Option 1: Woody Walk

$$\int \mathrm{d}s = 5\,\mathrm{km}$$

$$\Delta z = 100\,\mathrm{m}$$

$$\Delta t \approx 1\,\mathrm{h}$$

$$\max_t f_{\mathrm{heart}} = 90\,\mathrm{min}^{-1}$$

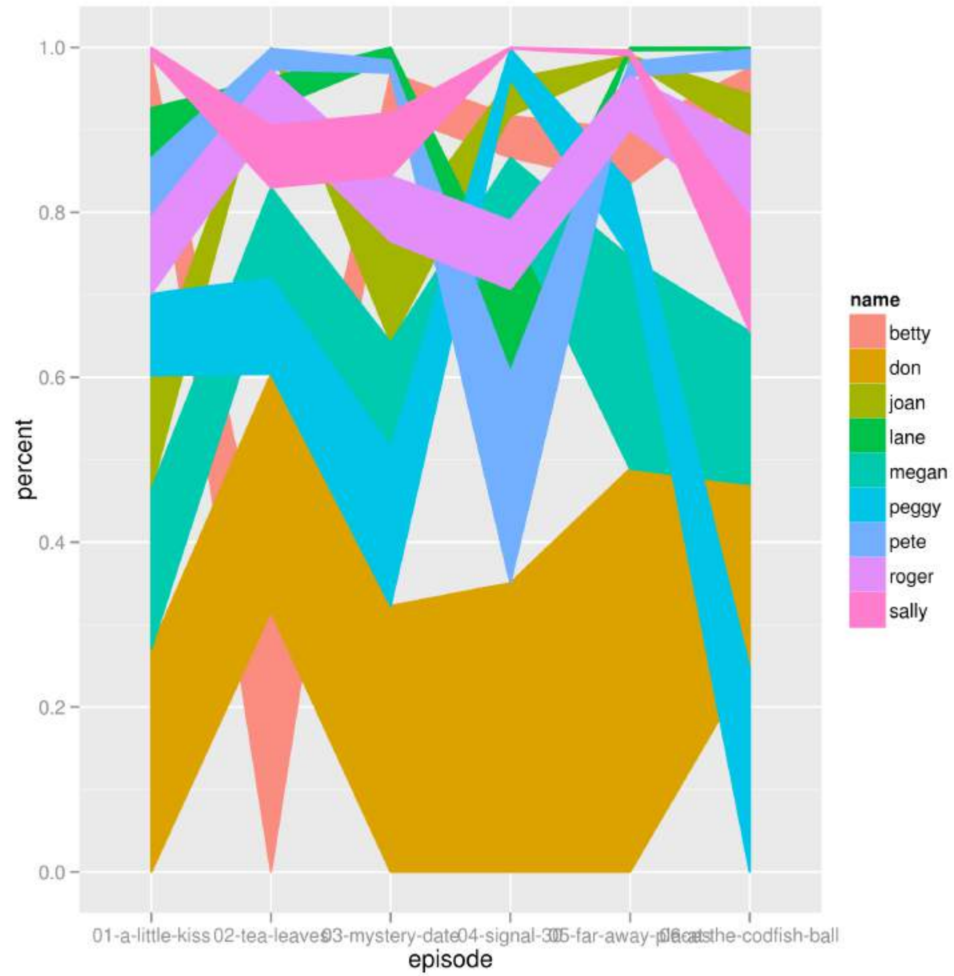
Option 2: Rifugio Plose

$$\int \mathrm{d}s = 8\,\mathrm{km}$$

$$\Delta z = 600\,\mathrm{m}$$

$$\Delta t \approx 2\,\mathrm{h}$$

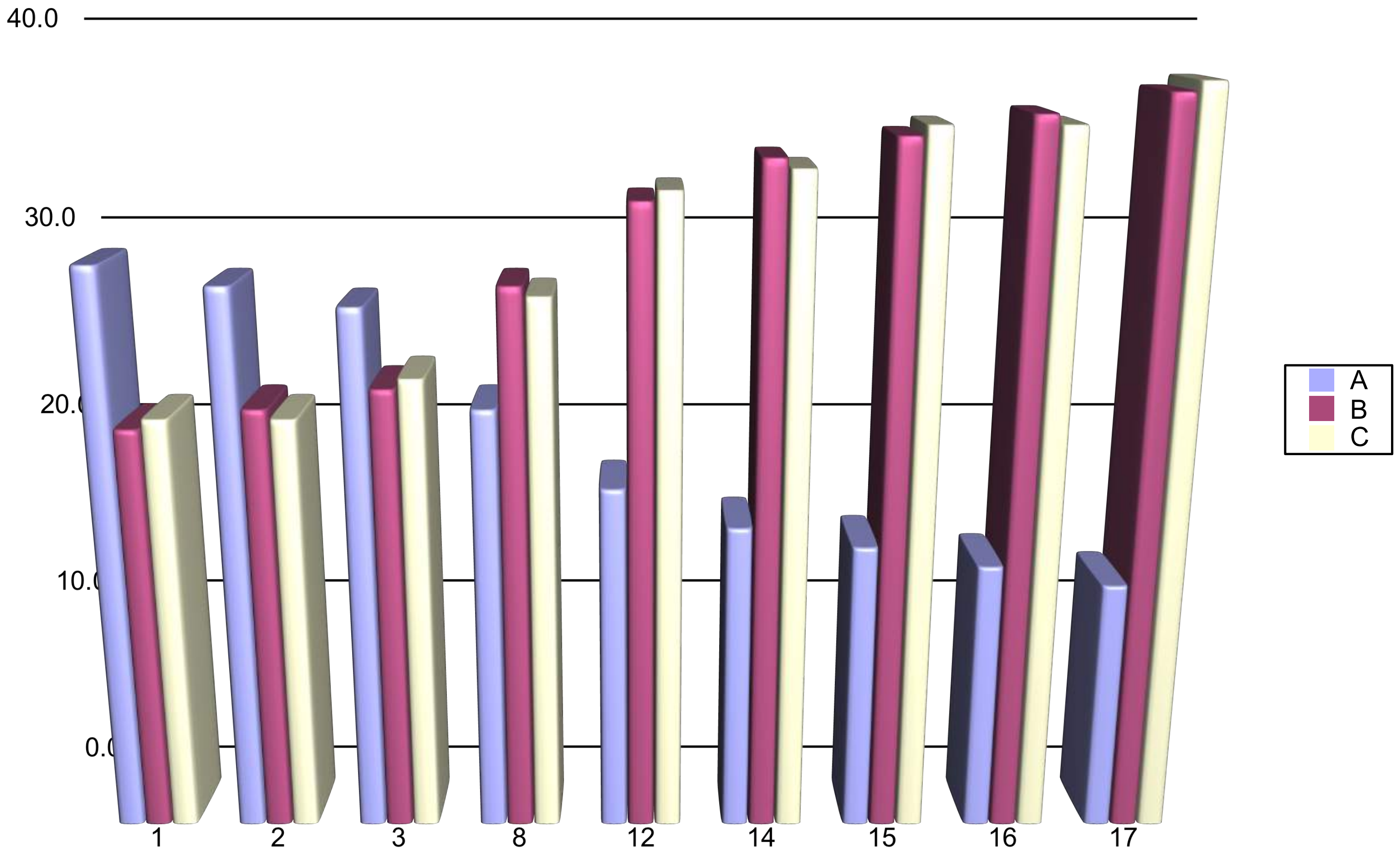
$$\max_t f_{\mathrm{heart}} = 165\,\mathrm{min}^{-1}$$

