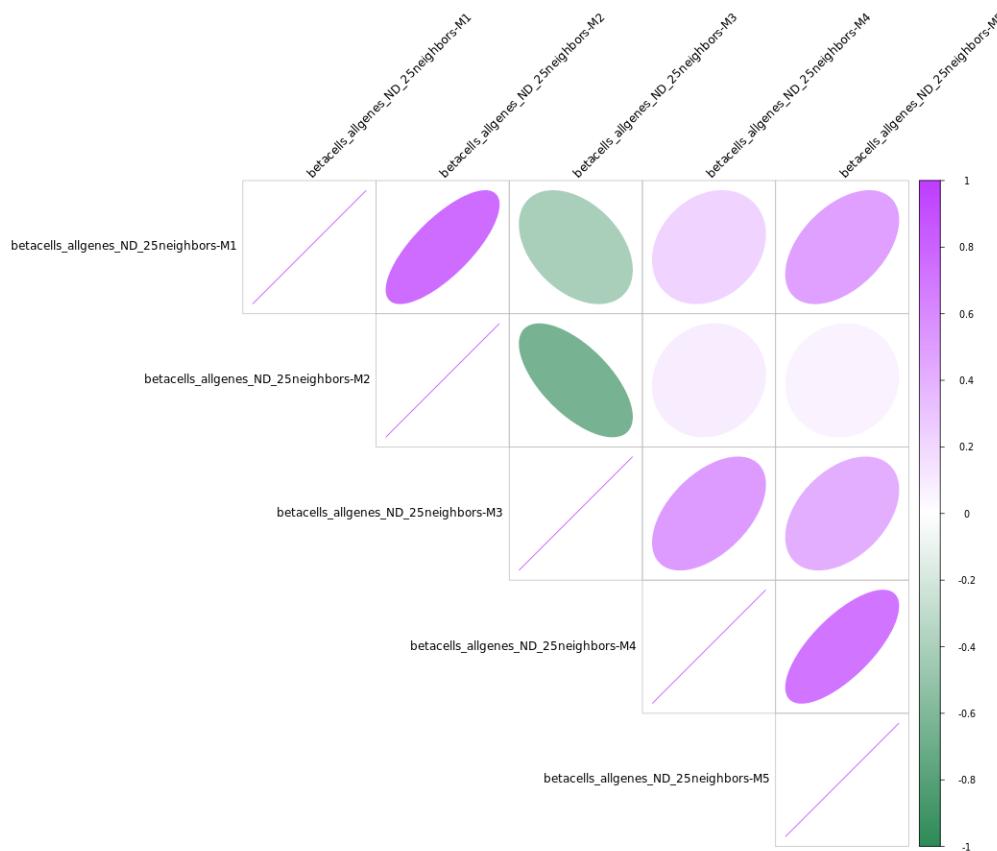
- Did not include a `group_by.vars` argument in the `ModuleEigengenes` function, which look like stops Harmony batch correction from running. The module eigengenes for each module are the 1st PC of the subsetted matrix
- Did not include `group_by` argument in the `ModuleConnectivity` function (which seems to make it take a long time —> ended up taking ~30 minutes). Computes a correlation for every gene in every module with its module eigengene.

## ▼ Basic Visualization

Plotting the raw MEs (PCA yields a sample x PC matrix for each module, I can overlay these onto the previous UMAP plot:



