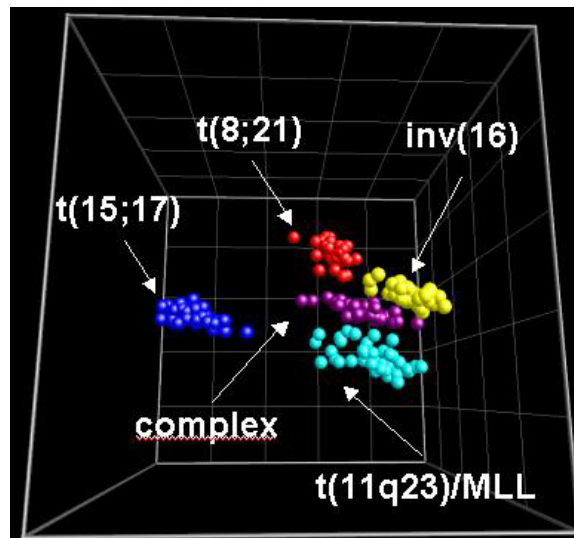# Principal Components Analysis
STAT/GSAT/BIOF 540 2017

Paul Pavlidis

Outline

- Motivation
- Correlation structure of data and dimensionality
- The singular value decomposition & PCA
- Illustrative example
- Additional applications
  - Batch & artifact correction
  - GWAS population structure detection/correction

http://atlasgeneticsoncology.org/Deep/MicroarraysID20045.html

"There are two books devoted solely to principal components … which we think overstates its value as a technique"

- Venables and Ripley, *Modern Applied Statistics with S*, Fourth edition p 305

Search PubMed for "microarray 'principal component analysis'" →
789 results including 14 review articles.

principle component

About 3,010,000 results (**0.22** sec)

Did you mean: *principal* component

The purpose of PCA
(and other multivariate exploratory methods)

- What is the "overall structure" of my data?
  - Where "overall" implies "I want a summary"

- Summarize the data → Dimensionality reduction:

- It is **Unsupervised** – like clustering

Review: data vectors

Assays ("Samples")

Each column is a 54,569 element vector

Genes

Profile for a gene, 59-element vector
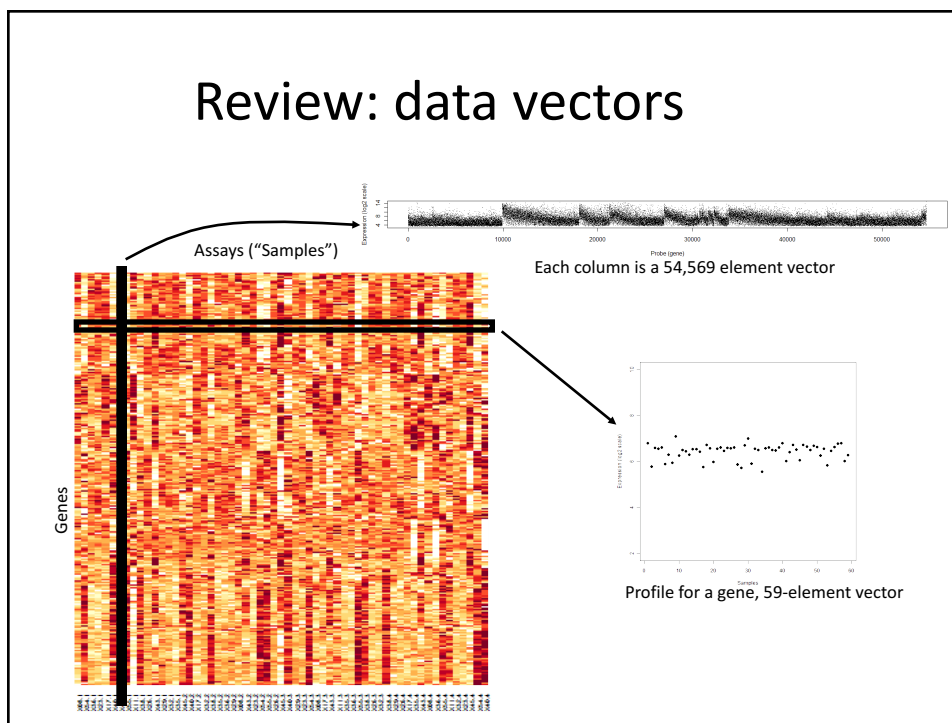
Our data is high-dimensional so…

- We can't "look at it" in that space
- We *can* look at "parts of it" (up to 3 dimensions)
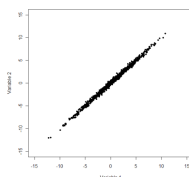
- Or we can look at **projections** into a 2-D space we can visualize.
- Which directions or projections are the good ones?

Why would we want to do this?

- Find a more compact representation of the data.
- "Decompose" the data into "components" that might have some useful interpretation.
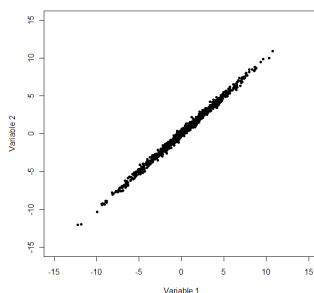 - Explanations
  - Corrections

What is a useful projection?

"White Trash (With Gulls)" by Tim Noble and Sue Webster (flickr user pashasha under Creative Commons license)
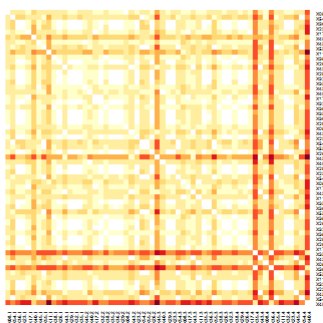
What PCA does

- It choose a "good" set of directions in which to view (project) the data.
- The definition of "good" starts with "find the direction in the data that has the largest variance"

Other decomposition methods have different criteria for "good"

Preliminary: thinking about the correlation structure of data

- Where does this come from?
- What are its properties?

Review: Dot product, covariance

$$\boldsymbol{x} \cdot \boldsymbol{y} = \sum x_i y_i = \cos\theta \|x\| \|y\|$$
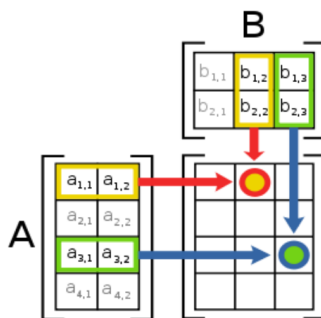
Sample variance

$$s^2(\mathbf{x}) = \frac{\sum_i^N (x_i - \overline{x})^2}{N - 1}$$

Sample covariance

$$\mathrm{Cov}(\mathbf{x}, \mathbf{y}) = \frac{\sum_i^N (x_i - \overline{x})(y_i - \overline{y})}{N - 1}$$

"Are you far from your mean when
I am far from mine?"