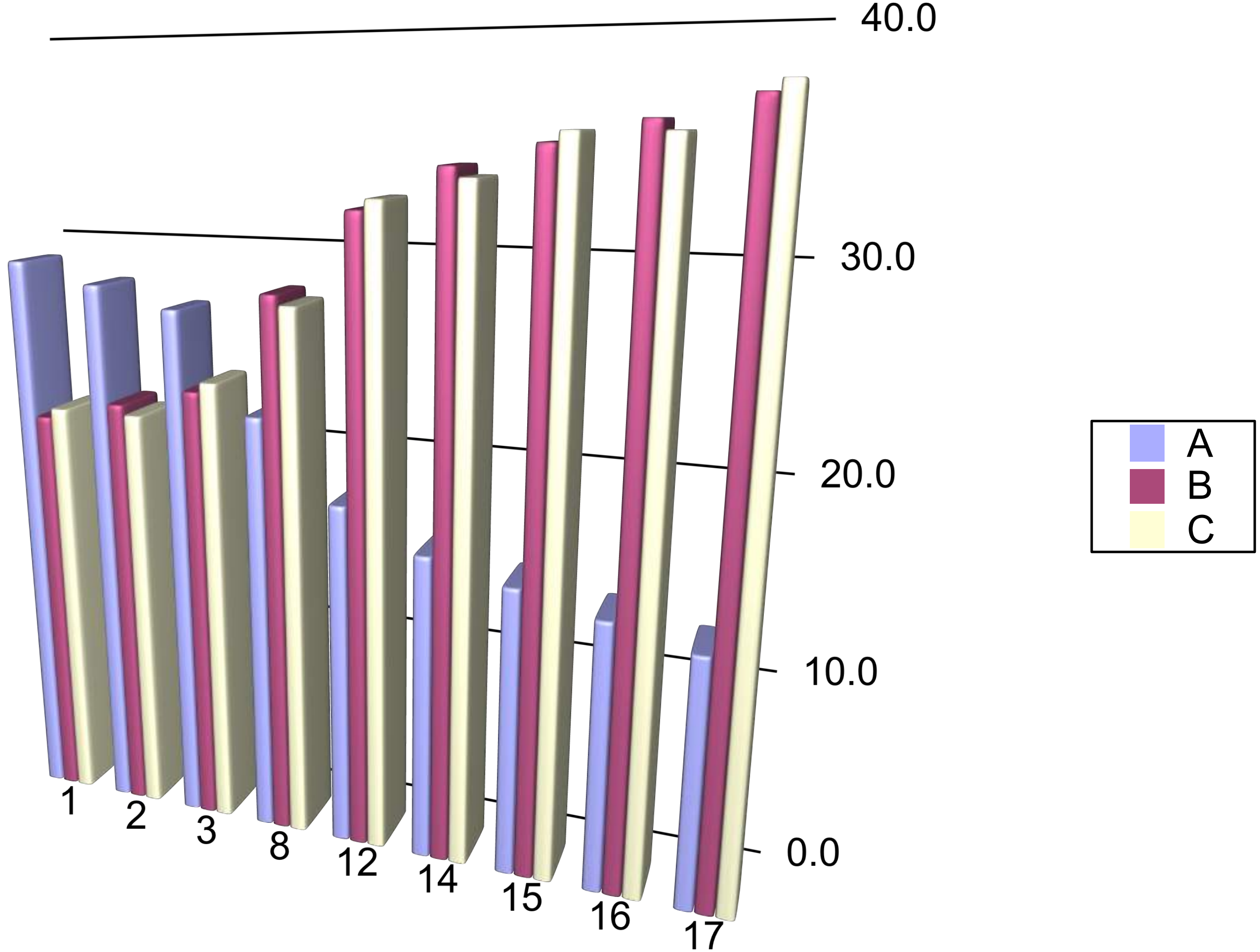


Graphics

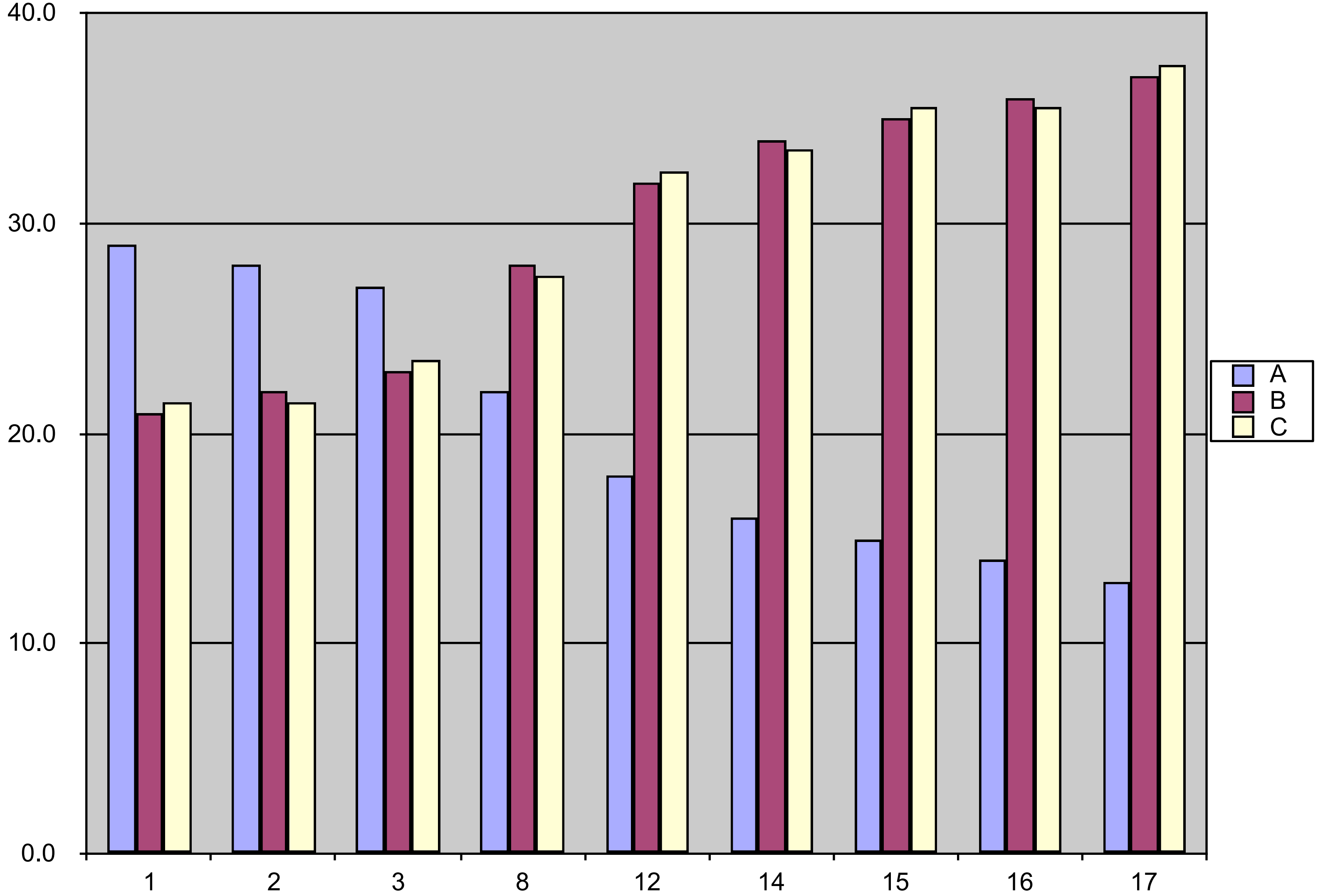
Wolfgang Huber

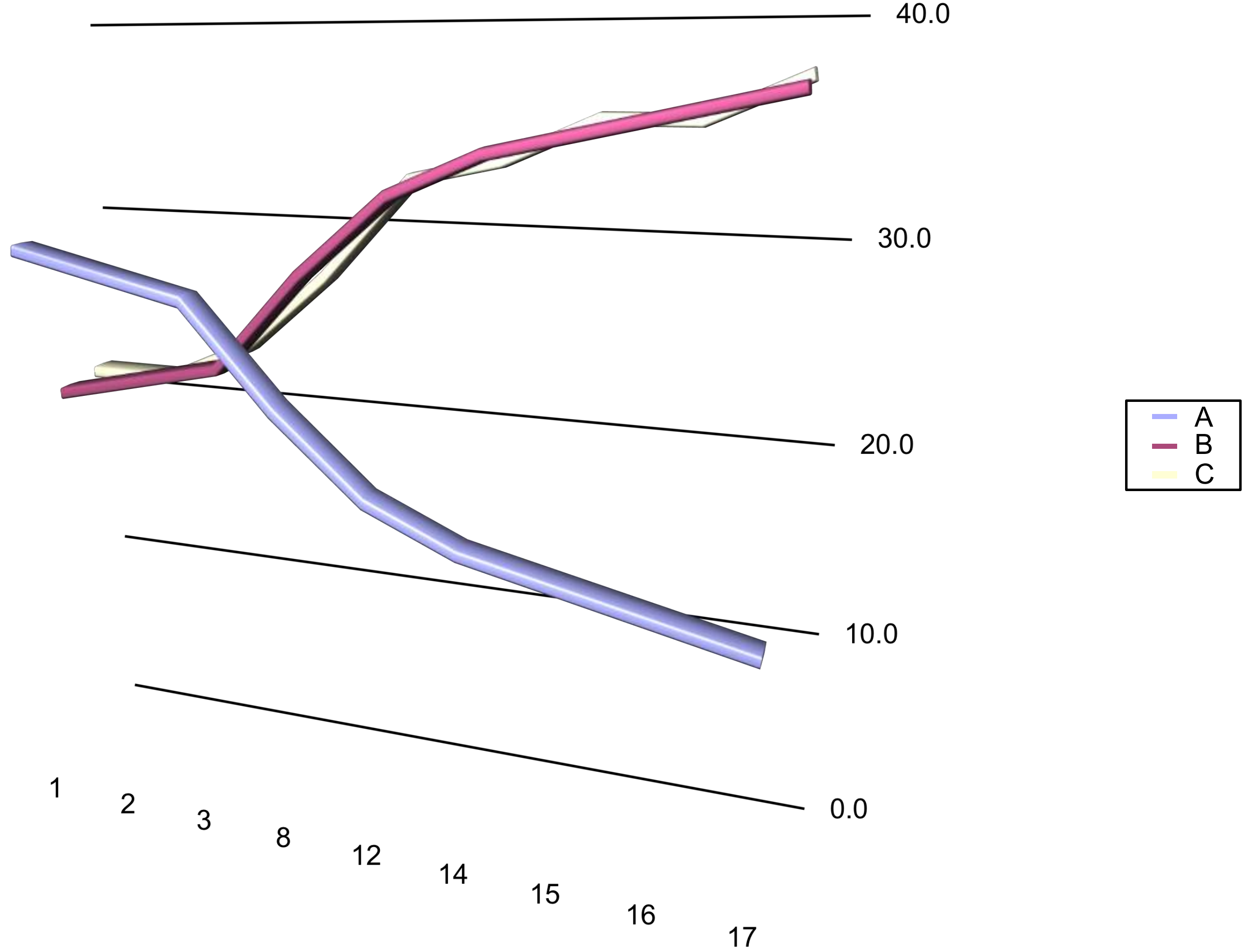
Horror Picture Show

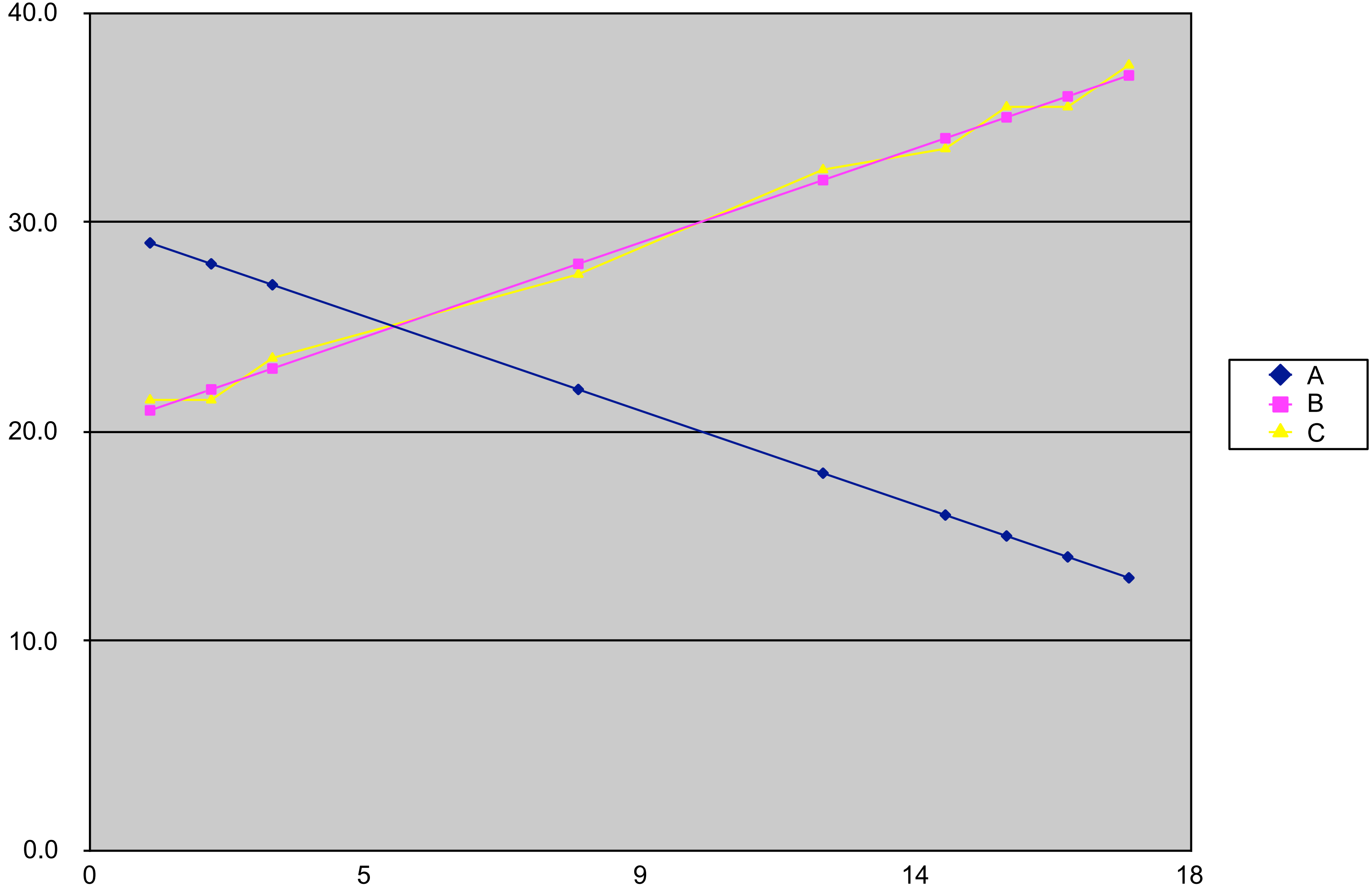

40.0
30.0
20.0
10.0
0.0
1
2
3
8
12
14
15
16
17
A
B
C



40.0
30.0
20.0
10.0
0.0
1
2
3
8
12
14
15
16
17
A
B
C


40.0
30.0
20.0
10.0
0.0
1
2
3
8
12
14
15
16
17
A
B
C


40.0
30.0
20.0
10.0
0.0
1
2
3
8
12
14
15
16
17
A
B
C


40.0
30.0
20.0
10.0
0.0
0
5
9
14
18
A
B
C

Why graphics?