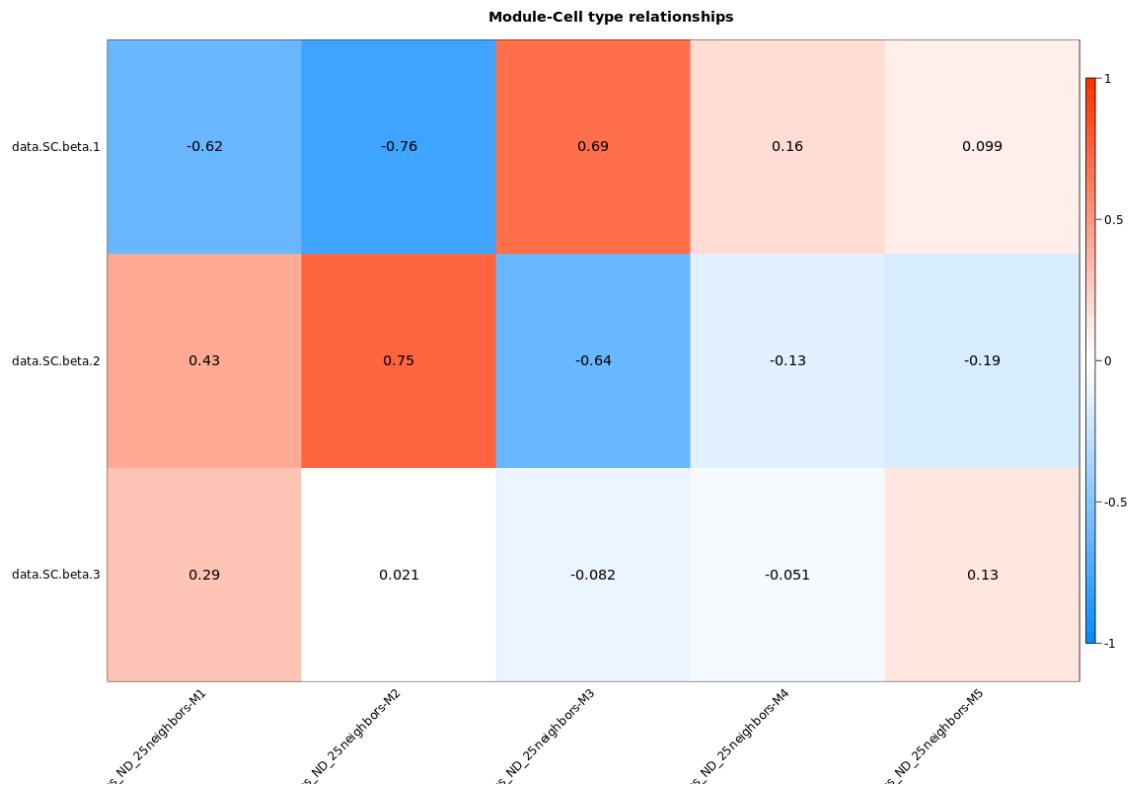
As a supplementary visualization, we can look at how correlated modules are with each other based on their MEs:

