Statistics Introduction

Christof Angermueller

November 27, 2014

1. Clustering

2. PCA

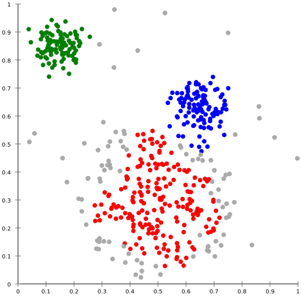
1 Clustering

2 PCA

Clustering

- **Goal**: Finding groups of related items

- How do define relatedness?
- Which clustering methods exist?
- What is Hierarchical clustering?
- How to visualize clustering?



How to define relatedness?

Distance

- Euclidean distance
- Manhattan distance
- Binary distance

Similarity

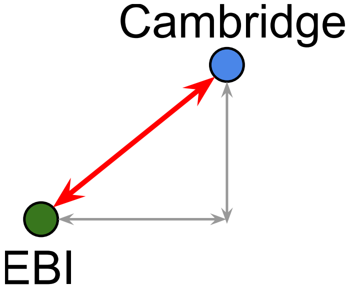
- Identity
- Correlation
- → Every similarity can be converted into distance

Euclidean distance

Definition

Euclidean distance

$$d(x,y) = \sqrt{\sum_i (x_i - y_i)^2}$$

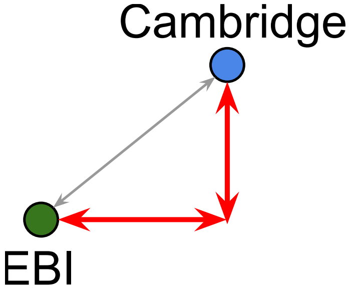


Manhattan distance

Definition

Manhattan distance

$$d(x, y) = \sum_i |x_i - y_i|$$



Methods

Partitioning clustering

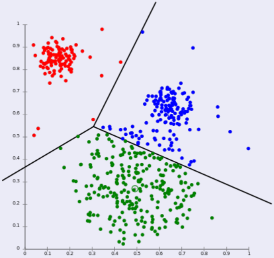
Density clustering

Hierarchical clustering

Methods

Partitioning clustering

- Partition points into k clusters
- k known a-priori
- k-means
- k-medoids



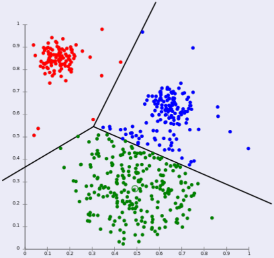
Density clustering

Hierarchical clustering

Methods

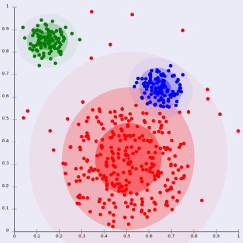
Partitioning clustering

- Partition points into k clusters
- k known a-priori
- k-means
- k-medoids



Density clustering

- Cluster points into dense regions
- k unknown a-priori
- DBSCAN
- OPTICS

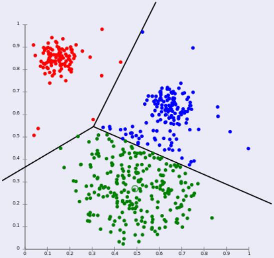


Hierarchical clustering

Methods

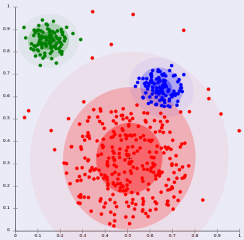
Partitioning clustering

- Partition points into k clusters
- k known a-priori
- k-means
- k-medoids



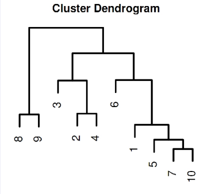
Density clustering

- Cluster points into dense regions
- k unknown a-priori
- DBSCAN
- OPTICS



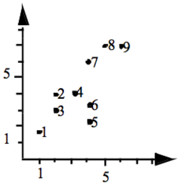
Hierarchical clustering

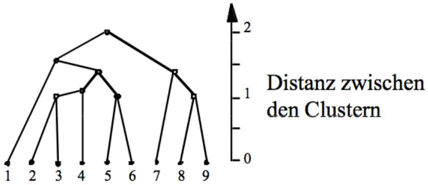
- Find hierarchy of clusters
- k unknown a-priori
- Single-linkage
- Complete-linkage
- Average-linkage



