# Single cell RNAseq analysis in R

## 22-23 August 2022

### Archive of questions from the workshop Slack Channel

(In the order they were asked during the workshop)

| **Question/comment** | **Answer** |
|---|---|
| How I can save and export the count matrix? | This dataset would be created from a count matrix generated by cellranger.<br><br>We're currently working with the end result from a workflow. But the count matrix already exists (you can find it in `data/pbmc3k/filtered_gene_bc_matrices/hg19/` )<br><br>so typically, you wouldn't need to export the count matrix because it is a seperate file that exists before you create the Seurat object, if that makes sense? |
| Do we need to annotate the data before we can call the CD14 feature? | If you mean annotate in terms of assigning a cell type label or cluster - no<br><br>If you mean annotate in the sense of gene names - that information already exists in the count matrix<br><br>In this Seurat object the gene Id is the gene symbol<br><br>You'd also need to find the correct annotation for your gene of interest because Seurat can be a bit inflexible when it comes to changing feature information<br><br>I think the Seurat object will have the gene |

| | |
|---|---|
| | name if it's generated from a cellranger output. but if you know a gene by a different annotation, you have to find the annotation that the gene is going by in your Seurat object |
| What was the code to get the gene names? | `rownames(pbmc_processed)`<br><br>Returns a vector of all genes. It is case sensitive<br><br>I find this website quite useful to check that your gene id is correct: https://www.genecards.org/<br>For example, "c-myc" is my favourite gene but the HGNC id for this gene is "MYC" |
| From our breakout room, we had a question about finding gene names that match a specific pattern, here's an example solution: | `## Get a vector of all genes`<br>`gene_names <- rownames(pbmc_processed)`<br><br>`# Extract specific genes starting with the phrase CD - this code only works if you have the stringr package installed`<br>`cd_gene_names <- gene_names[stringr::str_starts(string = gene_names,`<br><br>`pattern = "CD")]`<br><br>`# A more general solution for pattern searching is using grepl and doesn't require any external packages. It does require knowledge of regular expressions`<br>`sel_gene_names <-`<br>`gene_names[grepl(pattern = "^CD", x =`<br>`gene_names, ignore.case =T)]` |
| What parameters should we be assessing when examining the html report from cellranger in terms of the sequencing/library prep quality? | The major thing I usually look for is consistency amongst samples from the same run.<br><br>This is also covered in the QC section. |
| Do we usually just set min.cell=3 and min.feature=200? | I think those are the default values of that function. I tend to set them both to 0 so no cells are filtered out just so I can look at the whole dataset before I start applying QC filters.<br><br>But up to you if you feel the thresholds are |

| | |
|---|---|
| | reasonable. it's strongly encouraged to explore your dataset thoroughly to be certain that any thresholds you are applying are sensible |
| Why do we want to remove mitochondrial genes? | Low-quality / dying cells often exhibit extensive mitochondrial contamination<br><br>A gene seen in only 3 cells is not going to tell us anything interesting.<br>A cell where we only saw 200 genes is also not going to be much use. It might even be an empty droplet that didn't contain a cell in the first place. |
| I have added the column name using pbmc$samplename <- 1. Once I have added a column name (i.e. name is 1) to a data set (dataset1) and added a column name (i.e. name is 2) to a second data set (dataset2), how do I combine the two data sets together? | Check this out<br>https://satijalab.org/seurat/articles/merge_vignette.html<br>That's assuming they're from one study, its more complex  if its from different studies. |
| What about doublets? Isn't that a good way to remove them from the data? | You generally are better of using a doublet detection tool rather than QC thresholds<br><br>Here is one of tools that works on Seurat object.<br>https://www.cell.com/cell-systems/pdf/S2405-4712(19)30073-0.pdf<br>https://github.com/chris-mcginnis-ucsf/DoubletFinder |
| For breakout room 4, re: object conversion - turns out I was wrong. So zellkonverter - for when your collaborator wants you to use scVelo and you need to venture out to the world of scanpy/python (i.e convert SingleCellExperiment object to AnnData). Specifically, how I utilise Bioconductor's scater package for adaptive QC thresholds - I load my data in as a seurat object. I also separately load in my data as a SingleCellExperiment object, identify outliers with scater, then grab the idents of | |