Review: Matrix multiplication

- Dot products, done in a systematic way on two set of vectors (matrices). In R, use `%*%` (`?matmult`)

- If $\mathbf{A}$ is a $n \times m$ matrix, $\mathbf{AB}$ is defined only if $\mathbf{B}$ has m rows (number of columns in B doesn't matter)
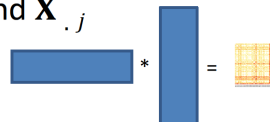

http://en.wikipedia.org/wiki/Matrix_multiplication

Column covariance matrix

$\mathbf{X^T X}$ is the matrix of all pairs of dot products of columns of $\mathbf{X}$

The i, j entry is the dot product of $\mathbf{X}_{.\,i}$ and $\mathbf{X}_{.\,j}$

- $\mathbf{X^T X}/(n-1)$ is the covariance matrix.
- Recall: When the data is standardized, the dot product, covariance and correlation matrices are the same (up to a constant n-1)
- When you see $\mathbf{X^T X}$, think "covariance of cols. of $\mathbf{X}$"

Covariance matrix of the rows

- Formed from $\mathbf{XX^T}$
- BIG, because number of rows can be 50,000 or more.
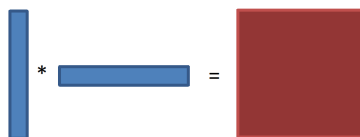
- We won't work with this directly for now – let's go back to the column correlation matrix.

Three thought experiments

Point to this exercise:

- Think about correlation structure of data
- Understand that the data can be described by at most m independent factors ("dimensions").

Launching point is the column correlation/covariance matrix.
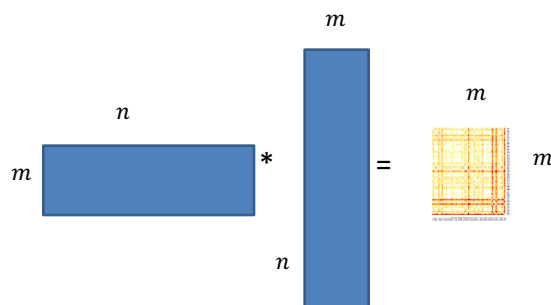
# What does the **column data** look like:

Assume data are **NOT** standardized by rows; n rows, m columns

1. If all the samples (assays) are perfectly correlated?
2. If there is no correlation among the samples?
3. If there are two types of samples, which are perfectly correlated within the groups, and completely uncorrelated across?
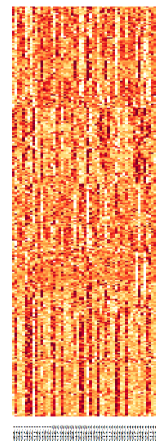
What happens to the column correlation matrix:

1. If all the samples (assays) are perfectly correlated?
2. If there is no correlation among the samples?
3. If there are two types of samples, which are perfectly correlated within the groups, and completely uncorrelated across?
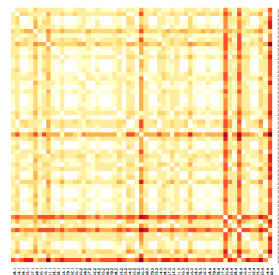
Discussion of the second case

If there is no correlation among the samples, they are vectors all at right angles to each other (orthogonal).

Note: While the samples have $n >> m$ elements, they only **span** an $m-$dimensional subspace of $\mathbb{R}^n$.

$\implies$ Realistic sample data can be described by a combination of **at most m vectors** (theoretically possible for the samples to be uncorrelated.)

Basis and Span

- A **basis** for $\mathbb{R}^m$ (m-dimensional space) is (roughly) a set of m vectors from which you can "make" any vector in $\mathbb{R}^m$
  - The basis **spans** $\mathbb{R}^m$
- An **orthogonal basis** is a nice kind of basis: the m vectors are orthogonal
  - These are like "axes". In $\mathbb{R}^2$ one such basis is [0,2],[2,0].
  - Any pair of orthogonal 2-vectors is a basis in $\mathbb{R}^2$
    - For example, [0.5,-0.5],[0.5,0.5]
- Even nicer is an **orthonormal basis**, where the vectors are normalized to length 1.
  - Standard basis for $\mathbb{R}^2$: [0,1],[1,0]

Example: 2D basis and a 1D subspace

- The X and Y axes [0,1],[1,0] are an example of an orthonormal basis
- Any vector v in the X,Y plane can be described by a combination of any two basis vectors.

$$\boldsymbol{v} = a_1 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + a_2 \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

- v is a **linear combination** of x and y. The "weights" are given by the vector a, which are (basically) the "coordinates" in the basis.
- **Subspace**: The X axis, while "2-dimensional", only spans a line.

$$\boldsymbol{v} = a_1 \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

- You need at least m vectors to span ("fill") a m-dimensional space
- **The situation for the columns of our data is the like this:** they are in an (at most) m-dimensional dimensional subspace.

What does the **row data** look like:

1. If all the samples are perfectly correlated?

2. If there is no correlation among the samples? (will the rows have no correlation too?)

3. If there are two types of samples, which are perfectly correlated within the groups, and completely uncorrelated across?
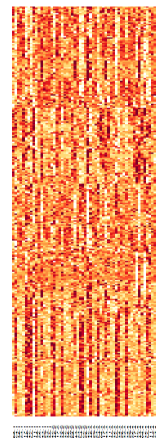
Discussion of second case for rows

The rows are more obviously "m-dimensional" – they are each completely described by m values.

- Each row (gene) is a linear combination of (at most) m orthogonal vectors.

- This guarantees that the rows (genes) will show correlations.

2D Example

- In 2D, each point has an "X" and "Y" coordinate.

- If we have more than two (different) vectors in this space, they can't *all* be at right angles to each other. The third one will always be correlated with the others.

- This is the same situation for our thousands of rows of data.
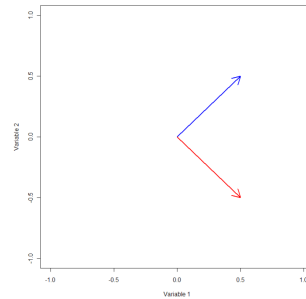
Summary so far

- We can think of *both* our genes *and* our samples as combinations of no more than m vectors that form orthogonal bases (one for genes, one for samples)

- It makes sense to find "good" bases.

- We'll end up calling one reasonable choice the **eigengenes** and the **eigensamples** (or eigenassays).

  - Finding them is easy (with computer)
  - Some of the dimensions are likely to be more interesting than others

Eigenvectors (square matrix only)

- A special vector that is only stretched by the linear transformation represented by the matrix; not rotated.