Hierarchical clustering

- Constructs hierarchy of clusters represented by a Cluster dendrogram
- Cluster dendrogram
  - Leaf nodes: single data points
  - Inner nodes: cluster of points
  - Root node: all data points

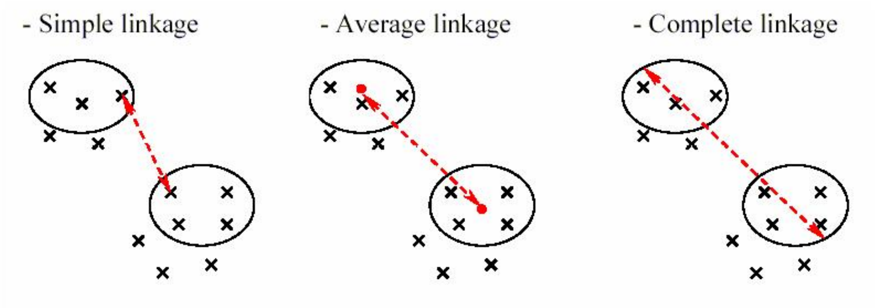




Algorithm

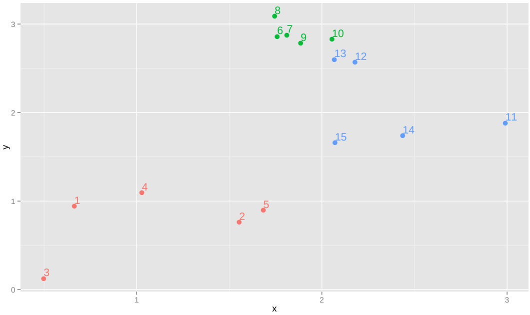
1. Start with clusters that contain only one point
2. Compute distance between clusters
3. Merge the two clusters with lowest distance into new cluster
4. Go to step 2 until one single cluster remains

How to compute the distance between clusters?



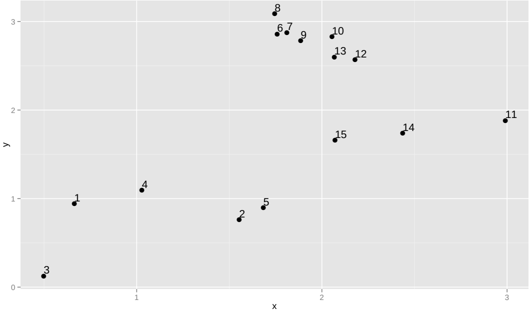
Data

```
set.seed(5)
n = 5
nclust = 3
x_means = c(1, 2, 2.5)
y_means = c(1, 2.5, 2)
x = rnorm(n * nclust, rep(x_means, each=n), sd=.4)
y = rnorm(n * nclust, rep(y_means, each=n), sd=.4)
clust = factor(rep(1:nclust, each=n))
id = 1:(n * nclust)
cdata = data.frame(id=id, x=x, y=y, clust=clust)
ggplot(cdata, aes(x=x, y=y)) + geom_point(aes(color=clust), size=3, show_guide=F) +
  geom_text(aes(label=id, color=clust), show_guide=F, vjust=-.3, hjust=0)
```



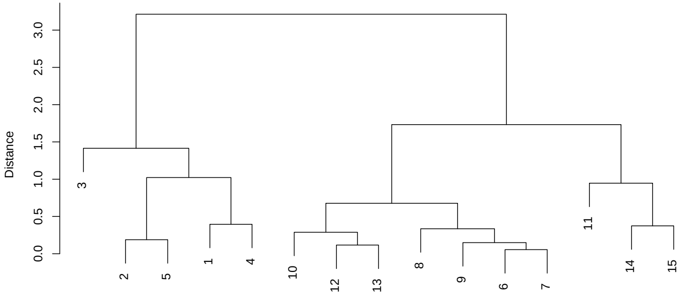
Data

```
set.seed(5)
n = 5
nclust = 3
x_means = c(1, 2, 2.5)
y_means = c(1, 2.5, 2)
x = rnorm(n * nclust, rep(x_means, each=n), sd=.4)
y = rnorm(n * nclust, rep(y_means, each=n), sd=.4)
clust = factor(rep(1:nclust, each=n))
id = 1:(n * nclust)
cdata = data.frame(id=id, x=x, y=y, clust=clust)
ggplot(cdata, aes(x=x, y=y)) + geom_point(color='black', size=3, show_guide=F) +
  geom_text(aes(label=id), color='black', show_guide=F, vjust=-.3, hjust=0)
```