| | |
|---|---|
| those cells and add them into my Seurat object. There is probably a nicer/more sensible way of doing this (ie calculating the MAD thresholds directly on my Seurat object), but that's my inelegant solution | |
| Given the modern rseq reads are deeper, shall we change the default scale? | NormalizeData uses log1p() to log transform that data after scaling, so scale.factor has some influence about how far apart small counts are. I think scale.factor should be similar to the typical total count in each cell. Not sure how important this is in practice.<br>Also noting log1p uses log to base e. |
| One student in breakout room 7 asked about resources for spatial transcriptomics, which is quite different to scRNA-seq and outside of the scope of this workshop. However, for those that are interested this resource might be helpful: https://lmweber.org/OSTA-book/ | |
| What was the code you used to look at the counts i.e unnormal versus the normalised? | I think she just used a violin plot:<br>`VlnPlot(pbmc, "LYZ")`<br><br>but she ran the same line of code before and after normalising which is why it looked different<br><br>She used GetAssayData(pbmc, slot = 'counts') to get the unnormalised counts |
| Where does RStudio save data to? | It saves it to your working directory |
| How to force Seurat to ignore certain genes (e.g. cell cycle genes) when identifying variable genes for clustering? | If you run the cmd<br>`VariableFeatures(pbmc)`<br>this lists the 2000 genes chosen. So you could remove the cell cycle genes from that list<br><br>We can also regress them out,using `ScaleData()`, could provide cell cycle genes to this function<br>Seurat has a vignette on this too, it appears:<br>https://satijalab.org/seurat/articles/cell_cycle_vignette.html |

| | |
|---|---|
| | Seurat does also have a Cell cycle function too<br>`CellCycleScoring(marrow, s.features = s.genes, g2m.features = g2m.genes, set.ident = TRUE)` |
| If you're getting no separation using the nFeatures = 2000 (i.e. because the cells are super similar), can you reduce this? If so, are there any important considerations to make? | I haven't encountered this, but I think a few extra genes shouldn't matter too much - better too many (undirected noise) than too few (missing variation). But if you look at the variable genes plot and can see that the top 2000 (red) genes are digging into 'not-variable' territory then it could make sense to adjust that number. |
| When do we choose to use the linear dim reduction PCA and when should we choose to use the non-linear method such as umap ? | Generally with single cell analysis you'll go for a non-linear approach, because there is often too much heterogeneity for a linear PCA approach to work on its own. This data is PBMCs and actually doesn't look too bad with a PCA, but for more complex samples (e.g. whole tissue with specialised cell types and immune cells e.t.c) you can end up with an unhelpful blob.<br><br>If you were working with a more homogenous FACs sorted population you might have less diversity, and maybe a PCA would work fine. Its just a judgement on whatever looks like a sensible layout for your specific data. |
| In the workshop we have seen the quantitative analysis of transcript counts. Can we also have a look at the novel transcripts for a particular gene or overall? That will be more of a qualitative analysis. | It depends on the sequencing technology. Technologies like 10X only capture a small part of the gene at the 3' end to count it, so can usually only give gene-level count information (https://assets.ctfassets.net/an68im79xiti/4fly9tr6qQuCWamIii0iEa/40658acce7a6756e38537584897840e3/CG000108_AssayConfiguration_SC3v2.pdf) . To look at full transcripts you'd need to use a full length sequencing approach.<br><br>In any case, you can look at the .bam alignment files from your preprocessing (these are not on the vms) with a genome |