- The relative amount of stretching is the eigenvalue

- Captures an idea of "size" of the matrix

Eigenvectors and PCA

- We compute the eigenvectors (and the eigenvalues) of the covariance matrix
- These define the **principal components**
- In the data space, they are the "directions of maximal variance" – projections on these vectors have max. variance.
- Intuition: Covariance matrix has information about variance in the data
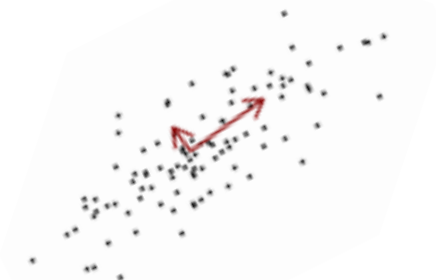
In practice, we use **singular value decomposition** (SVD) to find the PCs.
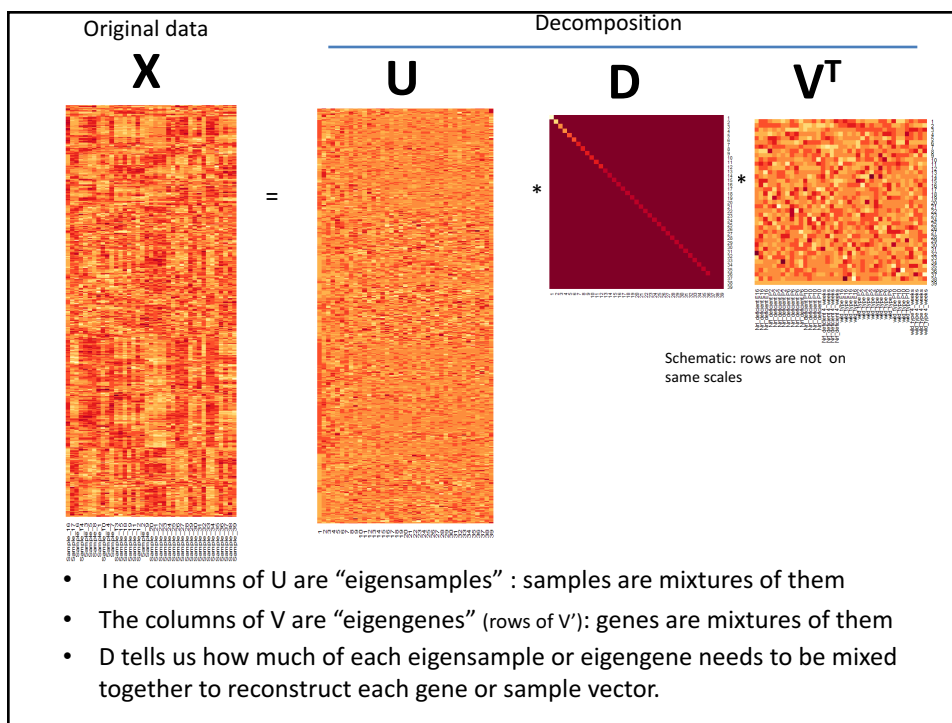
Singular value decomposition

We can decompose any $n \times m$ matrix X into two orthonormal matrices and a diagonal matrix of **singular values**.

$$X = UDV^T$$

U $(n \times m)$ – left singular vectors - orthonormal basis for the row space of X

V $(m \times m)$ – right singular vectors - orthonormal basis for the column space of X

D (diag. $m \times m$) contains the singular values; eigenvalues of $Cov(X)$ are $D^2/(n-1)$

Original data
X

Decomposition
U   D   V^T

Schematic: rows are not on same scales

- The columns of U are "eigensamples" : samples are mixtures of them
- The columns of V are "eigengenes" (rows of V'): genes are mixtures of them
- D tells us how much of each eigensample or eigengene needs to be mixed together to reconstruct each gene or sample vector.

Exploring data with SVD/PCA

Extensive worked example to give you a feel for how it works in a relatively interesting case.

SVD/PCA on the
photoreceptor dataset (GSE4051)

View the data three ways:

- Sum of "components"
- Projections of samples on the eigensamples (sample loadings; columns of U)
- Correlation of genes with eigengenes (gene loadings; columns of V)
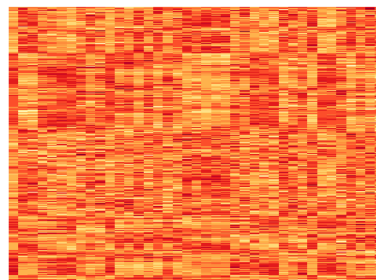
Visualizations:

- 39 samples organized by genotype and/or development stage.
- Rows clustered unless otherwise stated

Note: I only used at most 500 randomly-selected genes

Preparing the data

- Rescaling; Standardizing
  - If we don't at least mean-centre, the first PC will represent the "average expression level"
  - This constrains the next direction (must be orthogonal)
  - In fact I "double-standardized" the data so the rows and the columns both have mean 0, variance 1.
    - Just standardizing the rows gives similar results

- Do we start with X or X^T?
  - SVD doesn't care, but convention is to start with X (rows>cols)
  - For PCA, it does matter (R functions only give you either the eigengenes or eigensamples), but we get both at the same time with SVD
  - The number of eigenvalues corresponds to the smaller dimension

Supervised PCA

"A principal component analysis is shown based on gene expression signatures from n=800 genes which we identified to be differentially expressed…"

http://atlasgeneticsoncology.org/Deep/MicroarraysID20045.html
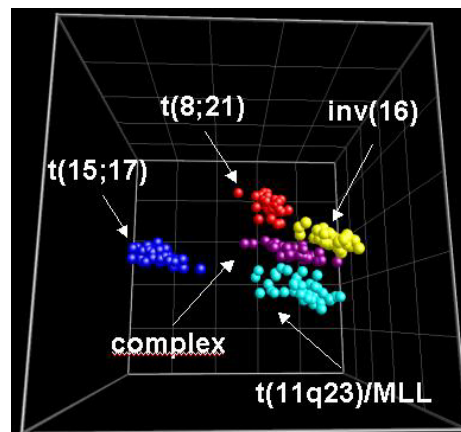
Unsupervised vs. Supervised

Unsupervised

Supervised

http://www.ncbi.nlm.nih.gov/pmc/articles/PMC1851158
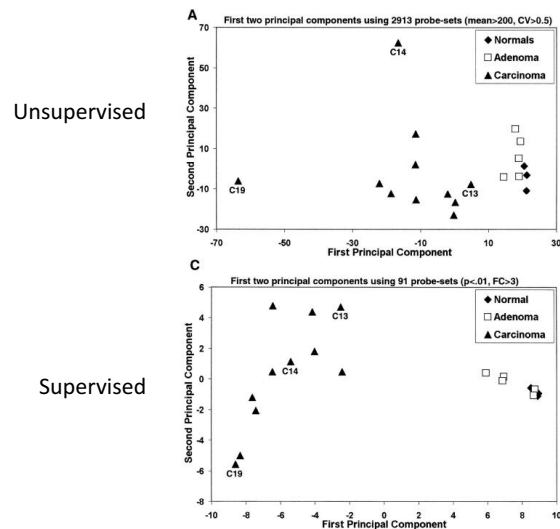
R notes

- `svd(dat)`
  - Provides u, d (vector) and v; columns of v contains the eigenvectors.
  - Eigenvalues: where n is nrow(dat)
- `prcomp(dat)`
  - Calls `svd(dat)`; Gives you `stdev` (square roots of eigenvalues) and `rotation` (columns are the eigenvectors) a.k.a. loadings
- `eigen(cov(dat))`
  - Gives you eigenvalues and eigenvectors
- `princomp` – prcomp is preferred.
- The signs are arbitrary: you may get .