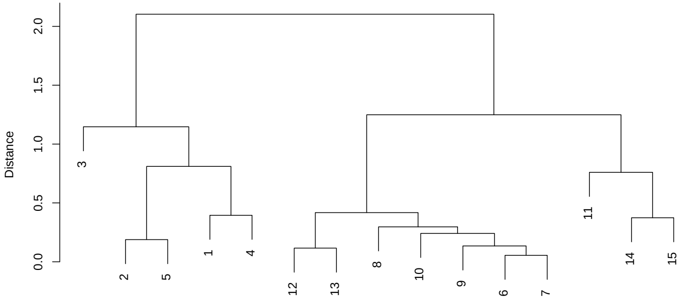
Complete linkage

```
cdist = dist(cbind(cdata$x, cdata$y), method='euclidean')
chclust = hclust(cdist, method='complete')
plot(chclust, ylab='Distance', xlab=NA, main=NA, sub=NA)
```



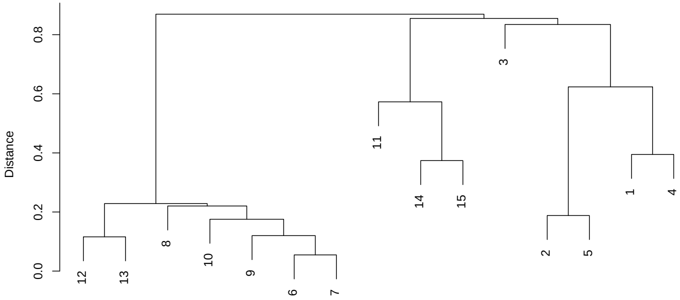
Average linkage

```
cdist = dist(cbind(cdata$x, cdata$y), method='euclidean')
chclust = hclust(cdist, method='average')
plot(chclust, ylab='Distance', xlab=NA, main=NA, sub=NA)
```



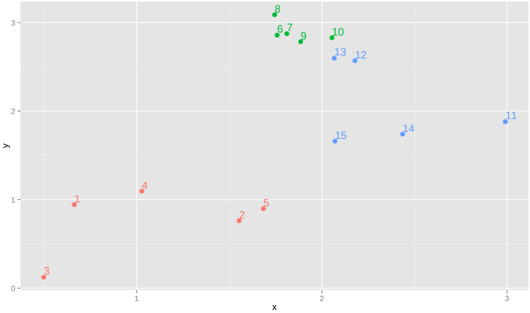
Single linkage

```
cdist = dist(cbind(cdata$x, cdata$y), method='euclidean')
chclust = hclust(cdist, method='single')
plot(chclust, ylab='Distance', xlab=NA, main=NA, sub=NA)
```



Data

```
set.seed(5)
n = 5
nclust = 3
x_means = c(1, 2, 2.5)
y_means = c(1, 2.5, 2)
x = rnorm(n * nclust, rep(x_means, each=n), sd=.4)
y = rnorm(n * nclust, rep(y_means, each=n), sd=.4)
clust = factor(rep(1:nclust, each=n))
id = 1:(n * nclust)
cdata = data.frame(id=id, x=x, y=y, clust=clust)
ggplot(cdata, aes(x=x, y=y)) + geom_point(aes(color=clust), size=3, show_guide=F) +
  geom_text(aes(label=id, color=clust), show_guide=F, vjust=-.3, hjust=0)
```

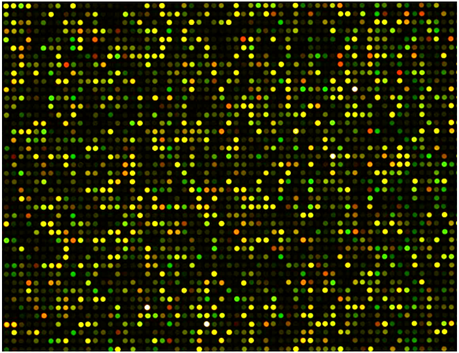


1 Clustering

2 PCA

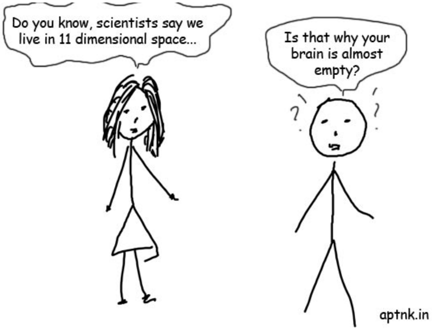
High-dimensional data

- Small n large p
- Few samples n (≈ 100)
- Many features p
  - > 10000 genes
  - > 27000000 CpG sites