| | |
|---|---|
| | browser tool like IGV (https://software.broadinstitute.org/software/igv/download) and see how reads are aligned to whichever gene. |
| Cell cycle sorting | `# Cell cycle scoring (for human)`<br>`#`<br>`https://satijalab.org/seurat/articles/cell_cycle_vignette.html`<br><br>`# A list of cell cycle markers, from Tirosh et al, 2015, is loaded with Seurat. We can`<br>`# segregate this list into markers of G2/M phase and markers of S phase`<br>`s.genes <- cc.genes$s.genes`<br>`g2m.genes <- cc.genes$g2m.genes`<br><br>`pbmc <- CellCycleScoring(pbmc, s.features = s.genes, g2m.features = g2m.genes)` |
| Sneaky thing about Seurat - there is also a `cc.genes.updated.2019` object which I think is a more up to date list of cell cycle genes (there's 6 different genes between the two lists) | |
| It is not working on mine. even if I change the width of the screen. I get this error:<br><br>`When using repel, set xnudge and ynudge to 0 for optimal results` | Try `repel = FALSE` |
| We talked about how one can add or remove genes from Variable features, such as sex or y chromosome related genes. what is the code bit to do that? | It's a bit of a hack but this would be one option<br>`listOfGenesToRemove = c('TNF', 'MYC') # vector of genes to remove`<br>`RemoveIndex = which(VariableFeatures(pbmc) %in% listOfGenesToRemove) # get position in Variable genes`<br>`VariableFeatures(pbmc) = VariableFeatures(pbmc)[-(RemoveIndex)]`<br>`#remove them and add back into VariableFeatures` |
| Where are the PCA data stored in pbmc? | pbmc@reductions |
| So how do we conceptualise the PC loading, and its positive/negative value? | In a PC:<br>genes have a loading<br>cells have an score (aka embedding) |

| | |
|---|---|
| | If a gene has a positive loading, it will tend to be up in cells with a positive score and down in cells with a negative score, vice versa for negative. |
| Can a gene have equal loadings on both PC1 and PC2? | Yes, its possible. Some genes could be highly discriminative and some may have similar loadings. |
| Can anyone figure out why I am getting this error? CST# worked but MALAT no?<br><br>`FeaturePlot(pbmc,'MALAT')`<br>`Error: None of the requested features were found: MALAT in slot data`<br>`In addition: Warning message:`<br>`In FetchData.Seurat(object = object, vars = c(dims, "ident", features), :`<br>`The following requested variables were not found: MALAT`<br>`> FeaturePlot(pbmc, "MALAT")`<br>`Error: None of the requested features were found: MALAT in slot data`<br>`In addition: Warning message:`<br>`In FetchData.Seurat(object = object, vars = c(dims, "ident", features), :`<br>`The following requested variables were not found: MALAT` | The gene name is MALAT1 |
| Would you ever plot other PCs or always just use PC1 and PC2 during this step of analysis? | You could but this is what the UMAP is doing.<br><br>It also gives poor resolution |
| How would you write it in the FeaturePlot() function to look at other PCs? | FeaturePlot has an argument `dims = c(1, 2)`<br><br>you'd change that argument<br><br>Example:<br>`FeaturePlot(pbmc, "CST3", dims=c(3,4), reduction="pca")`<br><br>you can always use ?function_name to check the documentation e.g `?FeaturePlot` |

| | |
|---|---|
| Where do we find a scree plot to see the total variance explained by our PCs? Instead of SD? | I guess<br>`plot(pbmc$pca@stdev^2)` |
| Does the Seurat seed for the UMAP change each time? Or always the same each time for same data? | I think it's normally the same seed for your session.<br>If you re-run, you'll get the same umap<br>In runPCA seed.use<br>Set a random seed. By default, sets the seed to 42. Setting NULL will not set a seed.<br><br>Generally your session persists from day to day until your RStudio crashes |
| Can you discard all the mitochondrial and ribosomal genes at the beginning if they are not important in your study? | You'd need to remove them from the matrix itself before you generate the seurat object if i recall correctly<br><br>You generally can't discard features from a seurat object - or so it has been for most versions - i don't know if they've changed it with v4 (edited)<br><br>you can use the 'features =' option in the FindMarkers function |
| `pred.cnts <- SingleR::SingleR(test = sce, ref = ref.set, labels = ref.set$label.main)`<br>`Error in rownames(x) : object 'sce' not found`<br><br>??? | Is your converted Seurat object called sce /did you run the line of code to create sce ? |
| Do you have recommendations for celltype packages other than celldex? I am interested in human bone marrow populations | I think, you can get various datasets from scRNAseq: https://bioconductor.org/packages/release/data/experiment/html/scRNAseq.html |
| So if we have multiple data on the same sorted cell type and want to do pseudobulk should that happen after harmony integration? | I think so. Users of harmony to confirm?<br><br>Yes or some other integration method |