

# Varying-Censoring Aware Matrix Factorization

Will Townes

Collaborators: Stephanie Hicks, Martin Aryee, Rafael Irizarry

Harvard Biostatistics

26 July 2017

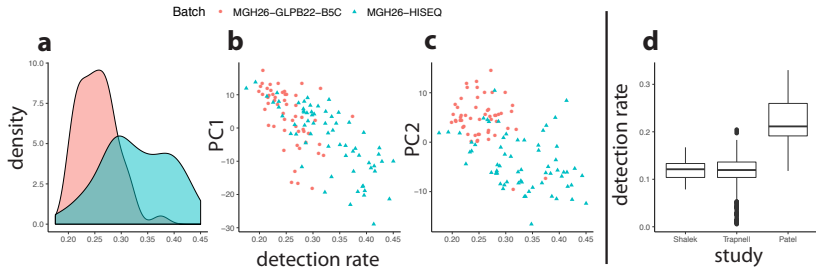
# Single Cell RNA-Seq

- ▶ Goal: cluster cells
- ▶ 40-90% zeros (vs  $\approx 20\%$  for bulk)
- ▶ **Detection Rate**: proportion of genes nonzero for each cell
- ▶ Some zeros may really be censored values

# Single cell RNA-Seq Statistical Modeling

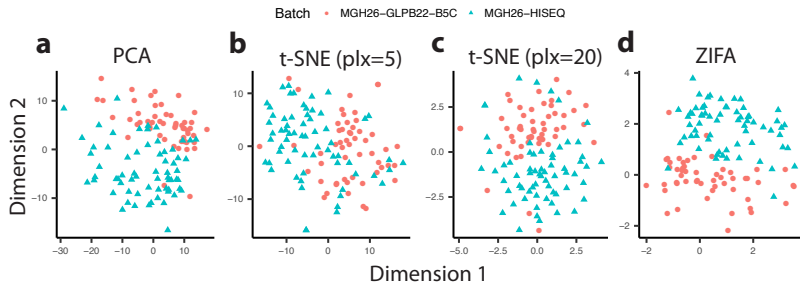
- ▶ Clustering requires pairwise distances
- ▶ Curse of dimensionality
- ▶ Popular dimension reduction: PCA and/or t-SNE
- ▶ Neither accounts for censoring
- ▶ ZIFA (Pierson & Yau)- single global parameter

# Batch Effects via Detection Rates

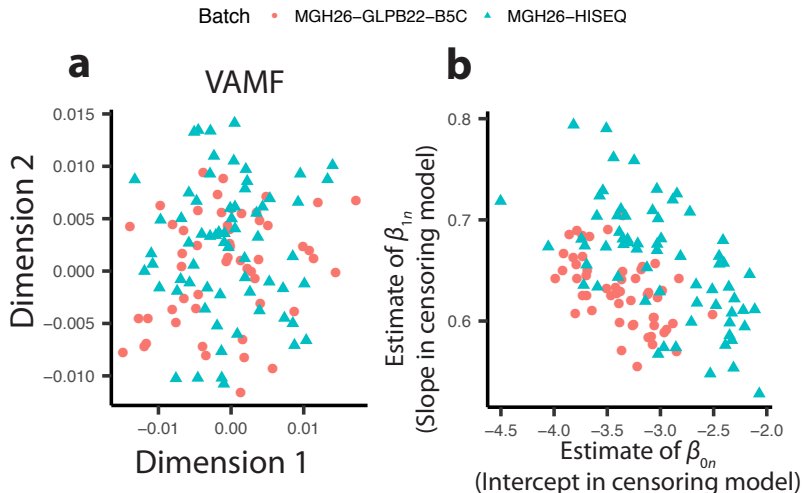


Patel et al 2014 glioblastoma tumor in two batches with varying detection rates.

# Distorted Factors on Glioblastoma Data



# VAMF Avoids False Clusters



## R package

```
devtools::install_github("willtownes/vamf")  
library(vamf)  
help(vamf)
```

```
vamf {vamf}
```

# Varying-Censoring Aware Matrix Factorization.

## Description

VAMF is a probabilistic dimension reduction method intended for single cell RNA-Seq datasets.

## Usage

```
vamf(Y, L, nrestarts = 4, log2trans = TRUE, pseudocount = 0,  
      output_samples = 100, save_restarts = FALSE, svmult = 1)
```

## Arguments

- Y** Sparse Matrix of gene expression measurements, with genetic features (gr
- L** Upper bound on the dimensionality of the latent space to be learned. Autor dimensions.

## R package

```
options(mc.cores=4)
vamf_result<-vamf(Y,10)
pd<-vamf_result$factors
ggplot(pd,aes(x=dim1,y=dim2))+geom_point()
```



# Varying-Censoring Aware Matrix Factorization

- ▶ Poster presentation tomorrow.
- ▶ R package under development on GitHub
- ▶ Townes, Hicks, Aryee, Irizarry bioRxiv preprint
- ▶ Built on top of Stan

# Simulation: Finding True Clusters

- ▶ 4 true clusters
- ▶ 160 cells, 1000 genes, 50 informative genes
- ▶ 80 cells medium censoring ( $P_n \approx .5$ )
- ▶ 80 cells high censoring ( $P_n \approx .85$ )
- ▶ censor through logit model.

# Simulation