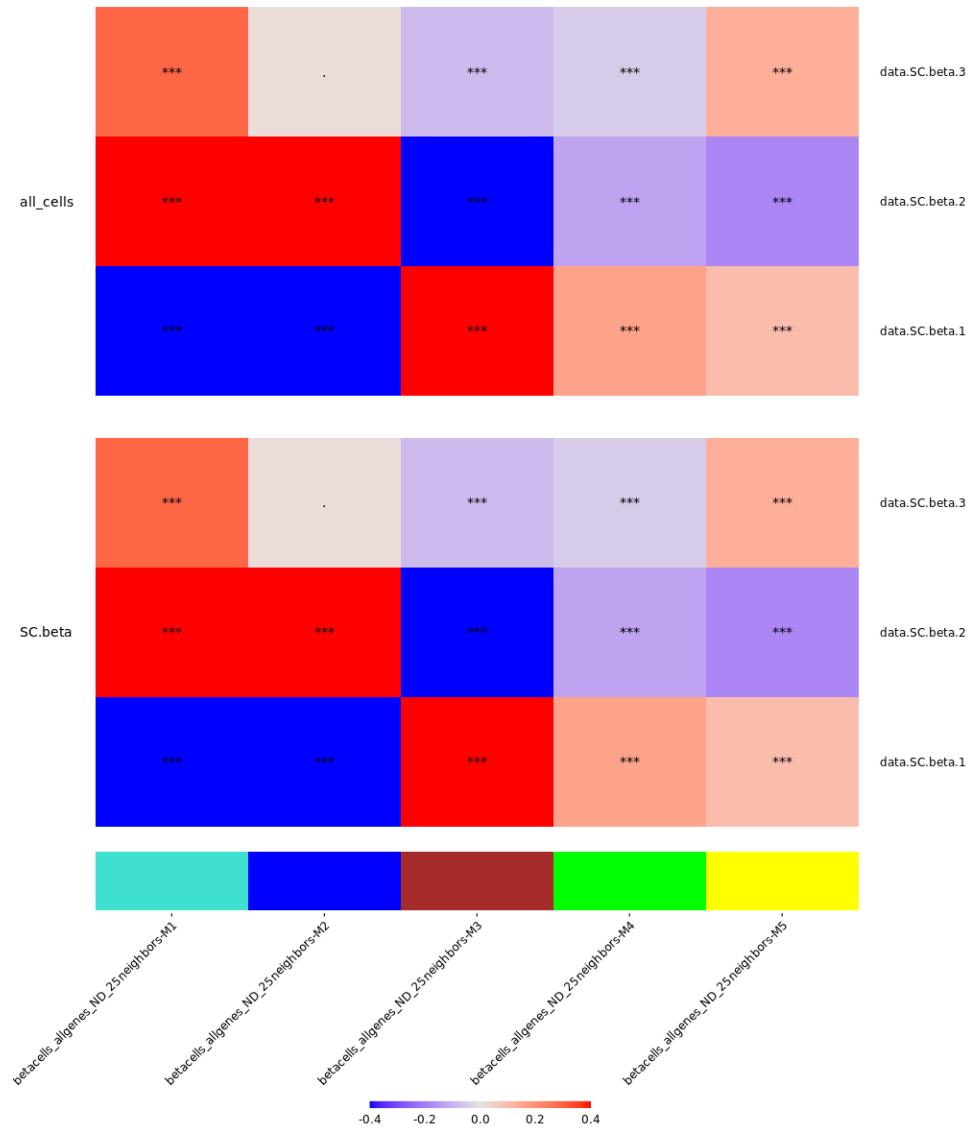


### ▼ Module trait correlation

I binarized each cell type in a 1 versus rest manner and calculated correlations with MEs across cells. For example, if the cell type is  $\alpha$  cells, each cell gets labeled a 1 if it is an  $\alpha$  cell and a 0 if it is some other cell type. I can then correlate this binary vector with ME vectors and get a module-trait correlation for  $\alpha$  cells across modules (last row of below matrix)

