

VelocityVisualizationTutorial

LylaAtta

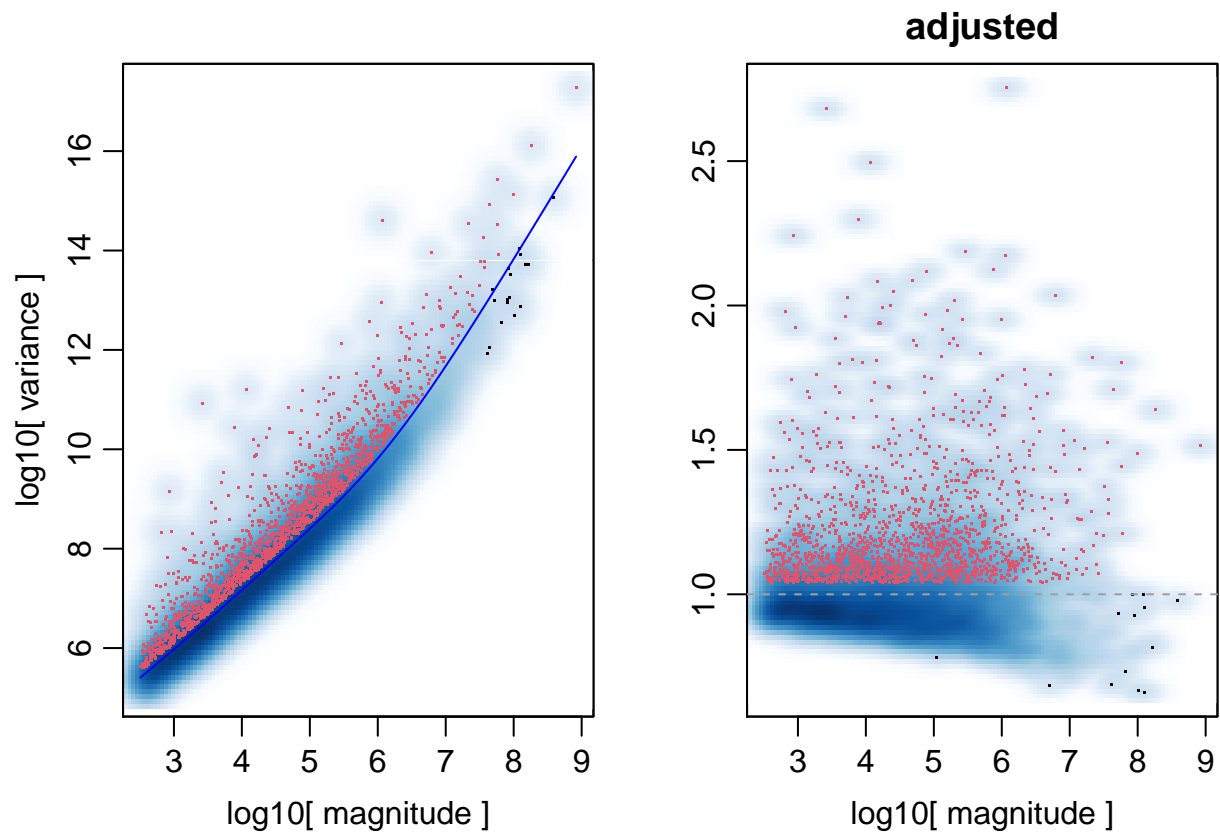
8/10/2020

R Markdown

RNA velocity analysis of MERFISH U2OS cells (Xia, Fan, Emmanuel, et al. 2019)