Scree plot

- Shows the relative magnitudes of the eigenvalues
- A steep plot indicates a lot of global correlation structure in the data

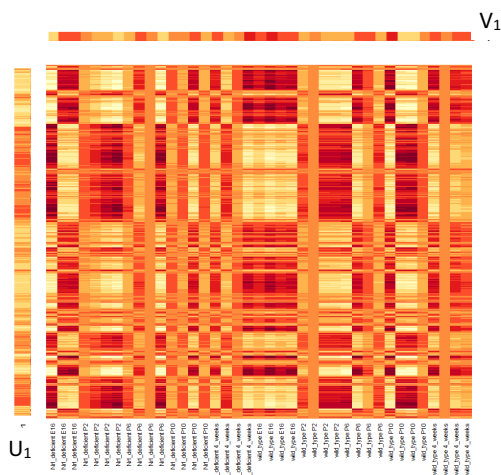plot(pca)
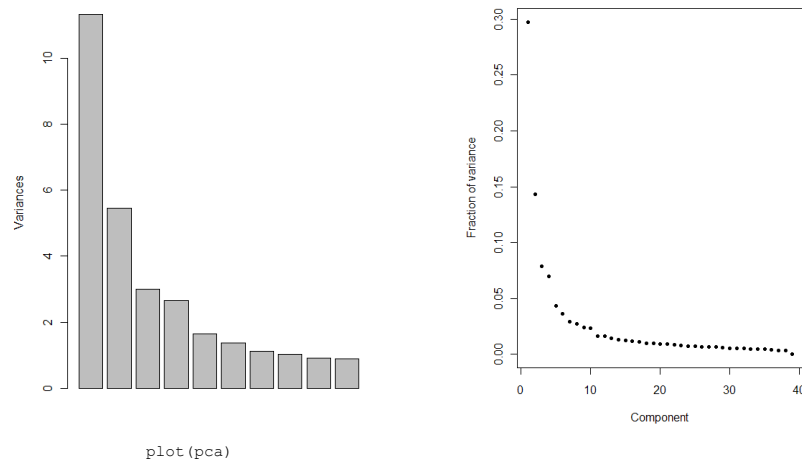
Reconstruction of a matrix with SVD

Original data (scaled etc.)

First component

$U_1 * D_1 * V_1^T$

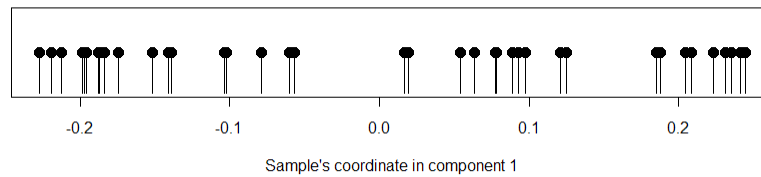75.3

(30% of the variance)

Original data
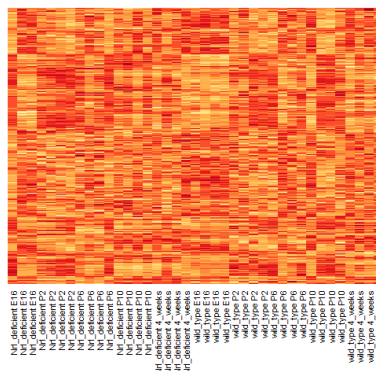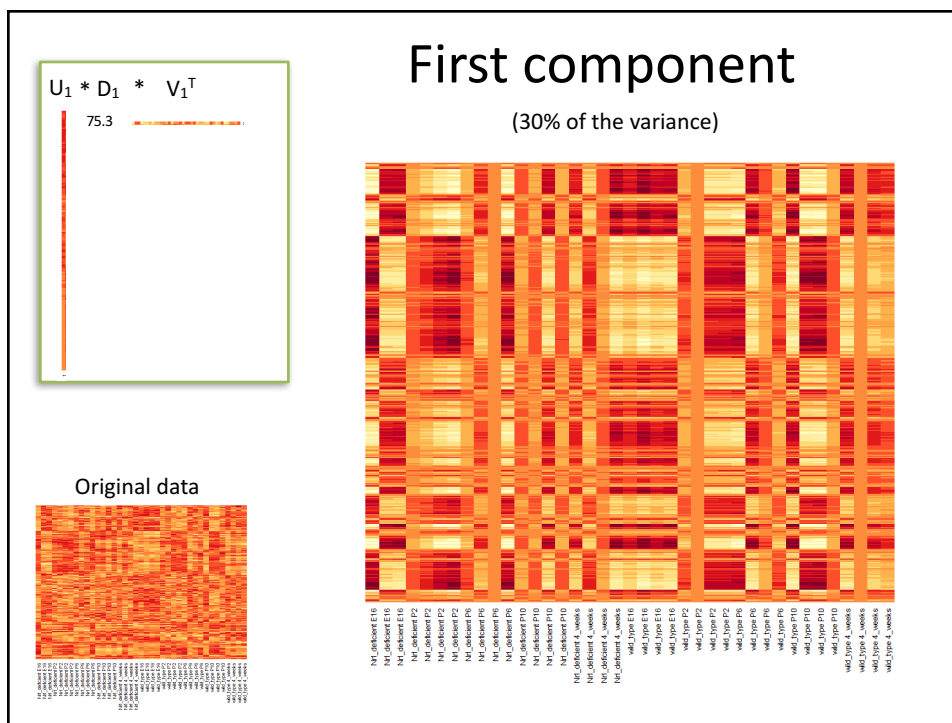
These data are "1-dimensional"

- All the columns are "the same" (U_1) up to a scalar.
  - The scalars are $D_1 V_1'$

- It's also true for the rows, which are weighted copies of V_1

- This is true for any of the components…

V_1^T

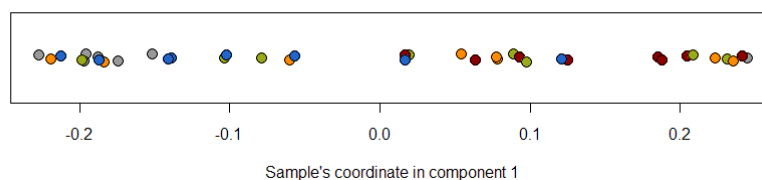U_1

Coordinates of the samples in the first eigengene dimension

Sample's coordinate in component 1

- This is dimensionality reduction

Values are taken from V_1.

V_1^T
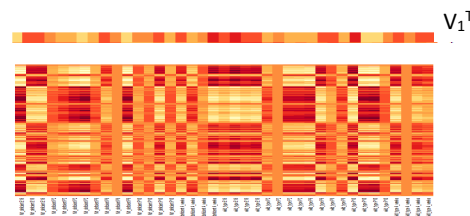
It might "mean something"

Coloured by developmental stage

Sample's coordinate in component 1

(There's a better view later)

Genes projected on first eigensample

Gene's coordinate in component 1

Values are first column of U

Not as easy to make sense of, since we don't have any "explanatory variables" for genes (yet)