- Correlation of MEs with cell type

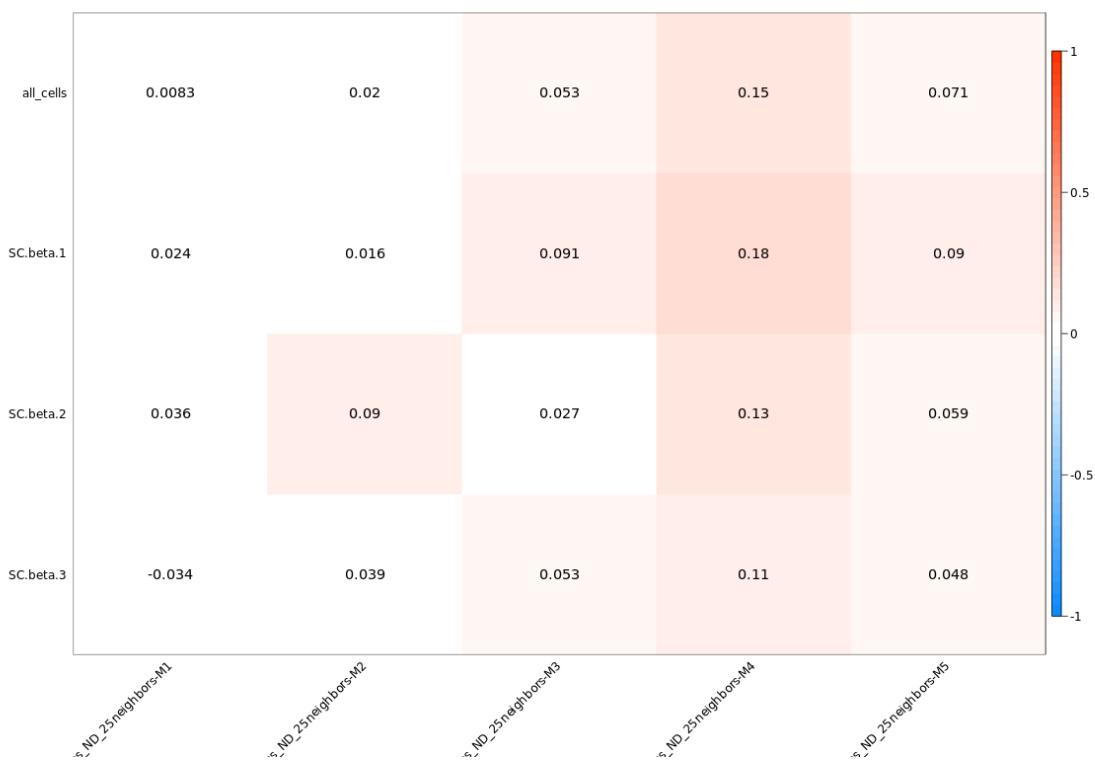




I can also look at correlations with treatments of the cells. To do this I binarized all “condition” metadata in a pairwise fashion, forcing untreated (`unt`) to come first and serve as 0. For example, if I am interested in the cytokine treatment (`cyt`), I label all cells that are treated with cytokines as 1 and label all cells that are untreated as 0. The rest are not considered and I calculate correlations on this subset of cells.