Load data and cleanup

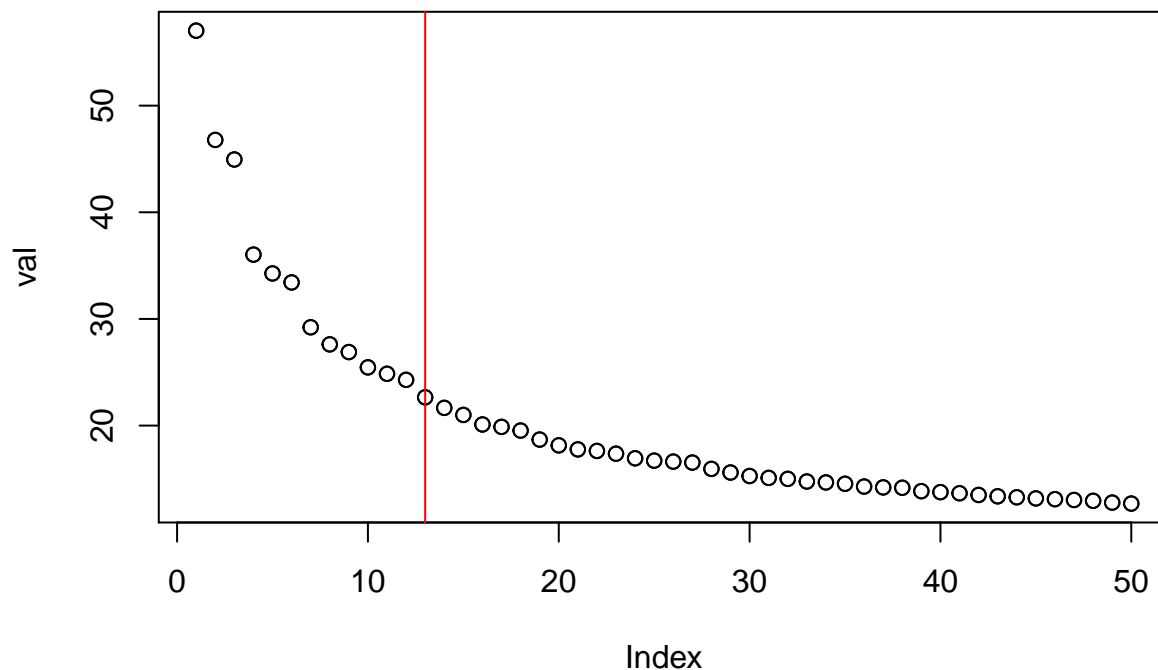
```
## Found3batches
## Adjusting for0covariate(s) or covariate level(s)
## Standardizing Data across genes
## Fitting L/S model and finding priors
## Finding parametric adjustments
## Adjusting the Data
## Warning in if (!class(counts) %in% c("dgCMatrix", "dgTMatrix")) {: the condition
## has length > 1 and only the first element will be used
## Converting to sparse matrix ...
## Normalizing matrix with 1368 cells and 6109 genes
## [1] "Calculating variance fit ..."
## [1] "Using gam with k=5..."
## [1] "1595 overdispersed genes ... "
```



Dimensional reduction, community detection, clustering

```
pca = svds(A = mat, k = 50, opts = list(center = TRUE, scale = FALSE, maxitr = 2000, tol = 1e-10))
val = pca$d
plot(val)
points(val)

N = 30
abline(v=13, col = 'red')
```



```
pcs = pca$u[,1:N]
rownames(pcs) = colnames(mat.bc)
colnames(pcs) = paste0('PC',1:N)
```

```
#low dimension embedding
```

```
emb.test = pcs[,1:2]
```

```
#community detection
```

```
com = getComMembership(pcs, k=300, method = igraph::cluster_louvain)
```

```
## [1] "finding approximate nearest neighbors ..."
```

```
## [1] "calculating clustering ..."
```

```
## [1] "WARNING"
```

```
## [1] "graph modularity: 0.121100754682579"
```

```
## [1] "identifying cluster membership ..."
```

```
## com
```

```
## 1 2 3 4
```

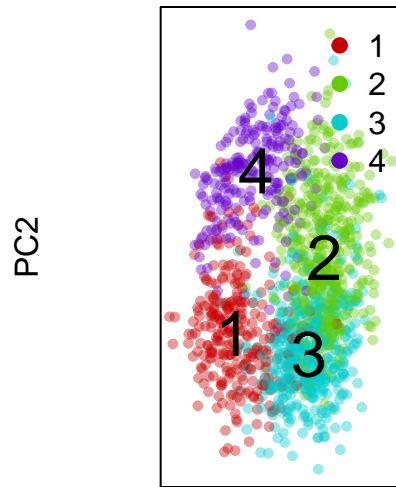
```
## 267 436 436 229
```

```
par(mfrow = c(1,2),mar=rep(5,4))
```