1. To explore data (interactively)
2. To communicate data & preliminary insights with collaborators
3. To publish results

Goals for this lecture

- Review base R plotting
- Understand the **grammar of graphics** concept
- Introduce ggplot2's ggplot function
- See how to plot 1D, 2D, 3-5D data and understand faceting
- Visualisation for quickling viewing large datasets and discover large-scale trends (e.g. batch effects)
- Use colours like a pro
- PCA

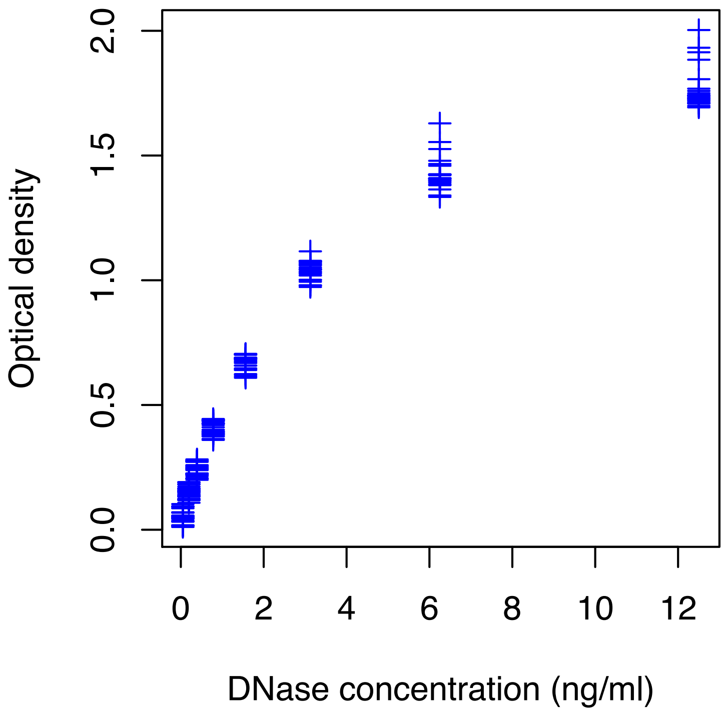
base R plotting

Canvas model: a series of instructions that sequentially fill the plotting canvas

```r
head(DNase)
```

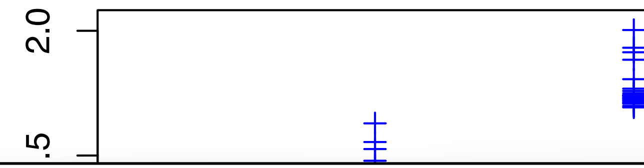
```
##   Run   conc density
## 1   1 0.0488   0.017
## 2   1 0.0488   0.018
## 3   1 0.1953   0.121
## 4   1 0.1953   0.124
## 5   1 0.3906   0.206
## 6   1 0.3906   0.215
```

```r
plot(DNase$conc, DNase$density,
ylab = attr(DNase, "labels")$y,
xlab = paste(attr(DNase, "labels")$x, attr(DNase, "units")$x),
pch = 3, col = "blue")
```



base R plotting

Canvas model: a series of instructions that sequentially fill the pl




```
head(DNase

##   Run
## 1   1 0
## 2   1 0
## 3   1 0
## 4   1 0
## 5   1 0
## 6   1 0
```

```
plot(DNase
ylab = att
xlab = paste(attr(DNase, "labels")$x, attr(DNase, "units")$x),
pch = 3, col = "blue")
```

Drawbacks:

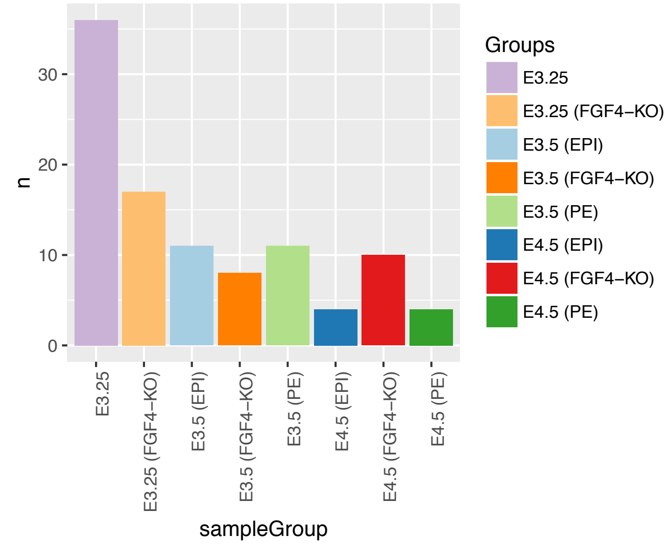
- Layout choices have to be made with no ‘global’ overview over what may still be coming
- Resizing often leads to unsatisfactory results
- Different functions for different plot types with different interfaces
- Many routine tasks require a lot of ‘boilerplate’ code
- No concept of facets / lattices / viewports
- Default colours are poor

The grammar of graphics

The components of *ggplot2*'s grammar of graphics are

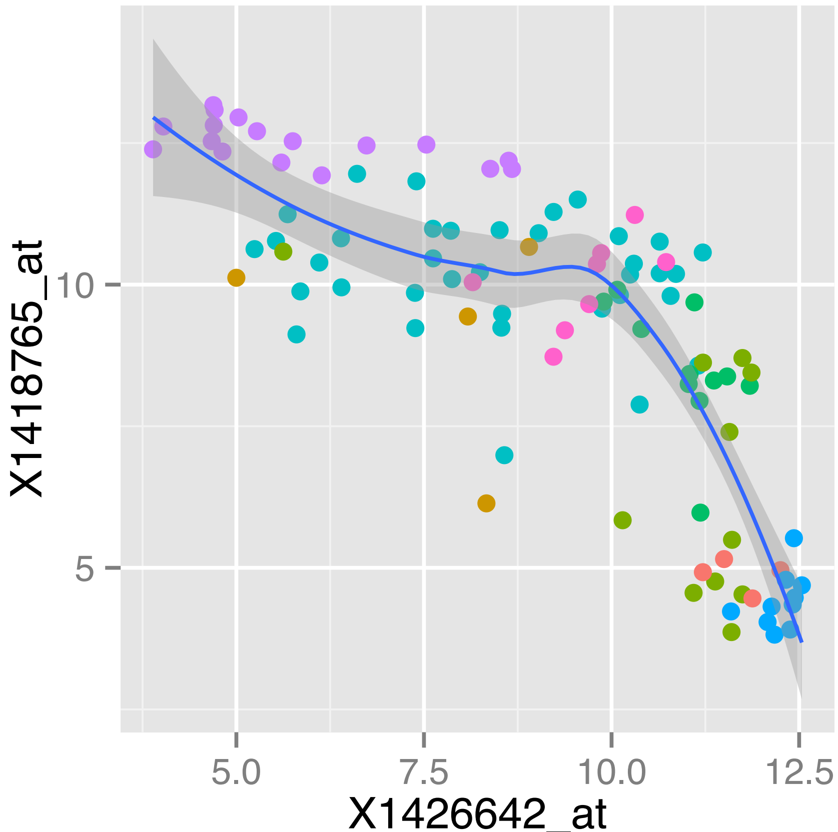
1. a dataset
2. a choice of geometric object that serves as the visual representations of the data – for instance, points, lines, rectangles, contours
3. a description of how the variables in the data are mapped to visual properties (aesthetics) of the geometric objects, and an associated scale, (e. g., linear, logarithmic, rank)
4. a statistical summarisation rule
5. a coordinate system
6. a facet specification, i. e. the use of several plots to look at the same data

```r
ggplot(groups, aes(x = sampleGroup, y = n, fill = sampleGroup)) +
  geom_bar(stat = "identity") +
  scale_fill_manual(values = groupColour, name = "Groups") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



Layers

```
ggplot( dftx, aes( x = X1426642_at, y = X1418765_at ))  +
  geom_point( aes( colour = sampleColour), shape = 19 ) +
  geom_smooth( method = "loess" ) +
  scale_colour_discrete( guide = FALSE )
```

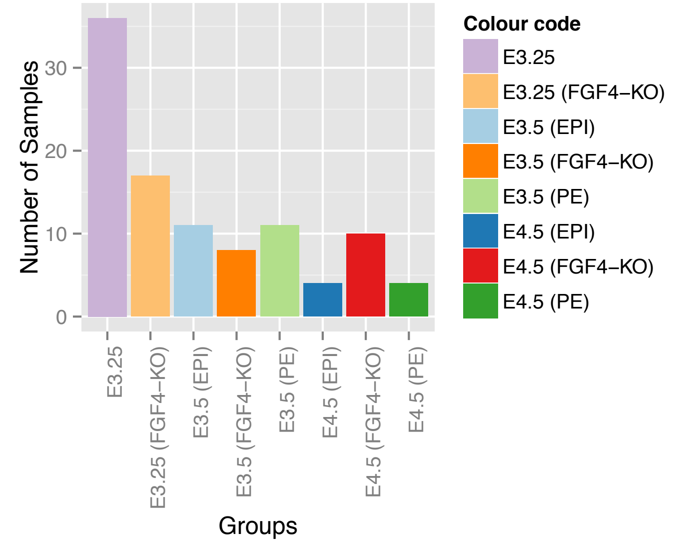


A more complex example: themes

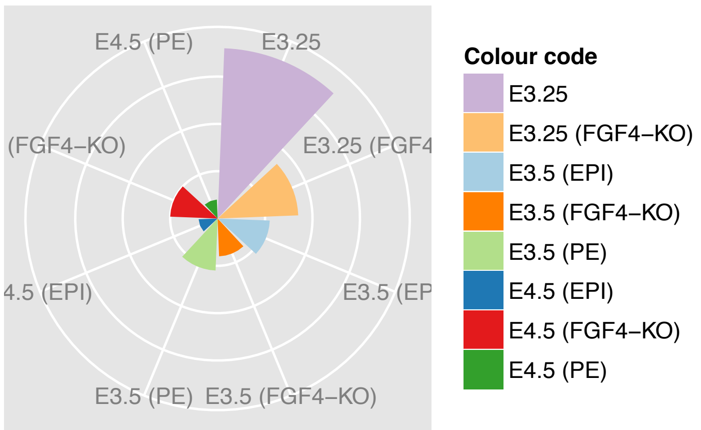
```
pb <- ggplot(data.frame(
                  name = names(groupSize),
                  size = as.vector(groupSize)),
              aes(x = name, y = size))
```

No geom defined yet!

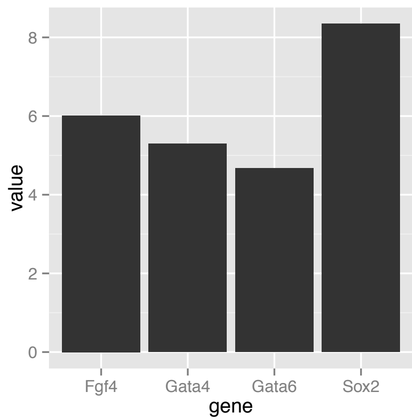
```
pb <- pb + geom_bar(stat = "identity") +
   aes(fill = name) +
   scale_fill_manual(values = groupColour, name = "Colour code") +
   theme(axis.text.x = element_text(angle = 90, hjust = 1))  +
   xlab("Groups") + ylab("Number of Samples")
```

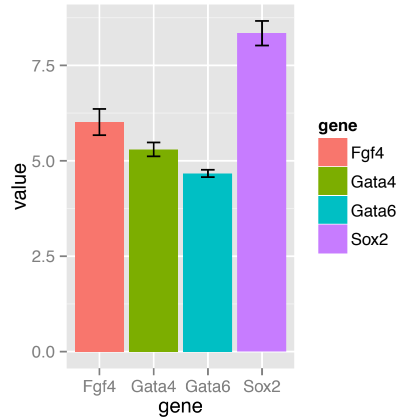


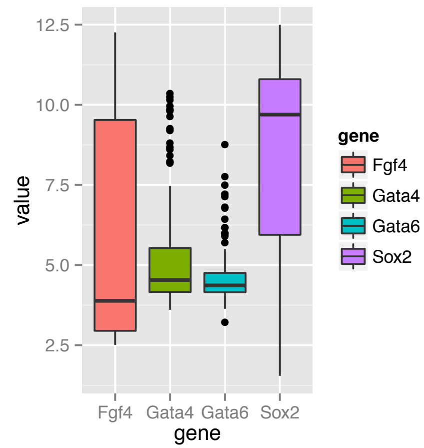