Second component
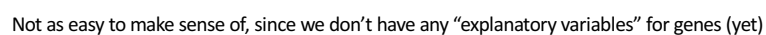
(15% of the variance)
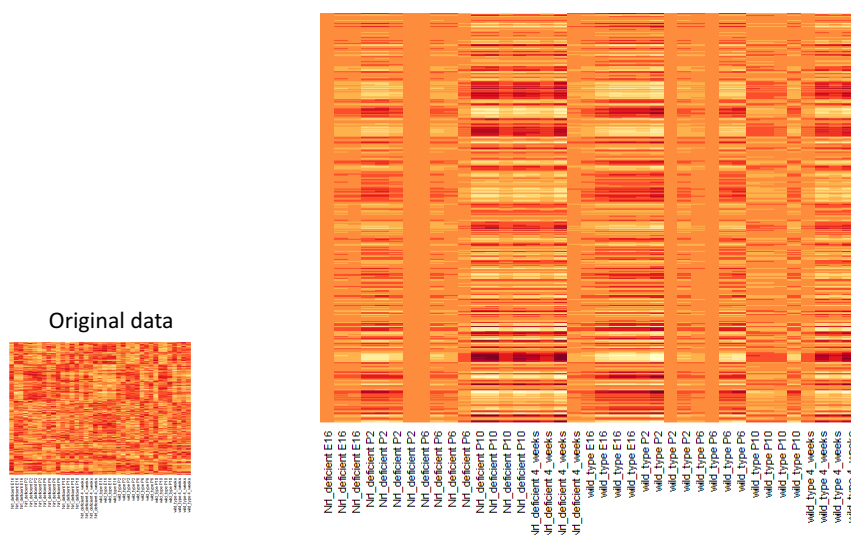
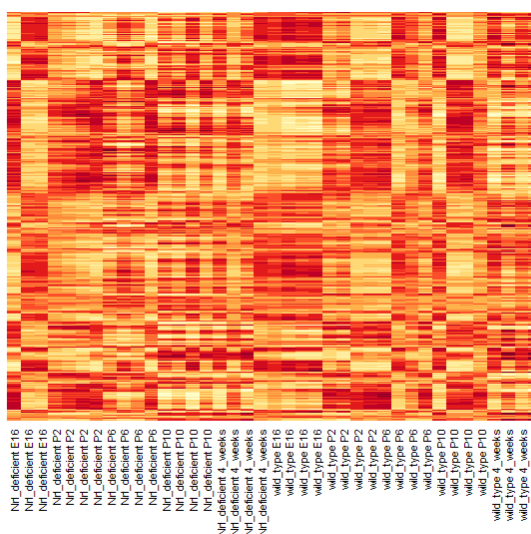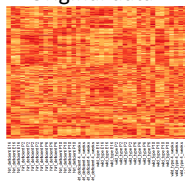Original data

Third component

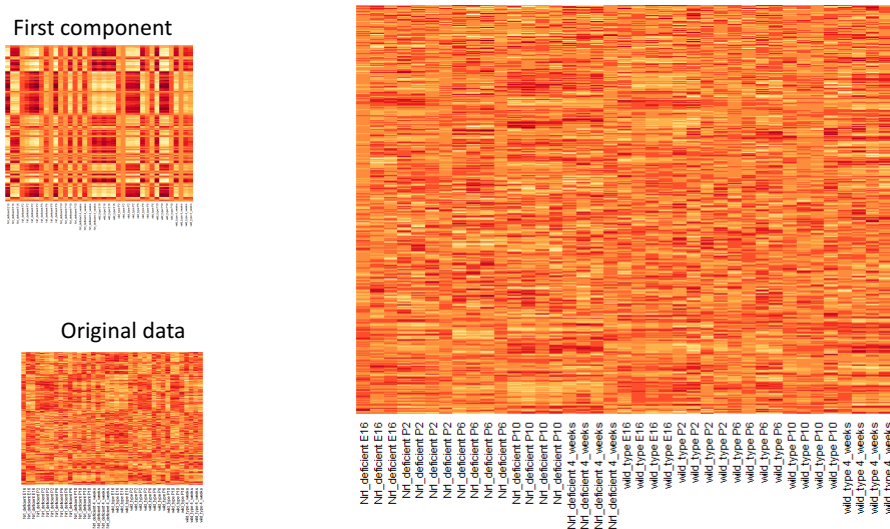(8% of the variance)

Original data
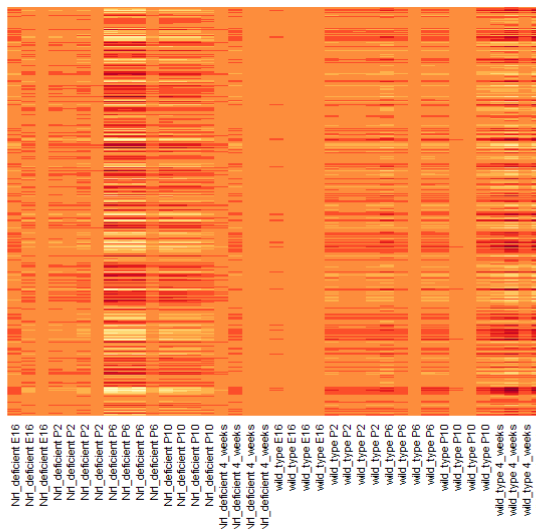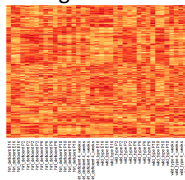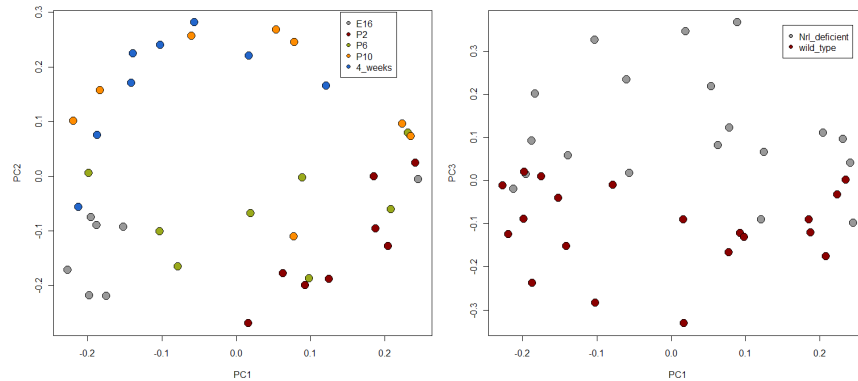

Reconstructing: 1+2+3

"3-Dimensional"

Original data

Reconstructing: first 10 components

Original data
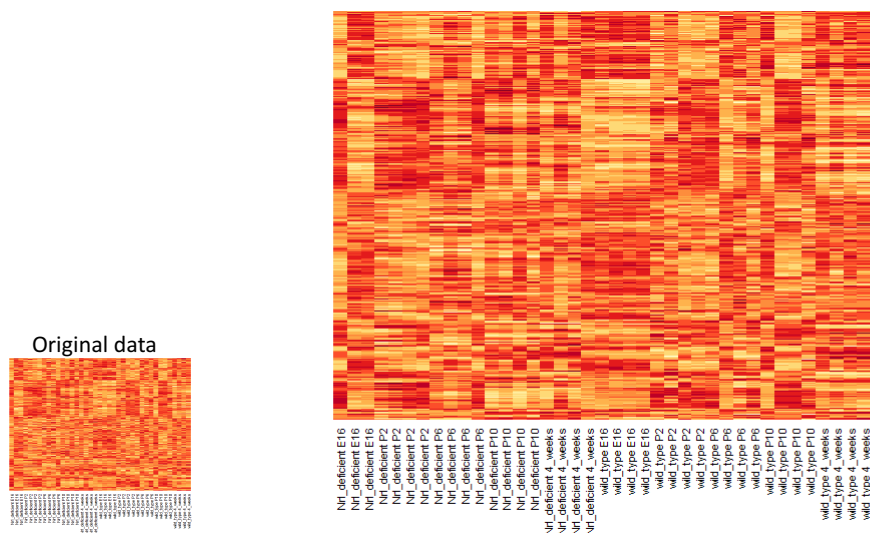
Remove the first component

First component

Original data

Remove the first 10 components

What would happen if we ran SVD on this data?