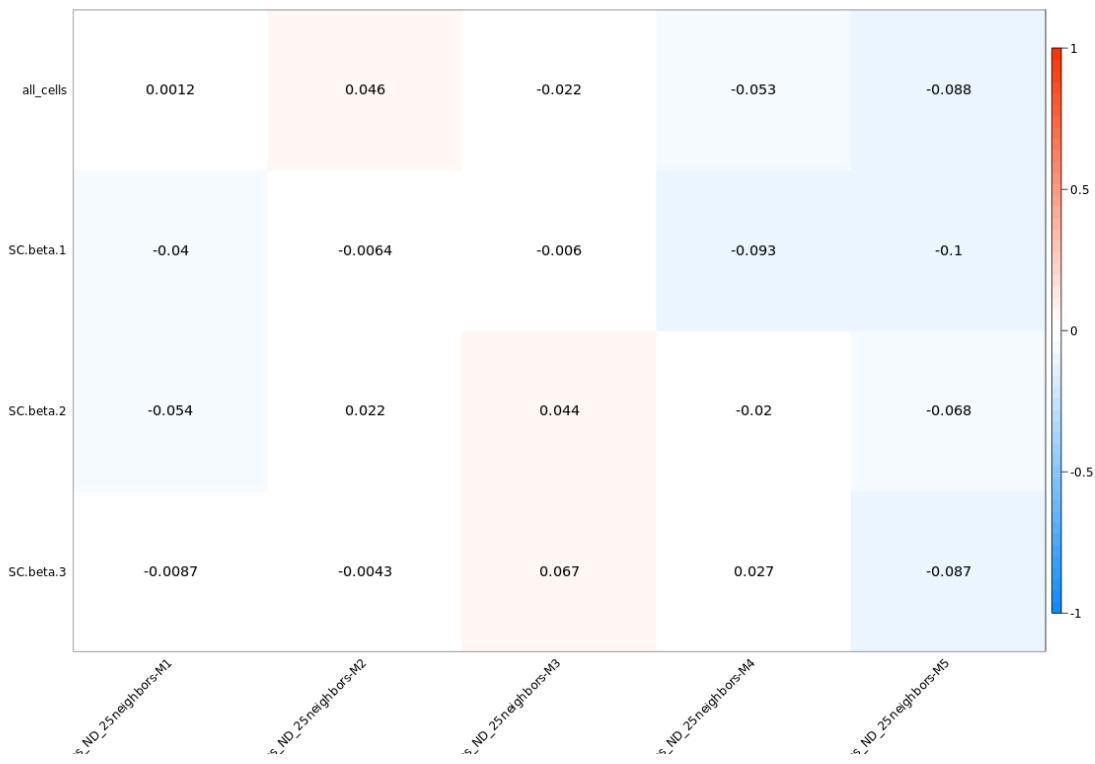
#### Correlation of MEs with treatment

**Module-palmitate treatment relationships in different cell types**