```
pb.polar <- pb + coord_polar() +
  theme(axis.text.x = element_text(angle = 0, hjust = 1),
        axis.text.y = element_blank(),
        axis.ticks = element_blank()) +
  xlab("") + ylab("")
pb.polar
```

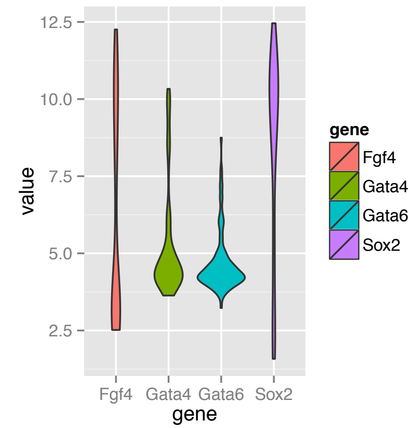


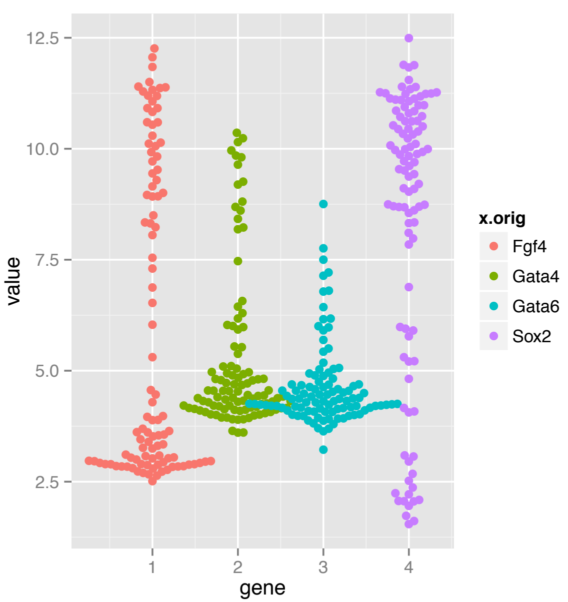
Showing 1D data

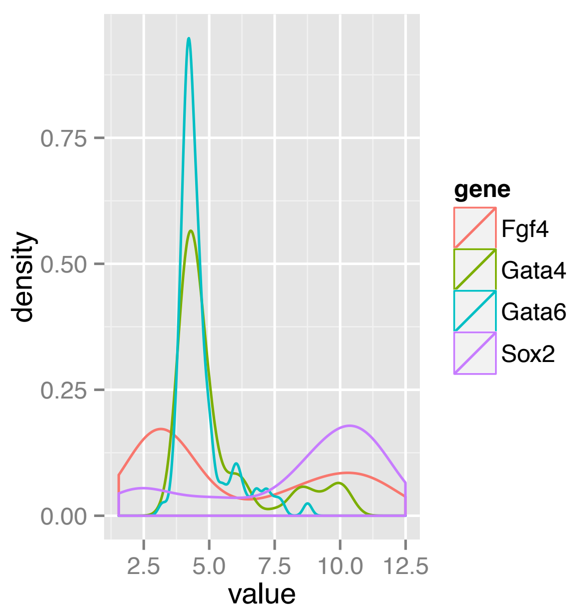


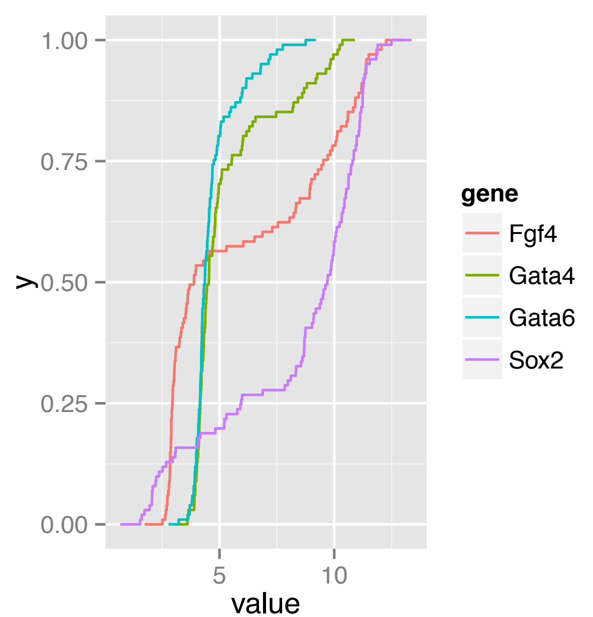












Discussion of 1D plot types

Boxplot makes sense for unimodal distributions

Histogram requires definition of bins (width, positions) and can create visual artifacts esp. if the number of data points is not large

Density requires the choice of bandwidth; plot tends to obscure the sample size (i.e. the uncertainty of the estimate)

ecdf does not have these problems; but is more abstract and interpretation requires some training. Good for reading off quantiles and shifts in location in comparative plots; OK for detecting differences in scale; less good for detecting multimodality.

Up to a few dozens of points - just show the data! (beeswarm)

Impact of non-linear transformation on the shape of a density

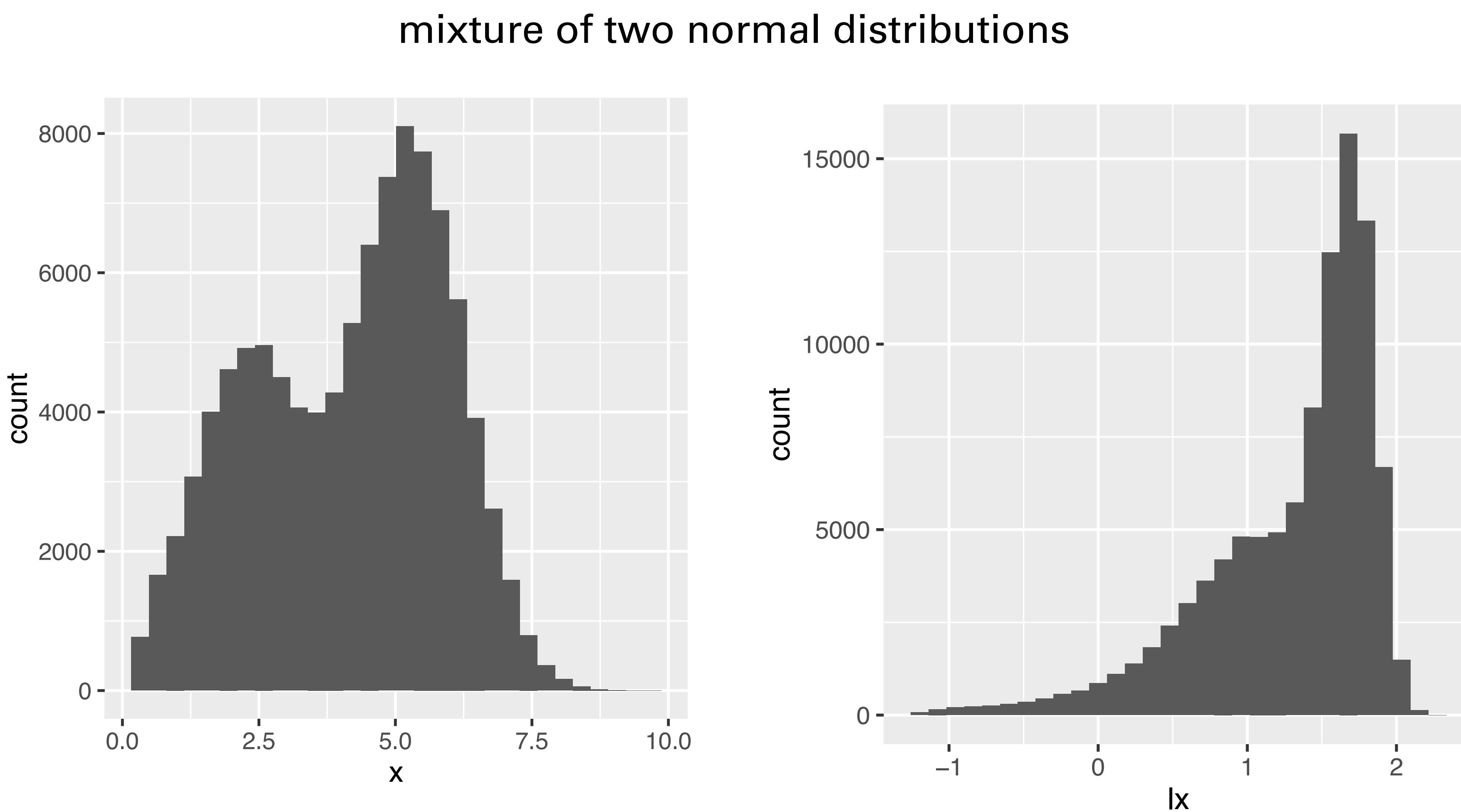
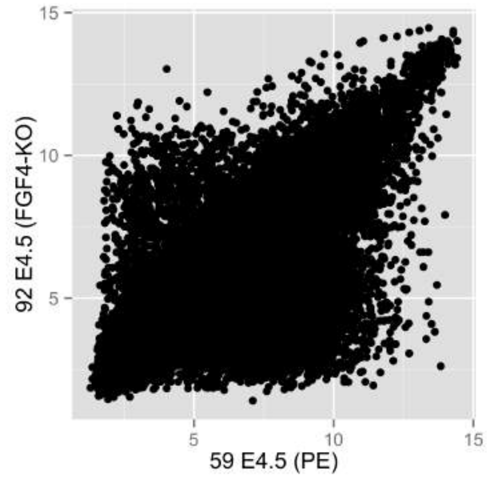
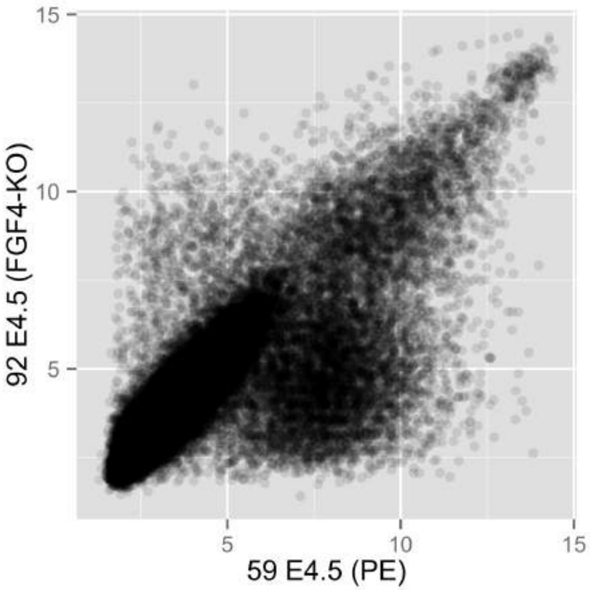


Figure 3.22: Histograms of the same data, with and without logarithmic transformation. The number of modes is different.

Showing 2D data

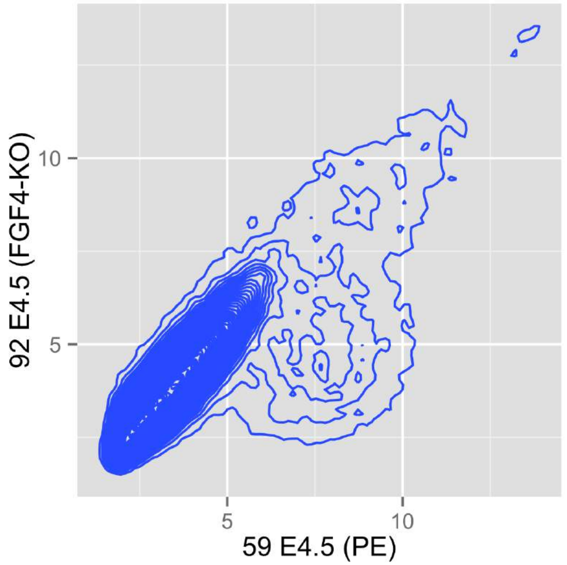
```
scp <- ggplot(dfx, aes( x = ‘59 E4.5 (PE)‘ ,
                        y = ‘92 E4.5 (FGF4-KO)‘))
scp + geom_point()
```



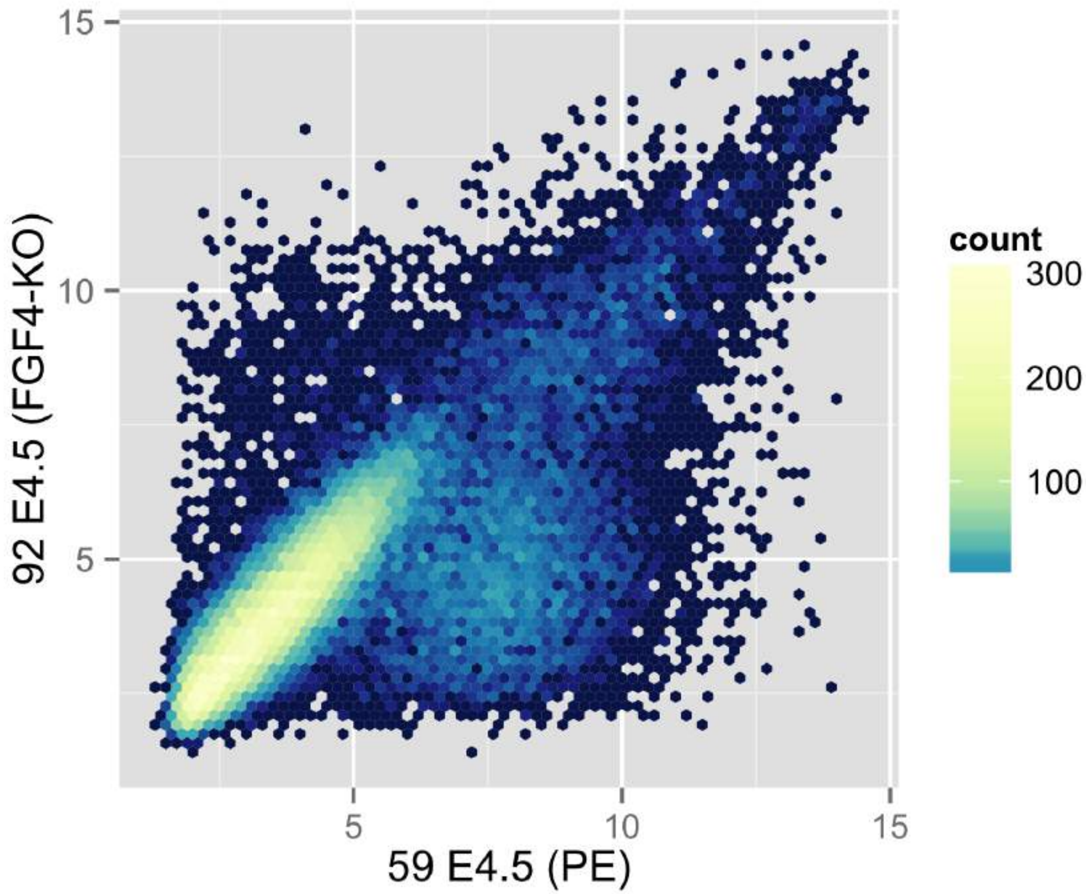