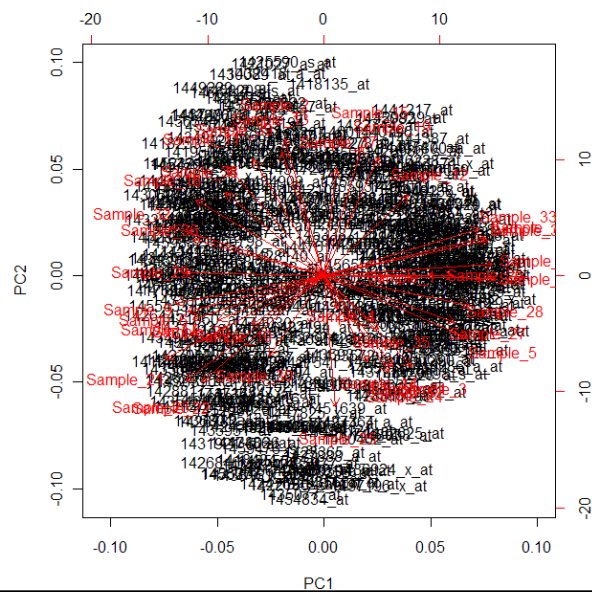Original data

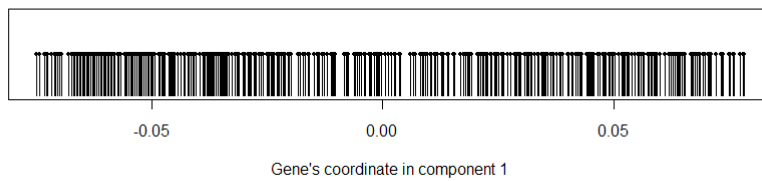# Visual summary of PCs

Closer look

Values plotted are columns of v, as in `plot(v[,1], v[,2])`

Not as useful: `biplot(pca)`

Correlation view
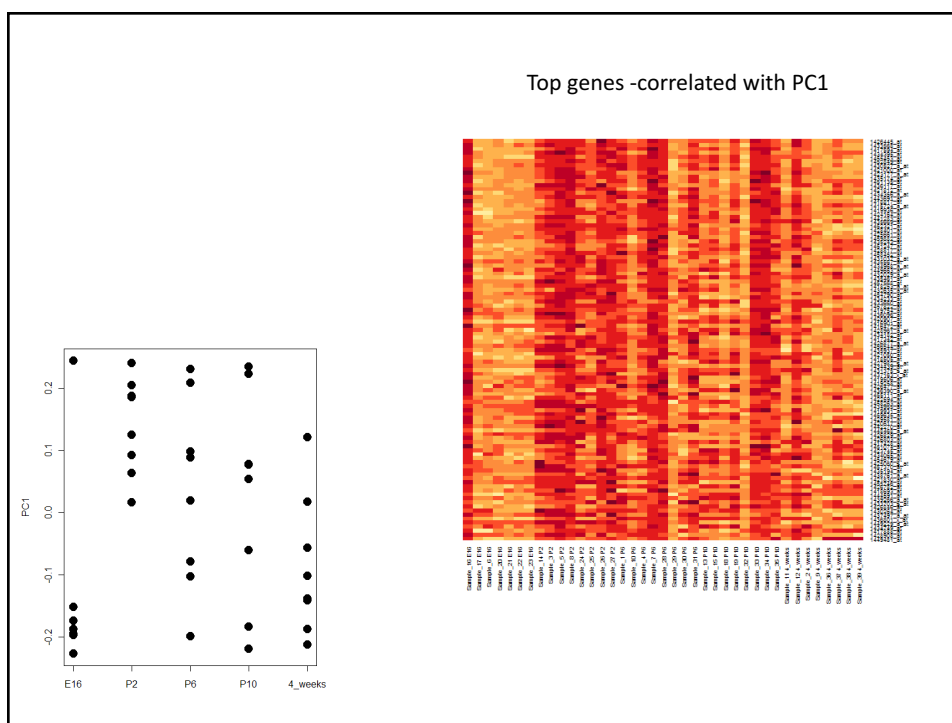
- Show genes "most loaded" with each of PC1, PC2, PC3 (first three columns of V)
- This is just the order given by the values in the columns of U.

Gene's coordinate in component 1

- (Dot products of data with PC gives same results, as does correlation if data are standardized).