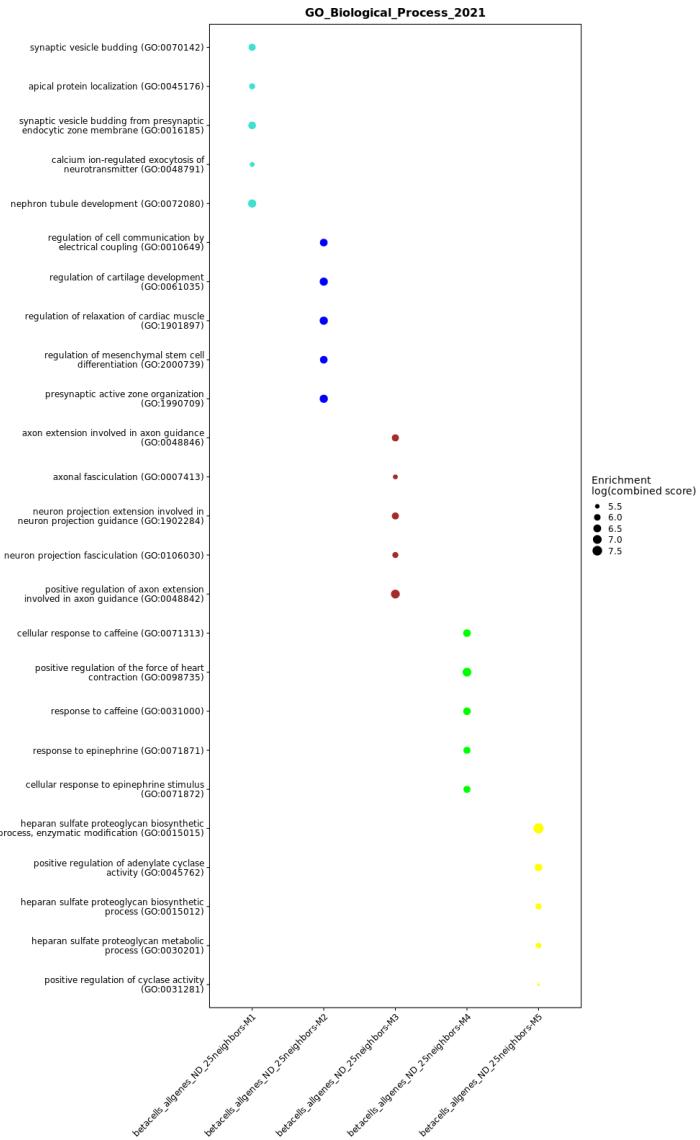
Correlation of MEs with timepoint

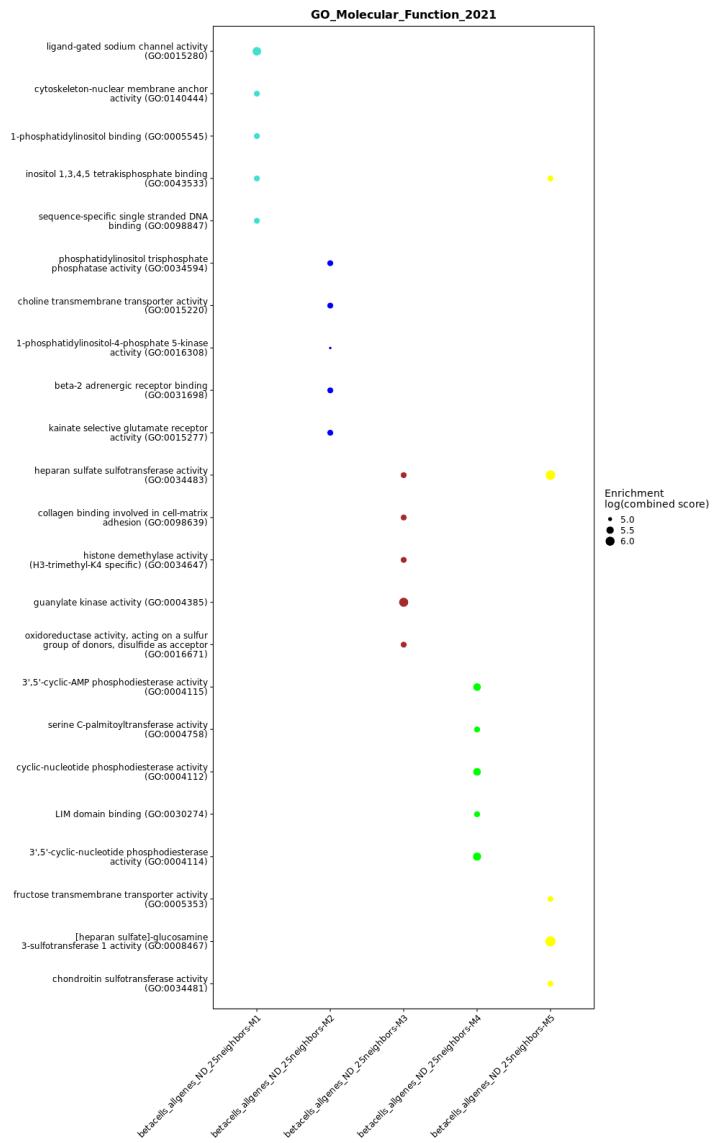
**Module-time point relationships in different cell types**