```
scp  + geom_point(alpha = 0.1)
```

```
scp + geom_density2d(h = 0.5, bins = 60)
```



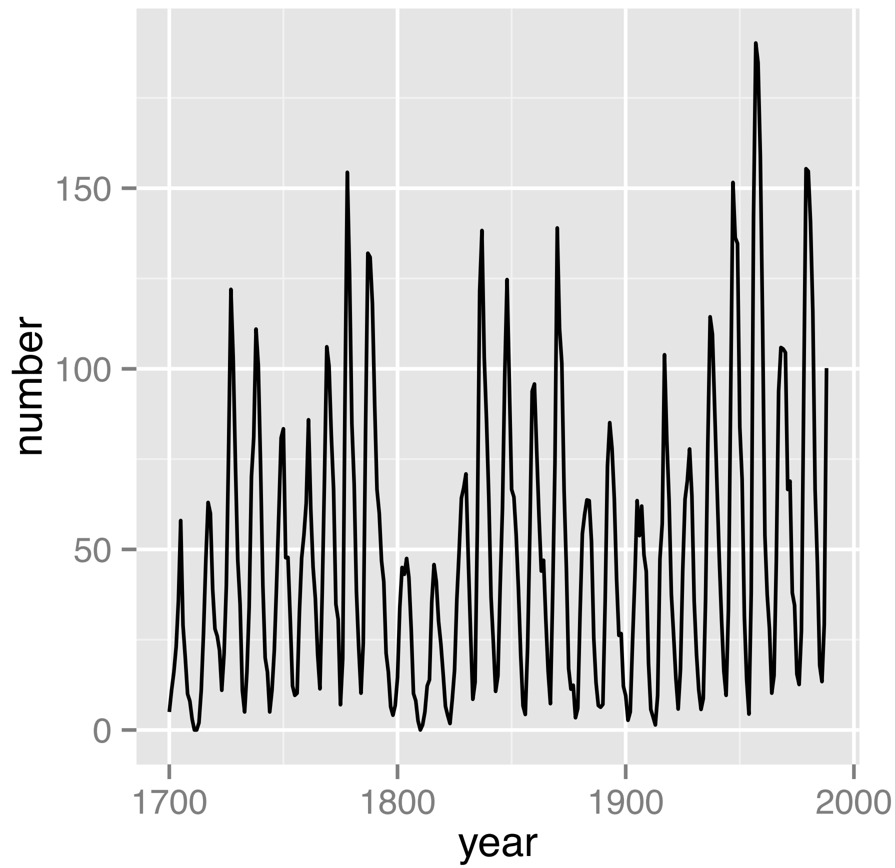
Showing 2D data




```
scp + stat_binhex(binwidth = c(0.2, 0.2)) + colourscale +
  coord_fixed()
```

Yearly sunspot numbers 1849-1924

Plot shape, banking



Changes in amplitude

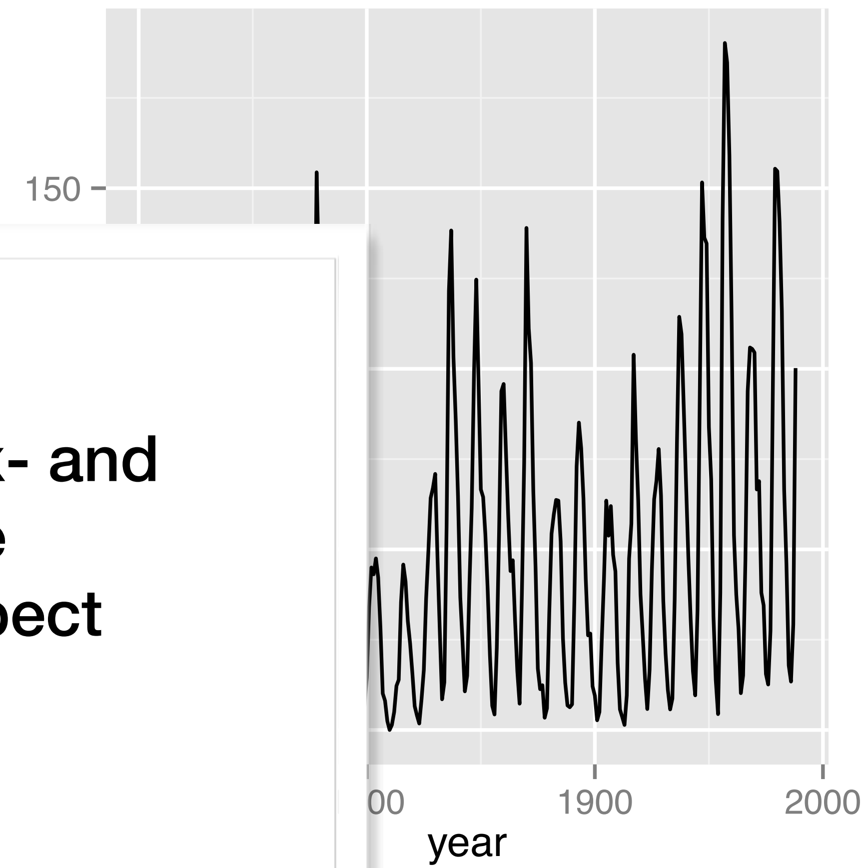
Banking to 45 degrees:

Choose aspect ratio so that center of absolute values of slopes is 45 degrees

Sawtooth: Sunspot cycles typically rise more rapidly than they fall (pronounced for high peaks, less for medium and not for lowest)

Yearly sunspot numbers 1849-1924

Plot shape, banking



Changes in amplitude

Bankin

Choose

center

slopes

Sawtoo

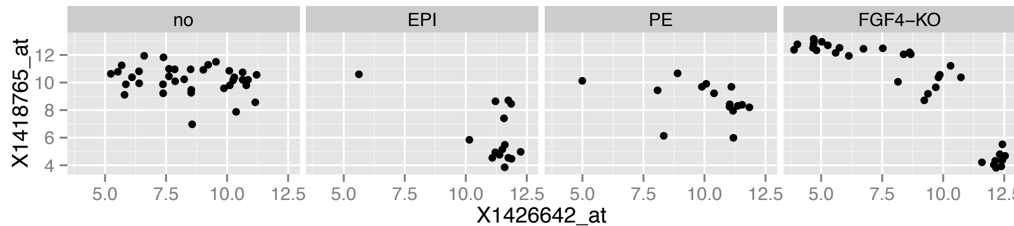
typically

than they fall (pronounced for high peaks, less for medium and not for lowest)

For plots where x- and y-axis have same units: use 1:1 aspect ratio (PCA!)

3-5 D, and faceting

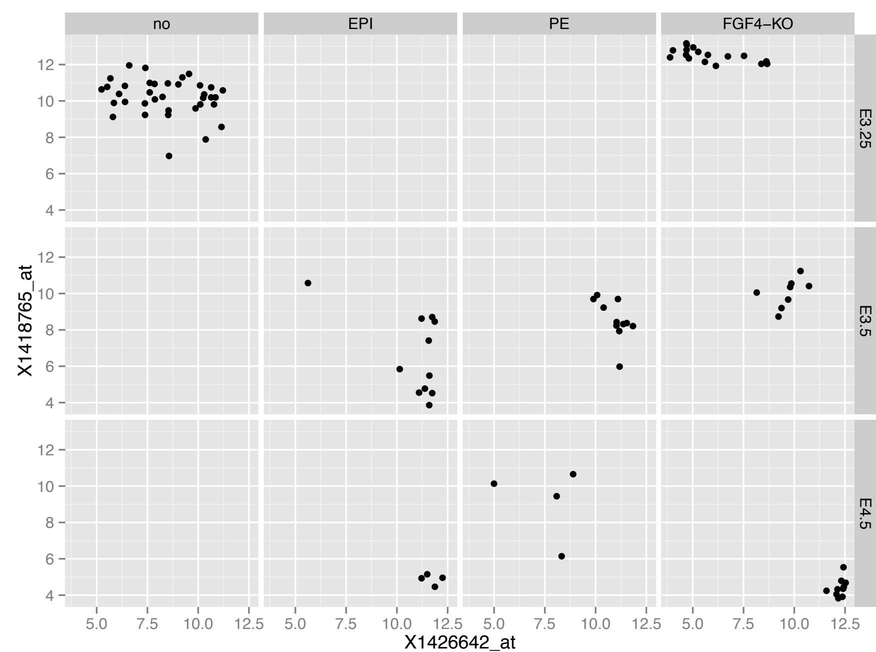
```
ggplot(dftx, aes( x = X1426642_at, y = X1418765_at)) +
  geom_point() + facet_grid( . ~ lineage )
```



geom_point
offers these
aesthetics
(beyond x and y):

- fill
- colour
- shape
- size
- alpha