Top genes +correlated with PC1

Top genes -correlated with PC1

Top genes +correlated with PC2
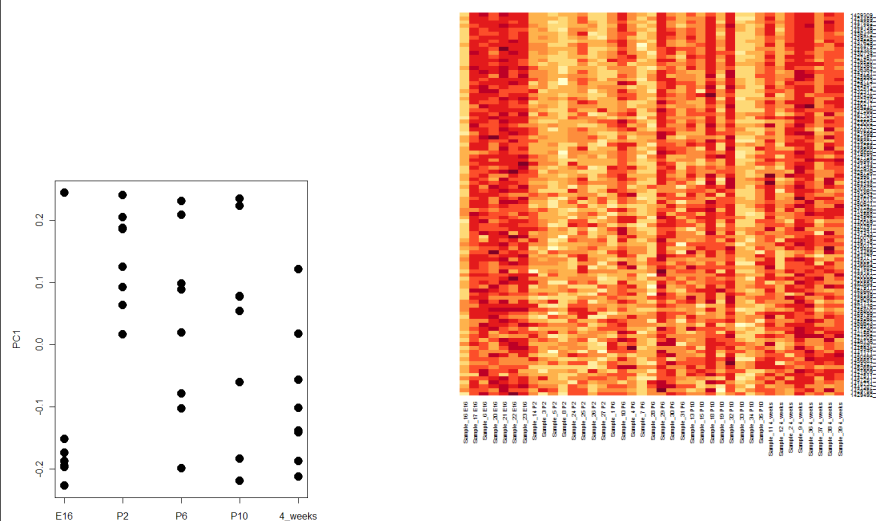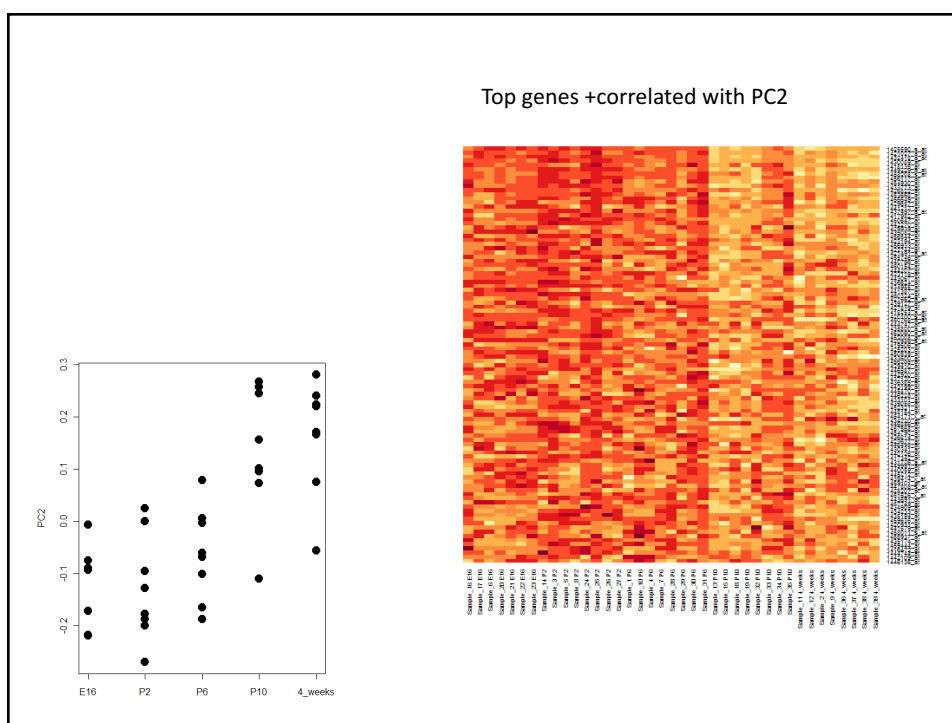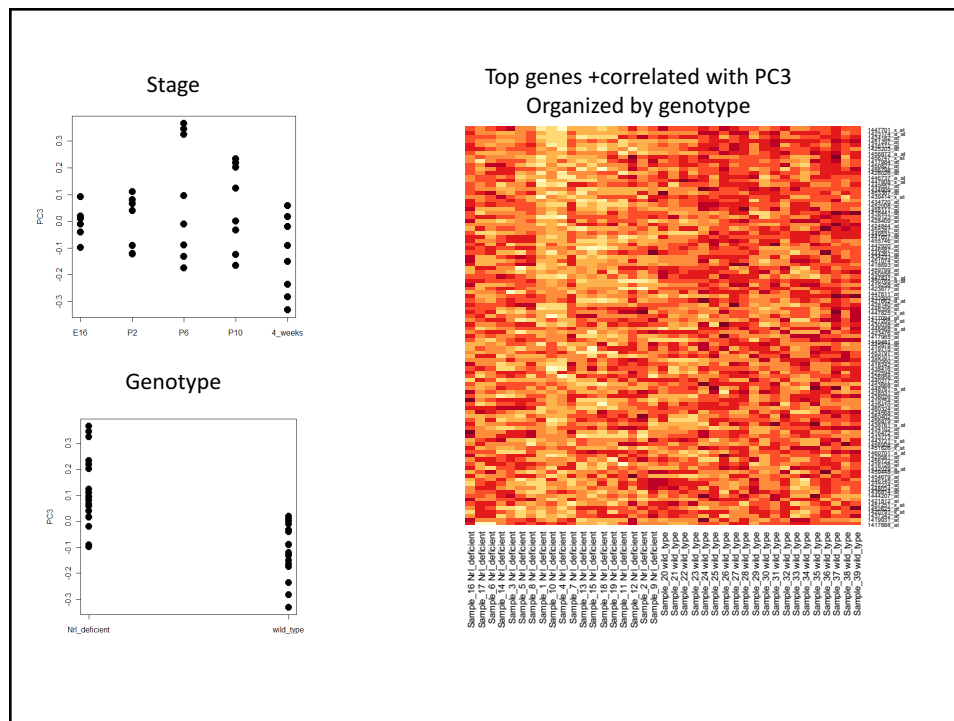
Caveats for PCA

- A principal component is not (automatically) a specific biological or technical signal
- PCA can fail if the data is very "non-Gaussian"
  - It assumes that the interesting directions are along lines, and are orthogonal.

What is the PCA like if:

1. All the samples are perfectly correlated?

2. If there is no correlation among the samples?

3. There are two types of samples, which are perfectly correlated within the groups, and completely uncorrelated across?

More applications

SVD as a batch correction

- Nielsen et al. 2002 Lancet 359
- Switched array types in the middle of the experiment
- Causes a major artifact ...

**Molecular characterisation of soft tissue tumours: a gene expression study**

Torsten O Nielsen, Rob B West, Sabine C Linn, Orly Alter, Margaret A Knowling, John X O'Connell, Shirley Zhu, Mike Fero, Gavin Sherlock, Jonathan R Pollack, Patrick O Brown, David Botstein, Matt van de Rijn

Warning: I don't advocate doing batch correction this way; it's still a good illustration

---
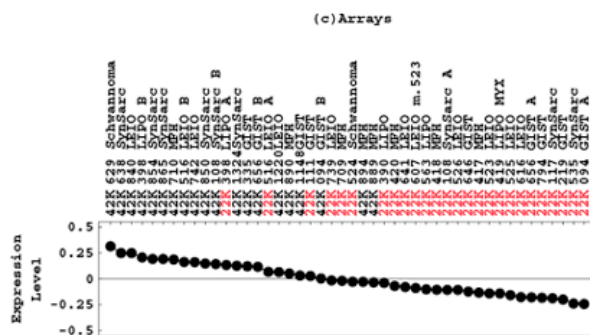
Aside: this "eigengene" does not *perfectly* capture "array type" – which makes you wonder what we're really trying to remove.