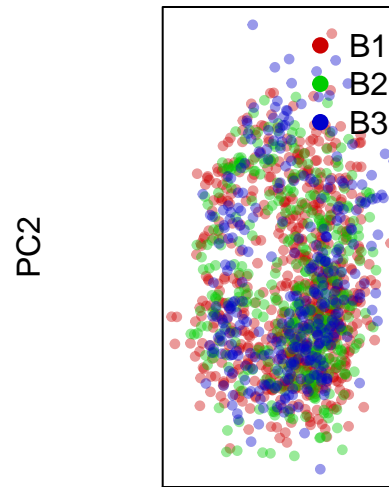
```
plotEmbedding(emb.test, groups = com, mark.clusters = TRUE, show.legend = TRUE, xlab = 'PC1', ylab = 'PC2')
```

```
plotEmbedding(emb.test, groups=batch, show.legend = TRUE,
              xlab = 'PC1', ylab = 'PC2', main = 'batch', verbose = FALSE)
```

clustering



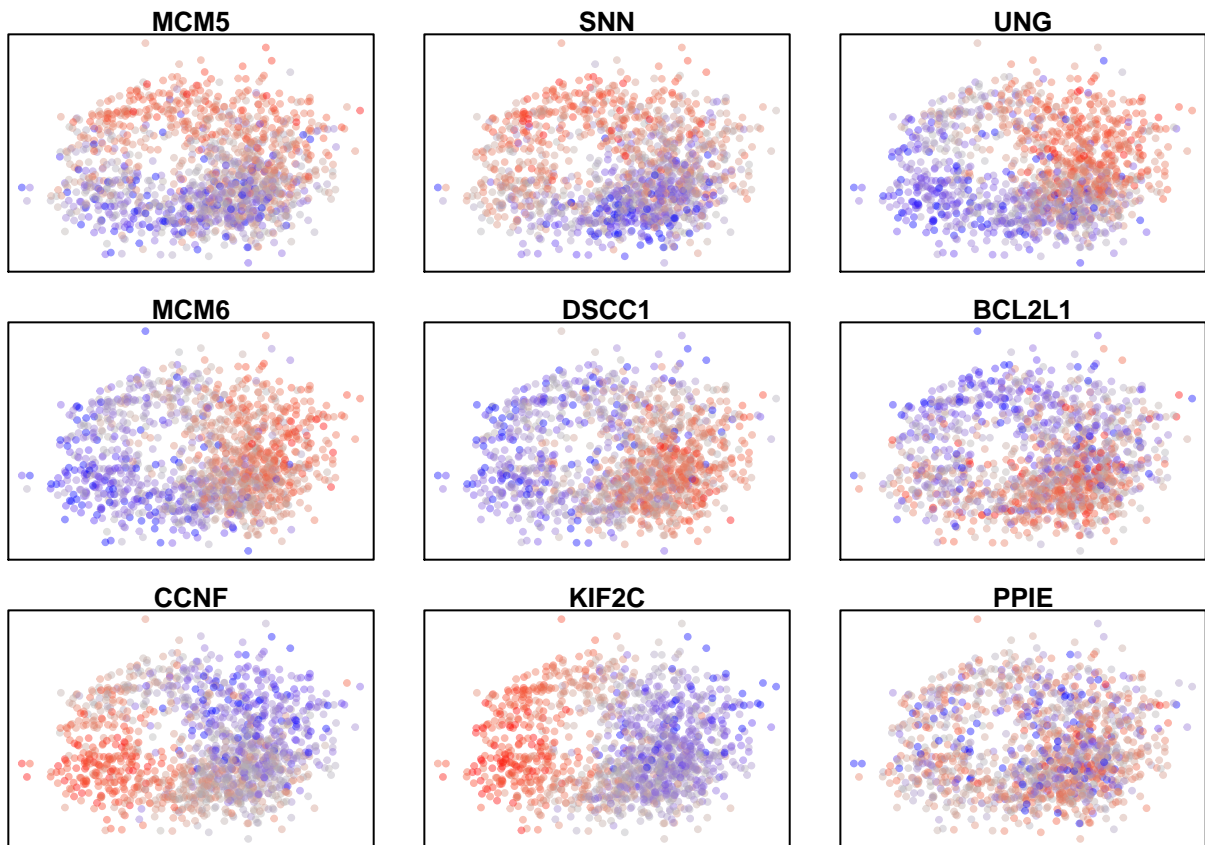
batch



PC1

PC1

```
## Plot cell-cycle and house keeping genes noted in the manuscript
par(mfrow=c(3,3), mar=rep(1,4))
gs <- c('MCM5', 'SNN', 'UNG', 'MCM6', 'DSCC1', 'BCL2L1', 'CCNF', 'KIF2C', 'PPIE')
invisible(lapply(gs, function(g) {
  gexp = scale(log10(mat.cpm[g,]+1))[,1]
  gexp[gexp > 2] <- 2
  gexp[gexp < -2] <- -2
  plotEmbedding(emb.test, main=g, col=gexp, verbose=FALSE)
}))
```



```
cluster.label = factor(com)
cell.color = MUDAN:::fac2col(cluster.label)
```