```
ggplot( dftx,
  aes( x = X1426642_at, y = X1418765_at)) + geom_point() +
   facet_grid( Embryonic.day ~ lineage )
```

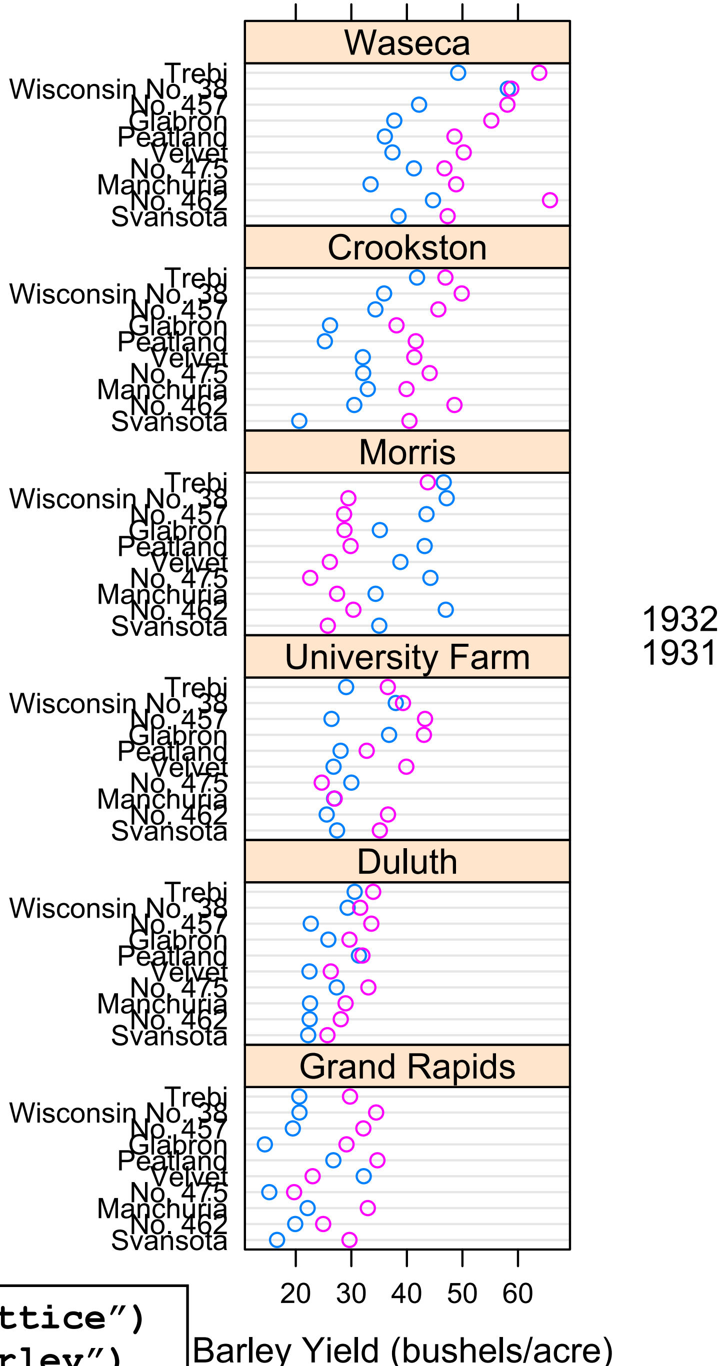


Data from an agricultural field trial to study the crop barley.

At 6 sites in Minnesota, 10 varieties of barley were grown in each of two years.

Data: yield, for all combinations of site, variety, and year (6 x 10 x 2 = 120 observations)

Note the data for Morris - reanalysis in the 1990s using Trellis revealed that the years had been flipped!

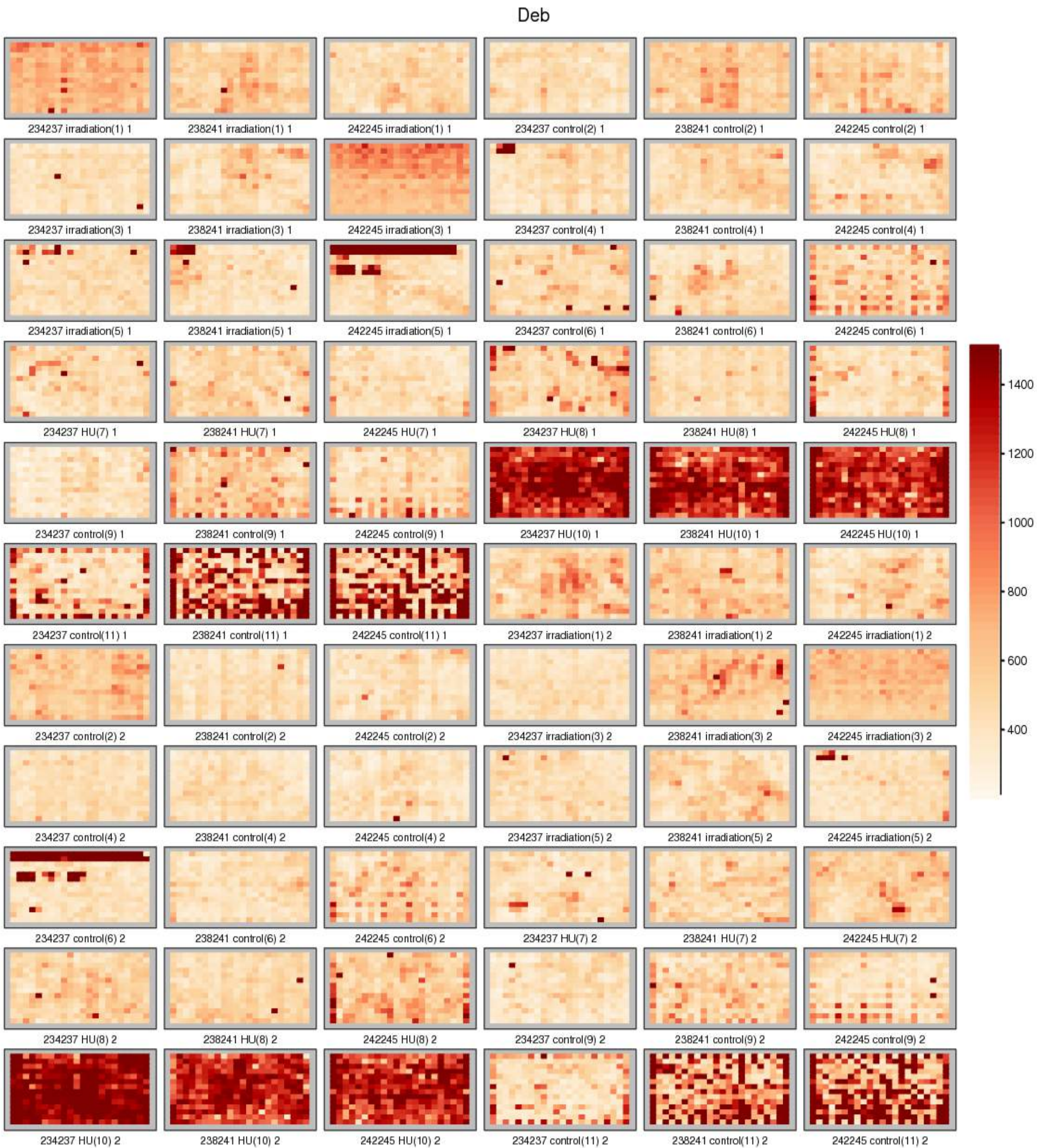



```
library("lattice")
example("barley")
```

Demo ggvis

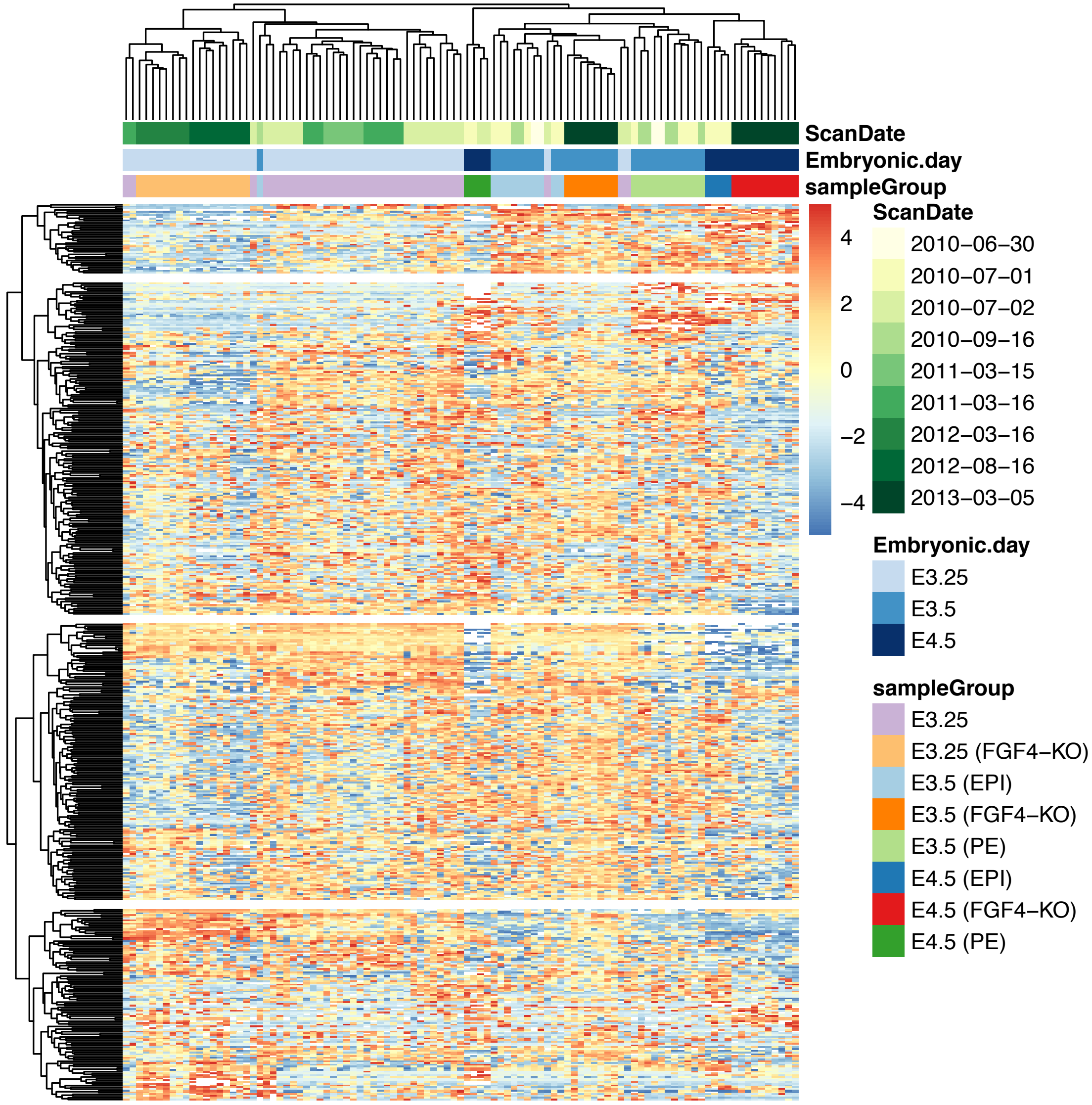
1. in R-Studio
2. http://ggvis.rstudio.com/interactivity.html

EDA for finding batch effects



package
splots

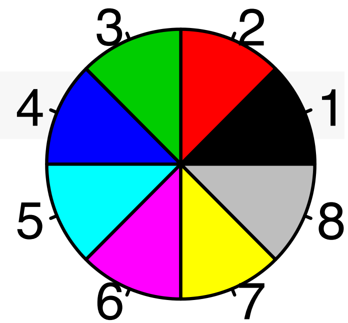
pheatmap



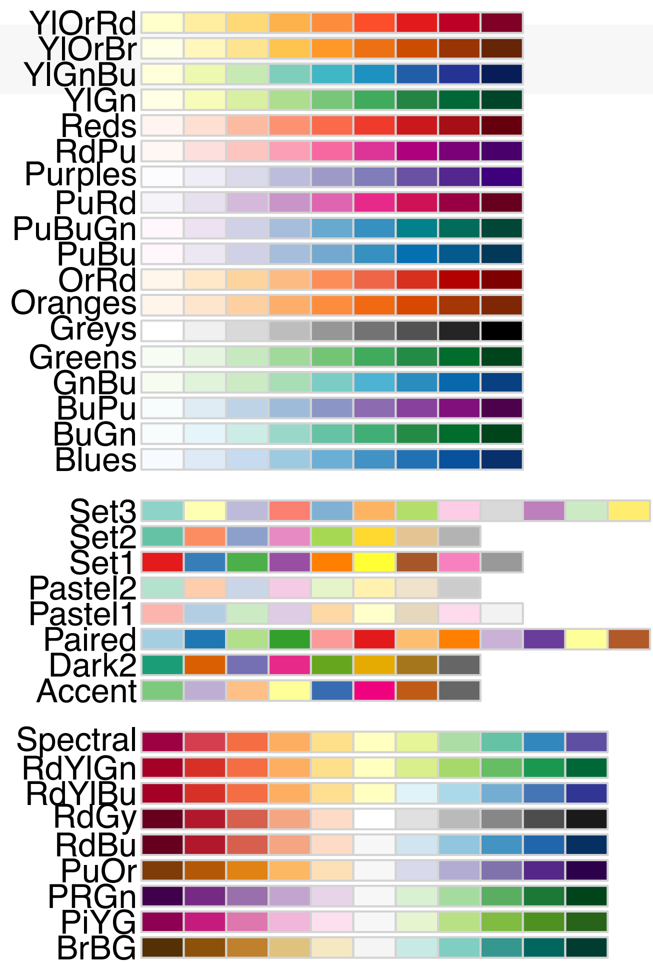
many reasonable defaults

easy to add column and row ‘metadata’ at the sides