http://genome-www.stanford.edu/sarcoma/Supplemental_data.shtml

Set this singular value to zero

Webfigure 3a

http://genome-www.stanford.edu/sarcoma/Supplemental_data.shtml

After SVD

Webfigure 5a

http://genome-www.stanford.edu/sarcoma/Supplemental_data.shtml
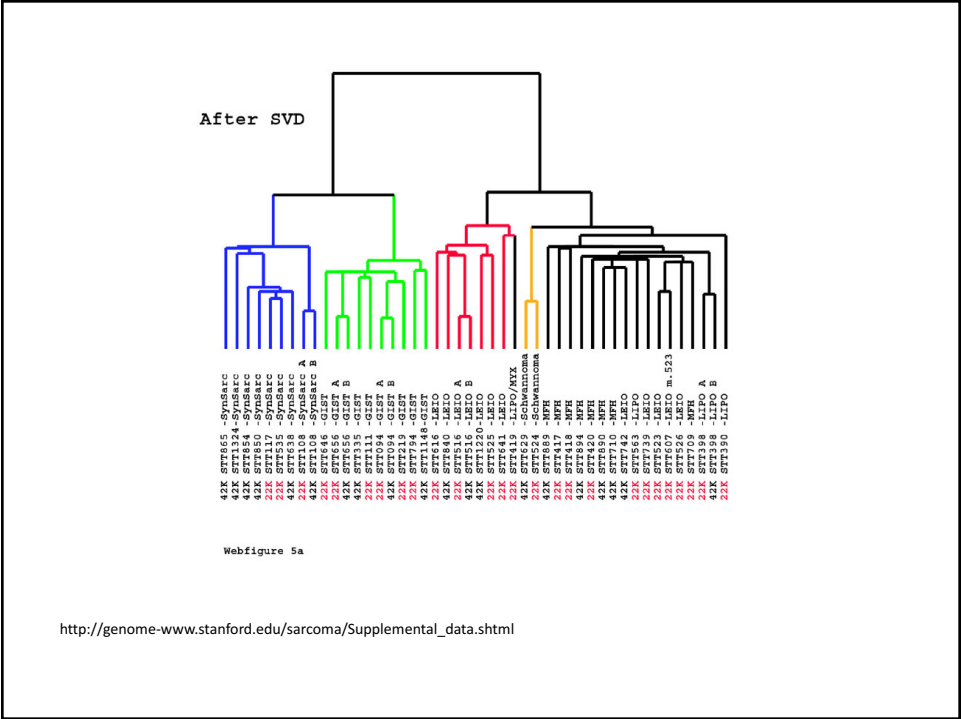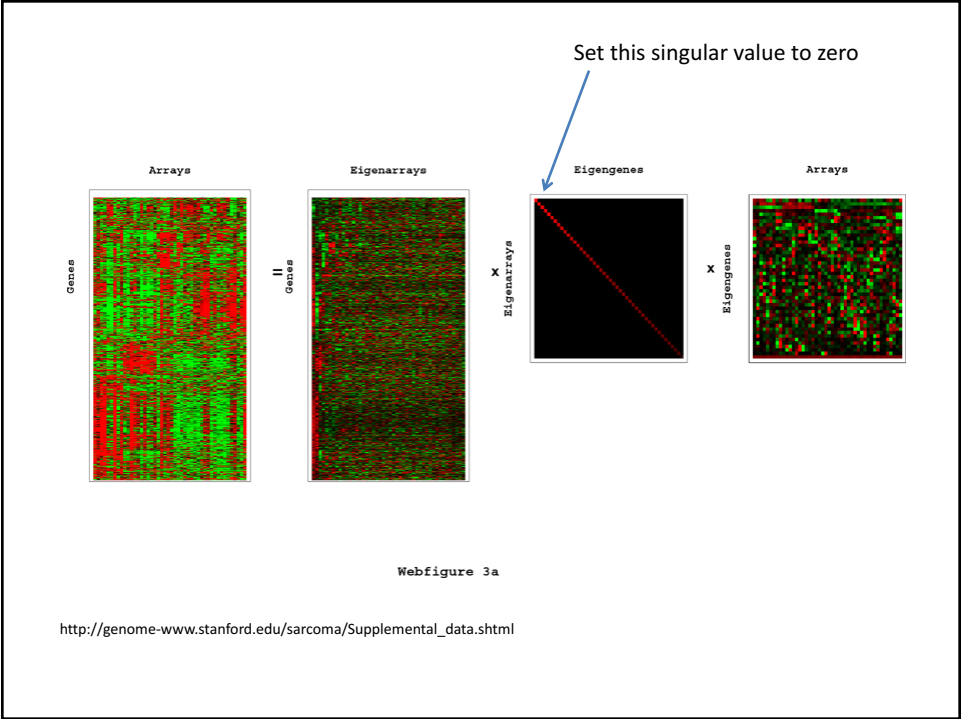
Surrogate variable analysis

- In the sarcoma example, it was easy to figure out that PC1 had something to do with "batch"; but could have used an explicit batch correction instead.
- What if we have "big components" that we don't understand?
- **Surrogate variable** proposal: remove them anyway

Surrogate variable analysis: outline of algorithm

1. Remove effects of known variables of interest using regression
2. The residual should be meet some criterion of being "completely noise" (based on distribution of singular values under the null). If not:
3. Estimate (reconstruct) the "surrogate variables" from the significant eigengenes.
4. These are then treated as covariates in downstream analysis.

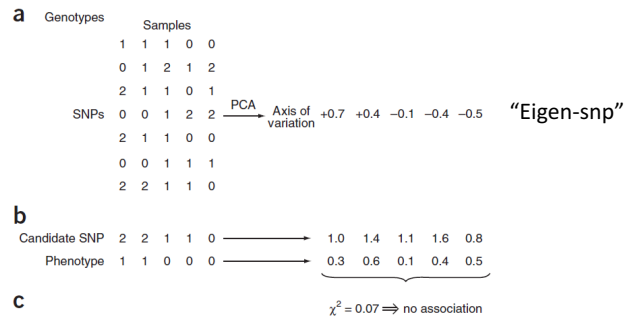# GWAS population stratification detection (and correction)

Figure 1 The EIGENSTRAT algorithm, illustrated on simulated data. (a) Principal components analysis is applied to genotype data to infer continuous axes of genetic variation; a single axis of variation is illustrated here. (b) Genotype at a candidate SNP and phenotype are continuously adjusted by amounts attributable to ancestry along each axis, removing all correlations to ancestry. (c) After ancestry adjustment, an association statistic between genotype at the candidate SNP and phenotype shows no significant association.

Price et al., Nature Genetics 38:8 2006

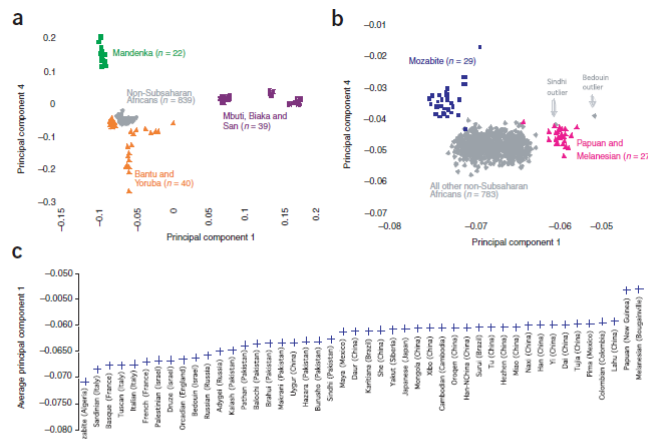The correction approach here is is similar to SVA

Figure 1 PCA continues to provide evidence of important migration events. (a) We carried out PCA on 940 individuals from the Human Genome Diversity Project that were scanned at approximately 650,000 SNPs[11] using data from 101 sub-Saharan African samples to define the PCs (Mandenka, Bantu from Kenya and South Africa, Yoruba, San, Mbuti Pygmy and Biaka Pygmy). We carried out the analysis on samples blinded to population labels (the coloring of samples was only carried out after the analysis). We plotted principal component 1 (negative values are more Bantu-related) and principal component 4 (positive values are more closely related to the Senegalese Mandenka). (b) Outlying populations are the Mozabite, who are more Mandenka-related, reflecting recent gene flow across the Sahara, and Papuans and Melanesians, who have inherited less Bantu-related gene flow. (c) To reveal the west-to-east gradient of Bantu-related ancestry across Eurasia, we averaged the first PC for each of the non-African populations and plotted the populations in rank order.

Reich et al., Nature Genetics 40:5 2008