Problems

- High storage costs (memory)
- High computational costs (time)
- Visualization?
- Curse of dimensionality

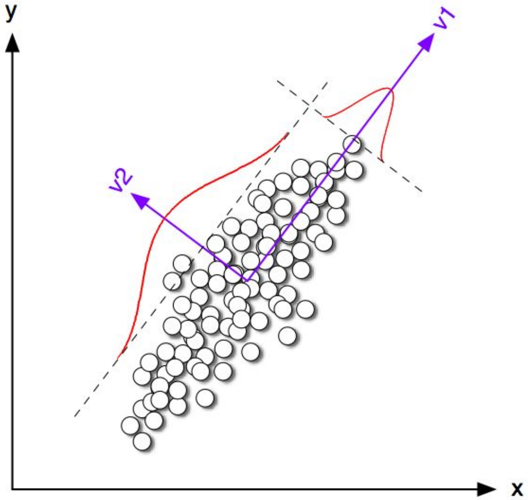

Do you know, scientists say we live in 11 dimensional space...
Is that why your brain is almost empty?
aptnk.in

Principle Component Analysis

- Dimensionality reduction
- Visualization
- Missing values imputation
- Latent factors estimation:
  - Population structure
  - Batch-effects
  - Cell-cycle

Principle components

- Minimize projection error
- Maximize variance



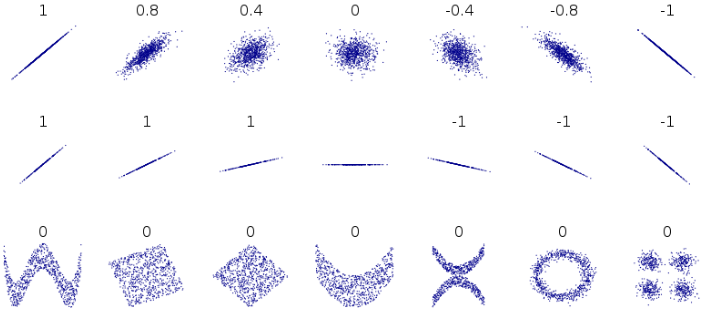
Pearson correlation coefficient

- Measures **linear** dependency between x and y
- $cor(x, y) =\in [-1, +1]$
- $cor(x, y) = 0$: no correlation
- $cor(x, y) = -1$: negative correlation
- $cor(x, y) = +1$: positive correlation

Definition (Pearson correlation coefficient)

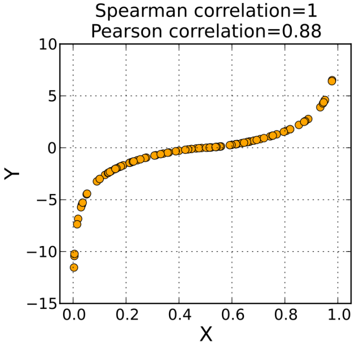
$$\mathrm{cor}(x, y) = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2} \sqrt{\sum_i (y_i - \bar{y})^2}}$$

Pearson correlation coefficient



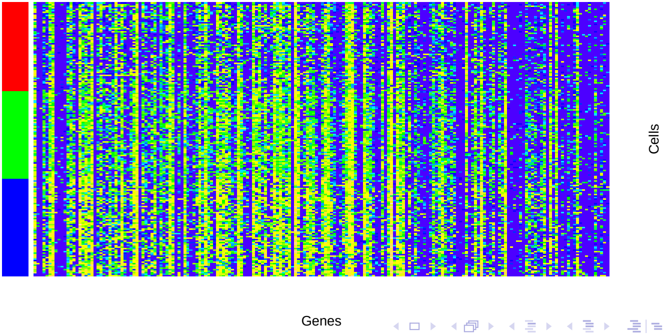
Spearman correlation coefficient

- Measures **monotonic** dependency between x and y
- Pearson correlation coefficient on rank of variables
- $cor(x, y) = \in [-1, +1]$



Embryonic Stem Cell (ESC) data

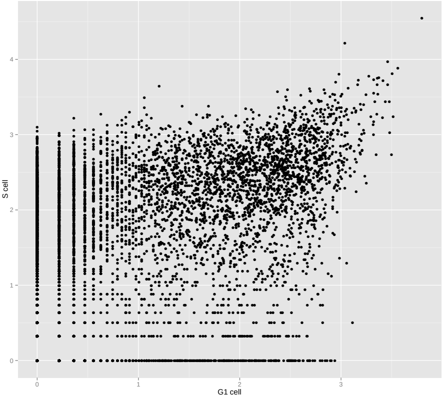
- Single-cell RNA-seq
- $n = 182$ Embryonic stem cells
- $p = 9571$ Genes
- Cell-cycle via Hoechst staining