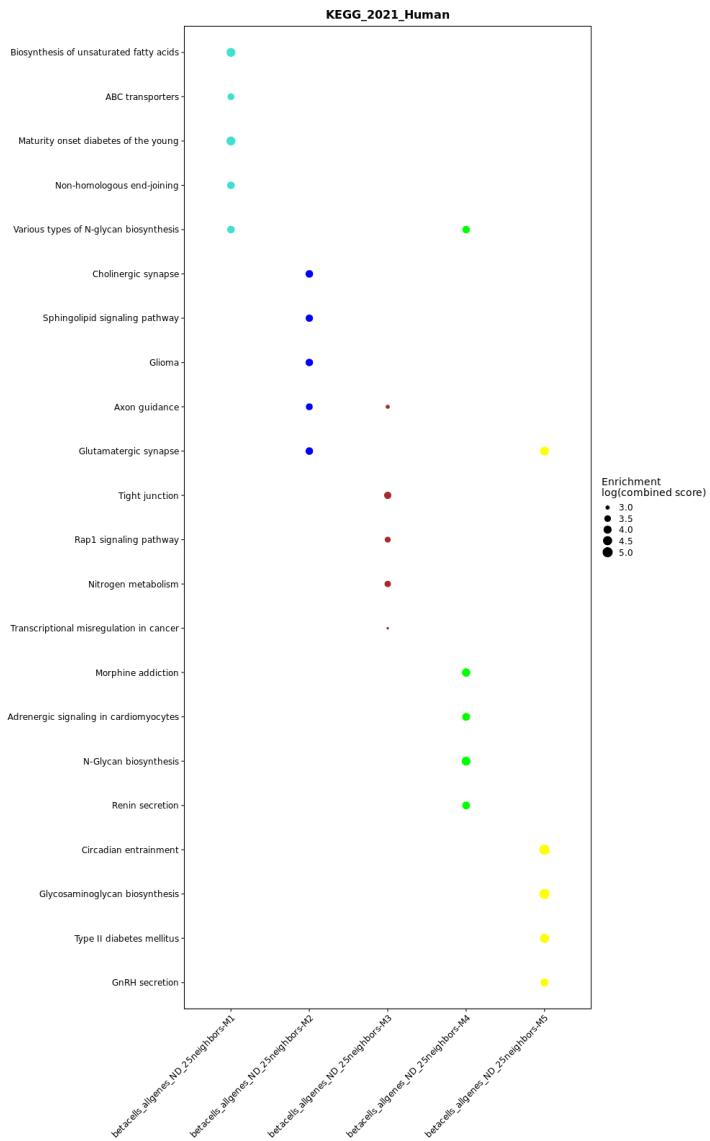


## ▼ Gene set enrichment

We can run gene set enrichment analysis with EnrichR for a given set of [available gene sets \(libraries\)](#) provided by the package. We can then get the top 5 for each module and plot them as a DotPlot (I also have per module barplots as supplementary but visualizing this way provides a better summary).



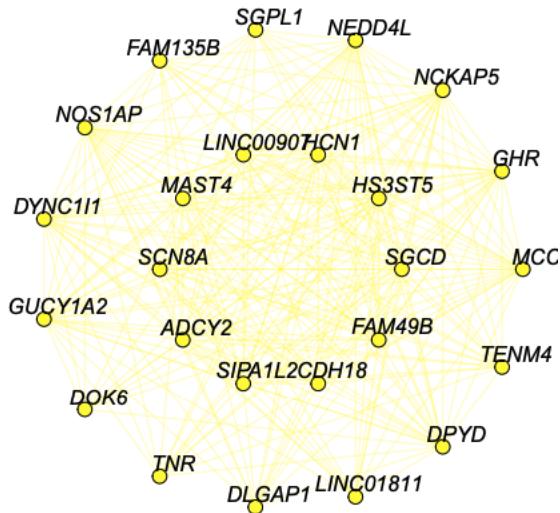




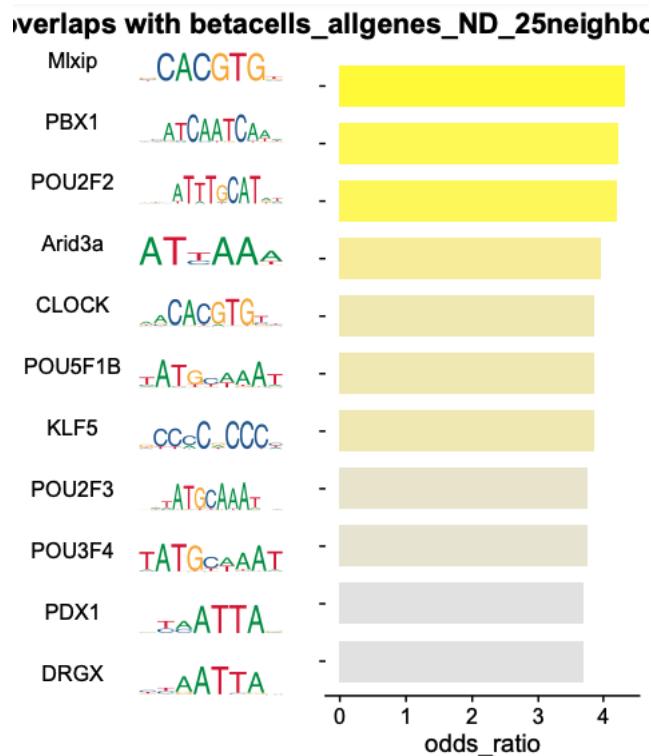
## ▼ Network analysis

We can visualize the “hub” genes of each module as a network with the top 10 on the inside and the top 15 on the outside. Edges represent the correlation strength for genes in this module. This really doesn’t do a whole lot except for provide a visual way to look at module membership\

### betacells\_allgenes\_ND\_25neighbors-M5



#### ▼ Motif analysis



## Help

- '\*\*\*': 0 - 0.001
- '\*\*': 0.001 - 0.01
- '\*': 0.01 - 0.05

- '+': 0.05 - 0.1
- (No symbol): 0.1 - 1.0

## Next steps

---

-