Correlation between two ESC

```r
c1 <- esc$counts[which(esc$cycle == 'G1')[1],]
c2 <- esc$counts[which(esc$cycle == 'S')[1],]
```

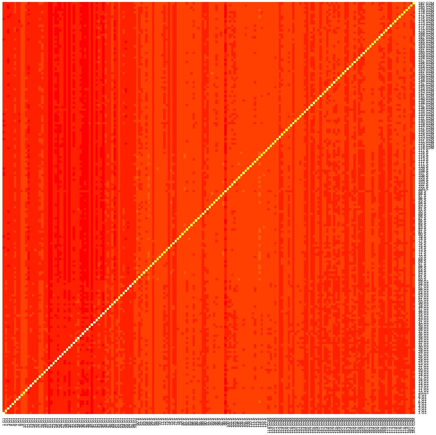
```r
qplot(c1, c2, xlab='G1 cell', ylab='S cell')
```



```r
cor(c1, c2,
    method='pearson')


## [1] 0.547453


cor(c1, c2,
    method='spearman')


## [1] 0.5785747
```

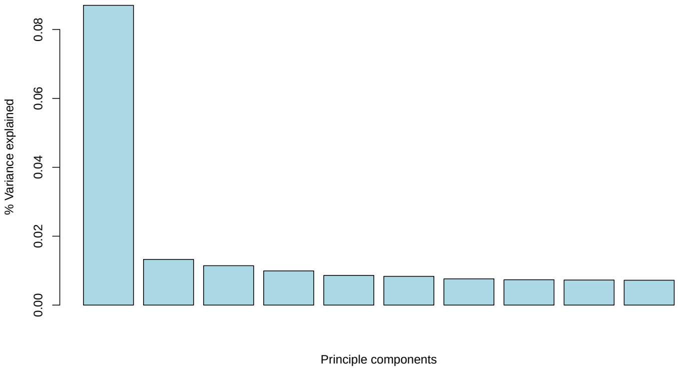
Correlation matrix

```
cor_cells <- cor(t(esc$counts), method='pearson')
heatmap(cor_cells, Rowv=NA, Colv=NA)
```



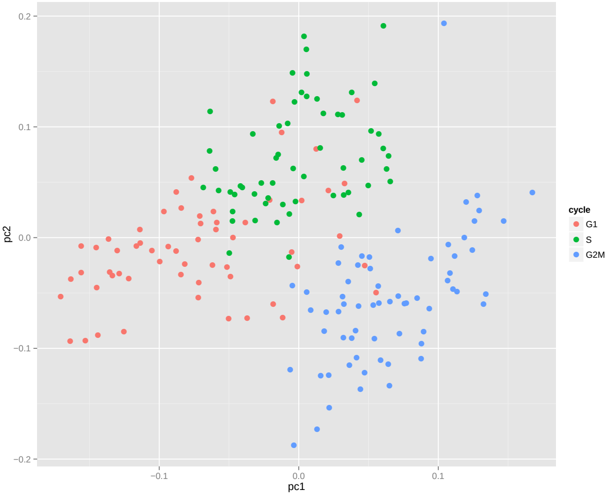
PCA

```
svd <- svd(scale(esc$counts))
dsvd <- as.data.frame(svd$u[, 1:10])
colnames(dsvd) <- paste0('pc', seq(1, ncol(dsvd)))
dsvd$cycle <- esc$cycle
ve <- svd$d^2 / sum(svd$d^2)
barplot(ve[1:10], xlab='Principle components', ylab='% Variance explained',
        col='lightblue')
```



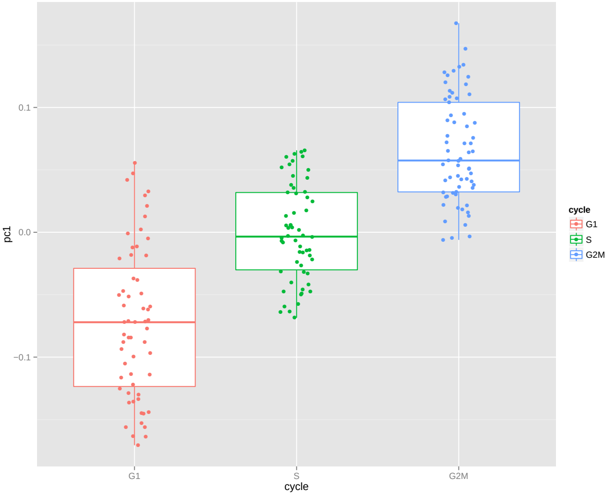
PC1 versus PC2

```
ggplot(dsvd, aes(x=pc1, y=pc2, color=cycle)) + geom_point(size=3)
```



Correlation PC1 and cell-cycle

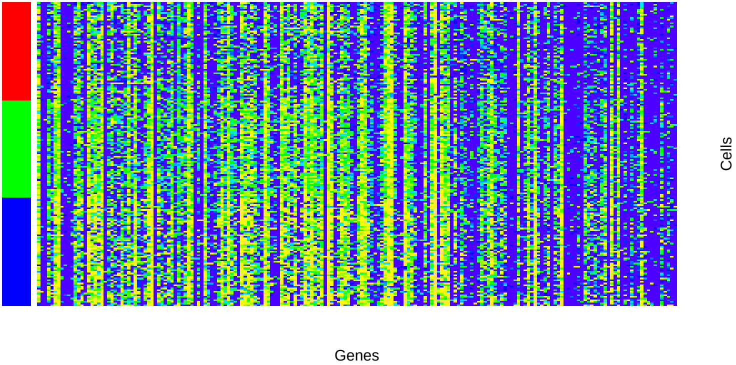
```
ggplot(dsvd, aes(x=cycle, y=pc1, color=cycle)) + geom_boxplot() + geom_jitter(position=position_jitter(width=.1
```



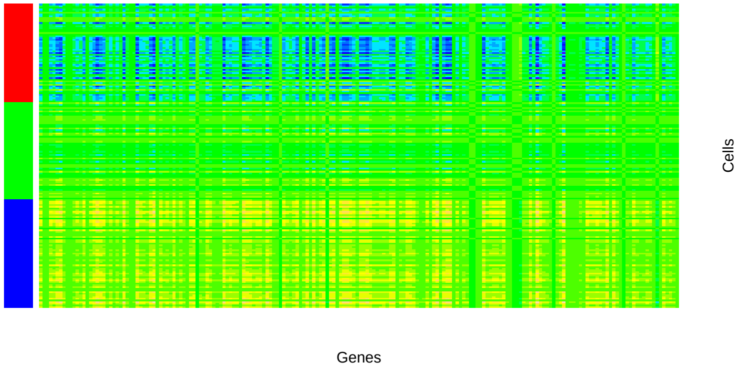
Accounting for cell-cycle

```r
svd_compress <- function(svd, k=1) {
  return (svd$u[, 1:k, drop=FALSE] %*% diag(svd$d[1:k], nrow=k) %*% t(svd$v)[1:k,, drop=FALSE])
}
counts_pc1 <- svd_compress(svd, 1)
counts_npc1 <- esc$counts - counts_pc1
```

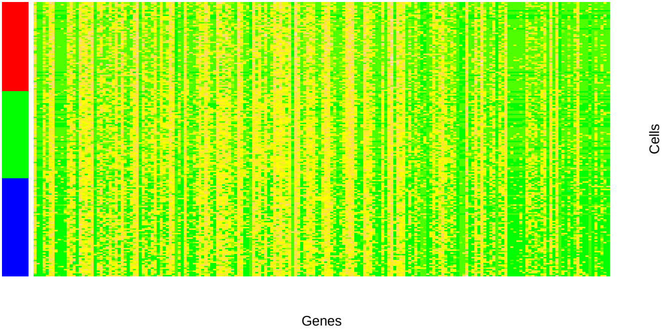
Raw counts



Counts explained by cell-cycle



Counts after adjusting for cell-cycle



Further readings

- David J. Balding and Christopher M Bishop.
  Handbook of statistical genetics.
  Wiley, Chichester [u.a.], 2007.

- Coursera.
  Data analysis and statistical inference.

- Coursera.
  Mathematical biostatistics boot camp 1.

- Bernard Rosner.
  Fundamentals of biostatistics.
  Brooks/Cole, Cengage Learning, Boston, 2011.

- Cosma Rohilla Shalizi.
  Advanced data analysis from an elementary point of view.
  Preprint of book found at http://www. stat. cmu. edu/ cshalizi/ADAfaEPoV, 2013.

Questions

Questions?