```
pie(rep(1, 8), col=1:8)
```




```
display.brewer.all()
```



```
pie(rep(1, 8), c
```

```
display.brewer.
```

Consider these:

Different requirements for line & area colours

Many people are red-green colour blind

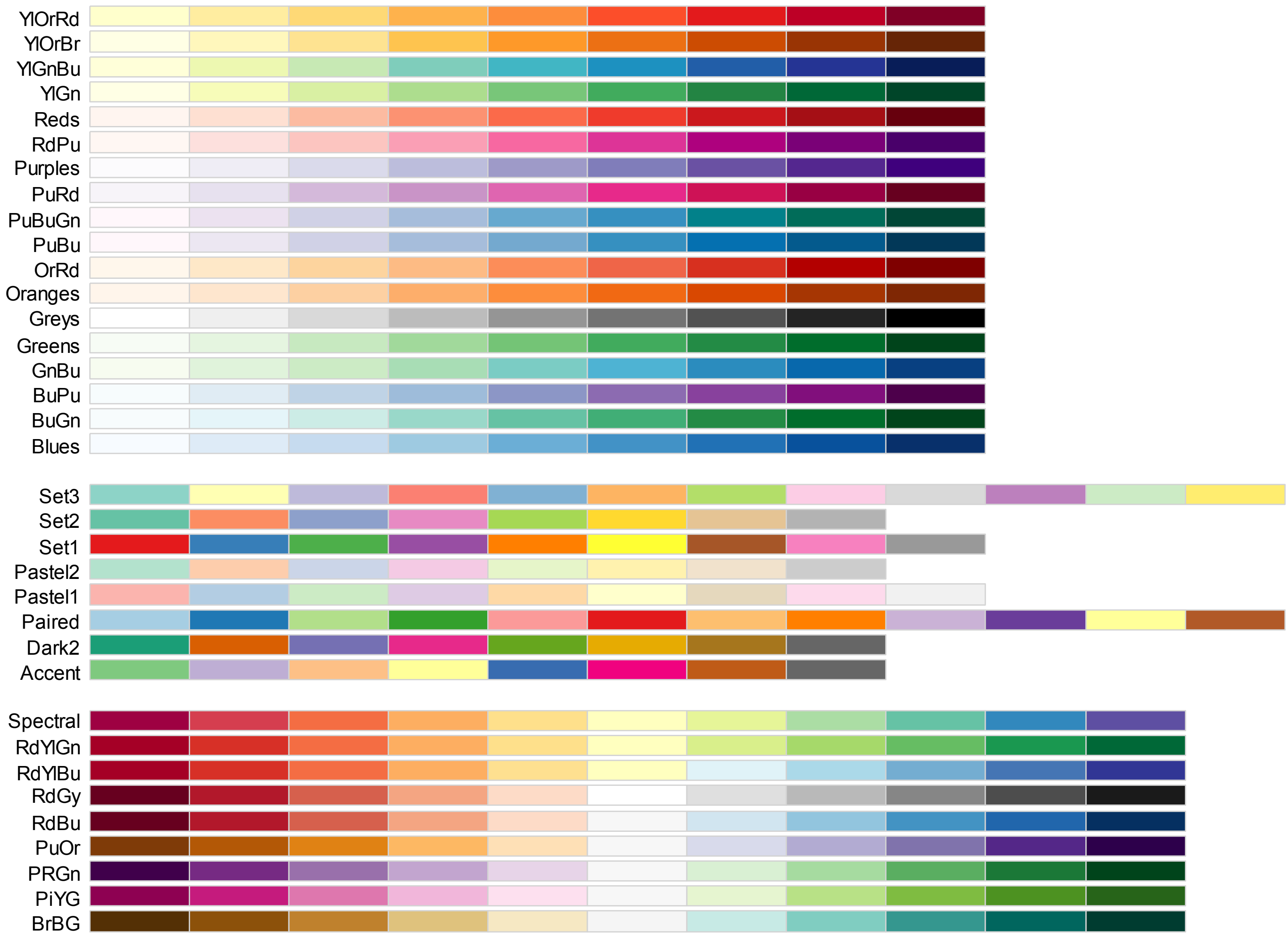
Lighter colours tend to make areas look larger than darker colours -> use colors of equal luminance for filled areas.

RColorBrewer

sequential

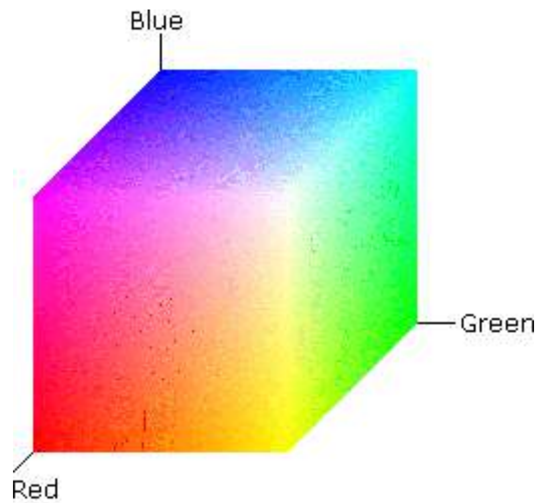
qualitative

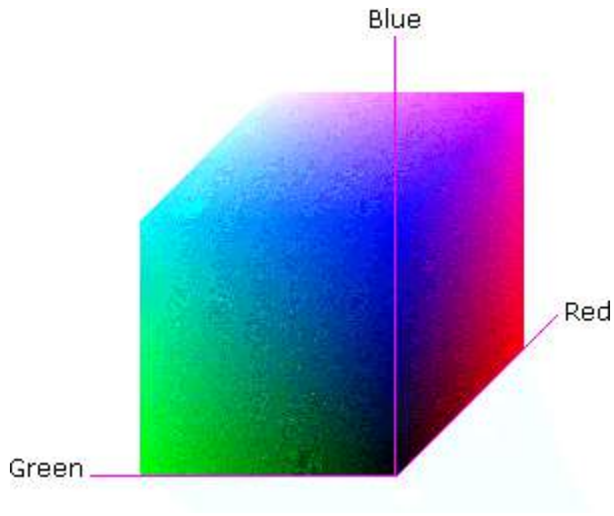
diverging



RGB color space

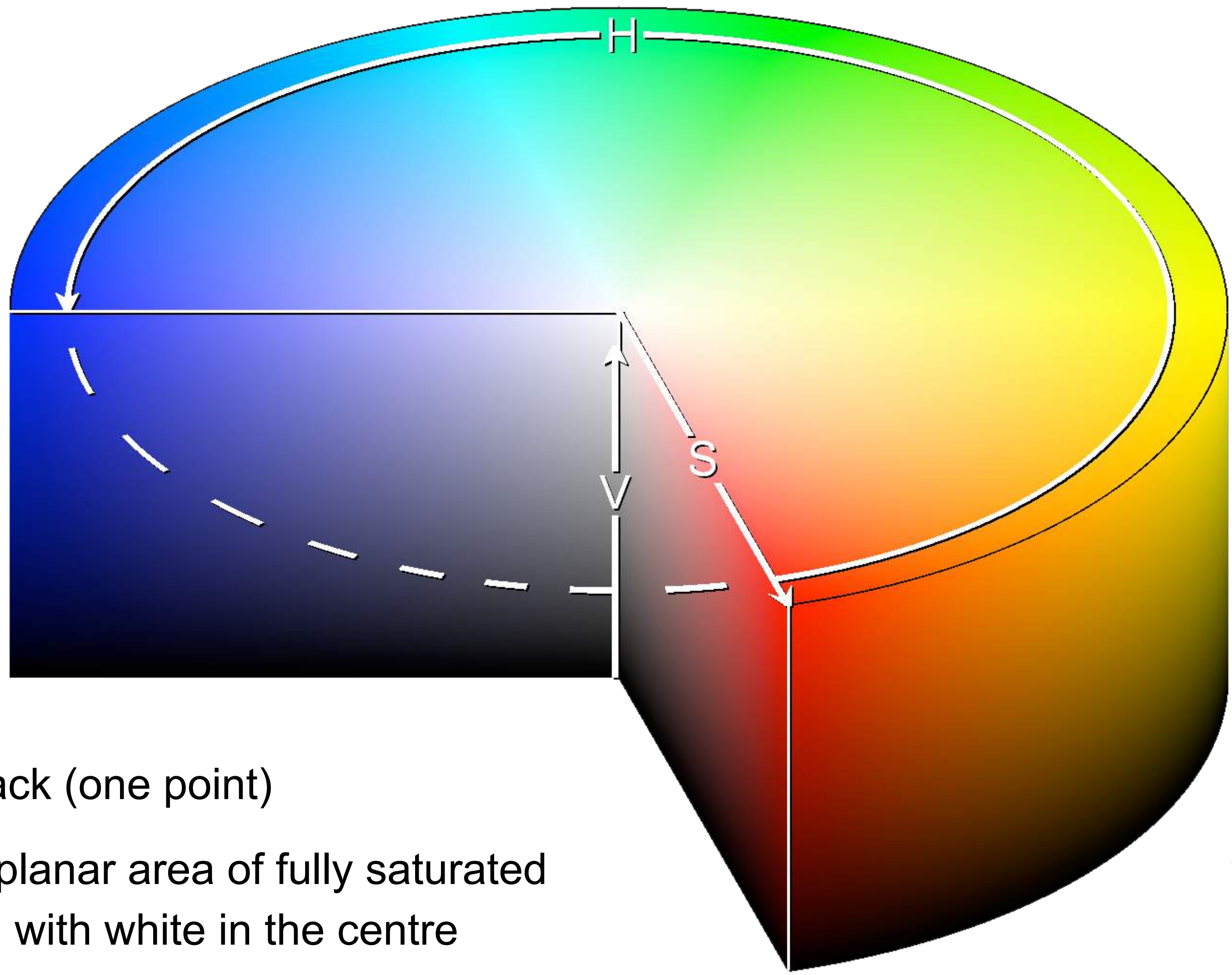
Motivated by computer screen hardware





HSV color space

Hue-Saturation-Value (Smith 1978)



V_{min}: black (one point)

V_{max}: a planar area of fully saturated colours, with white in the centre

wikipedia

HSV color space

GIMP colour selector

linear or circular hue chooser

and

a two-dimensional area (usually a square or a triangle) to choose saturation and value/lightness for the selected hue

(almost) 1:1 mapping between RGB and HSV space

Conversion from RGB to HSL or HSV

Let $r, g, b \in [0,1]$ be the red, green, and blue coordinates, respectively, of a color in RGB space.

Let max be the greatest of r, g, and b, and min the least.

To find the hue angle $h \in [0, 360]$ for either HSL or HSV space, compute:

$$h = \begin{cases} 0 & \text{if } \max = \min \\ (60^\circ \times \frac{g-b}{\max - \min} + 0^\circ) \bmod 360^\circ, & \text{if } \max = r \\ 60^\circ \times \frac{b-r}{\max - \min} + 120^\circ, & \text{if } \max = g \\ 60^\circ \times \frac{r-g}{\max - \min} + 240^\circ, & \text{if } \max = b \end{cases}$$

To find saturation and lightness $s, l \in [0,1]$ for HSL space, compute:

$$s = \begin{cases} 0 & \text{if } \max = \min \\ \frac{\max - \min}{\max + \min} = \frac{\max - \min}{2l}, & \text{if } l \leq \frac{1}{2} \\ \frac{\max - \min}{2 - (\max + \min)} = \frac{\max - \min}{2 - 2l}, & \text{if } l > \frac{1}{2} \end{cases}$$

$$l = \tfrac{1}{2}(\max + \min)$$

wikipedia

The value of h is generally normalized to lie between 0 and 360°, and $h = 0$ is used when $max = min$ (that is, for grays) though the hue has no geometric meaning there, where the saturation s is zero. Similarly, the choice of 0 as the value for s when l is equal to 0 or 1 is arbitrary.

HSL and HSV have the same definition of hue, but the other components differ. The values for s and v of an HSV color are defined as follows:

$$s = \begin{cases} 0, & \text{if } \max = 0 \\ \frac{\max - \min}{\max} = 1 - \frac{\min}{\max}, & \text{otherwise} \end{cases}$$

$$v = \max$$

The range of HSV and HSL vectors is a cube in the cartesian coordinate system; but since hue is really a cyclic property, with a cut at red, visualizations of these spaces invariably involve hue circles;[4] cylindrical and conical (bi-conical for HSL) depictions are most popular; Spherical depictions are also possible.

perceptual colour spaces

Human perception of colour corresponds neither to RGB nor HSV coordinates, and neither to the physiological axes light-dark, yellow-blue, red-green

Rather to polar coordinates in the colour plane (yellow/blue vs. green/red) plus a third light/dark axis. Perceptually-based colour spaces try to capture these perceptual axes:

1. hue (dominant wavelength)
2. chroma (colourfulness, intensity of coulor as compared to grey)
3. luminance (brightness, amount of grey)

CIELUV and HCL

Commission Internationale de l' Éclairage (CIE) in 1931, on the basis of extensive colour matching experiments with people, defined a "standard observer" who represents a typical human colour response (response of the three light cones + their processing in the brain) to a triplet (x,y,z) of primary light sources (in principle, this could be monochromatic R, G, B; but CIE choose something a bit more subtle)

1976: CIELUV and CIELAB are perceptually based coordinates of colour space.

CIELUV (L, u, v)-coordinates is prefered by those who work with emissive colour technologies (such as computer displays) and CIELAB by those working with dyes and pigments (such as in the printing and textile industries)

Ihaka 2003

HCL colours

(u,v) = chroma * (cos h, sin h)

L the same as in CIELUV, (C, H) are simply polar coordinates for (u,v)

1. hue (dominant wavelength)
2. chroma (colorfulness, intensity of color as compared to gray)
3. luminance (brightness, amount of gray)

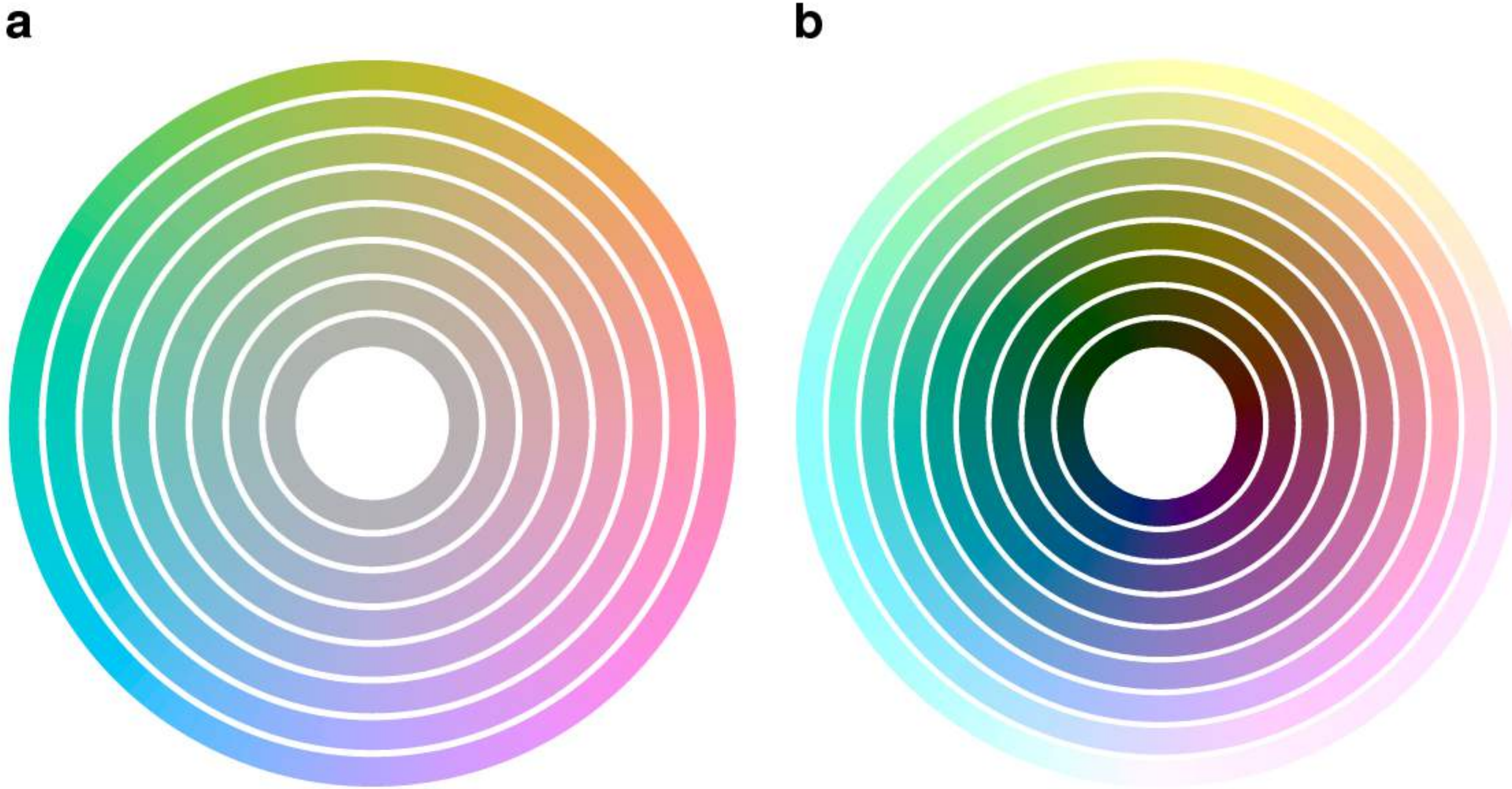
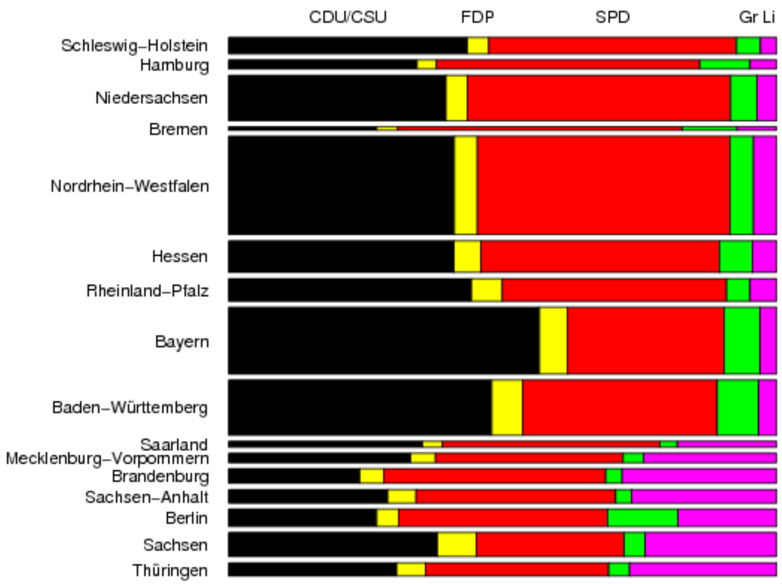
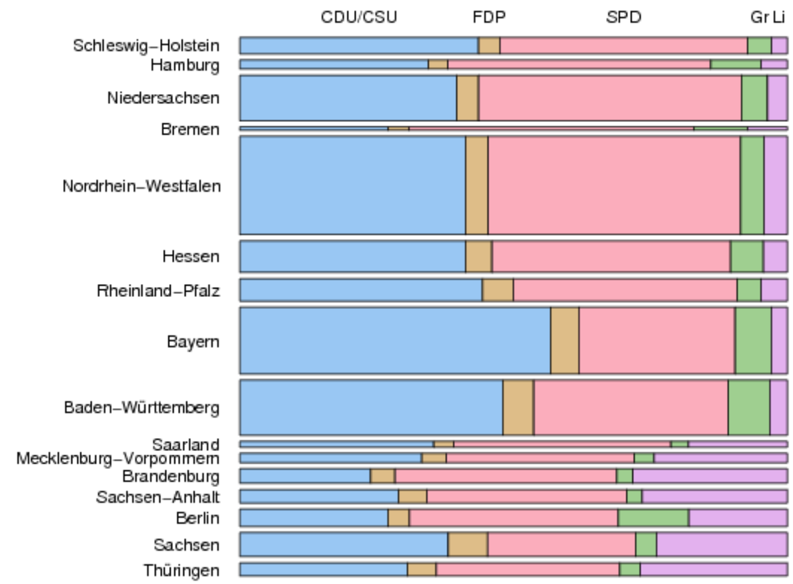


Figure 2: Circles in HCL colorspace. a: circles in HCL space at constant $L = 75$, with the angular coordinate H varying from 0 to 360 and the radial coordinate $C = 0, 10, \ldots, 60$. b: constant $C = 50$, and $L = 10, 20, \ldots, 90$.

Pick your favourite



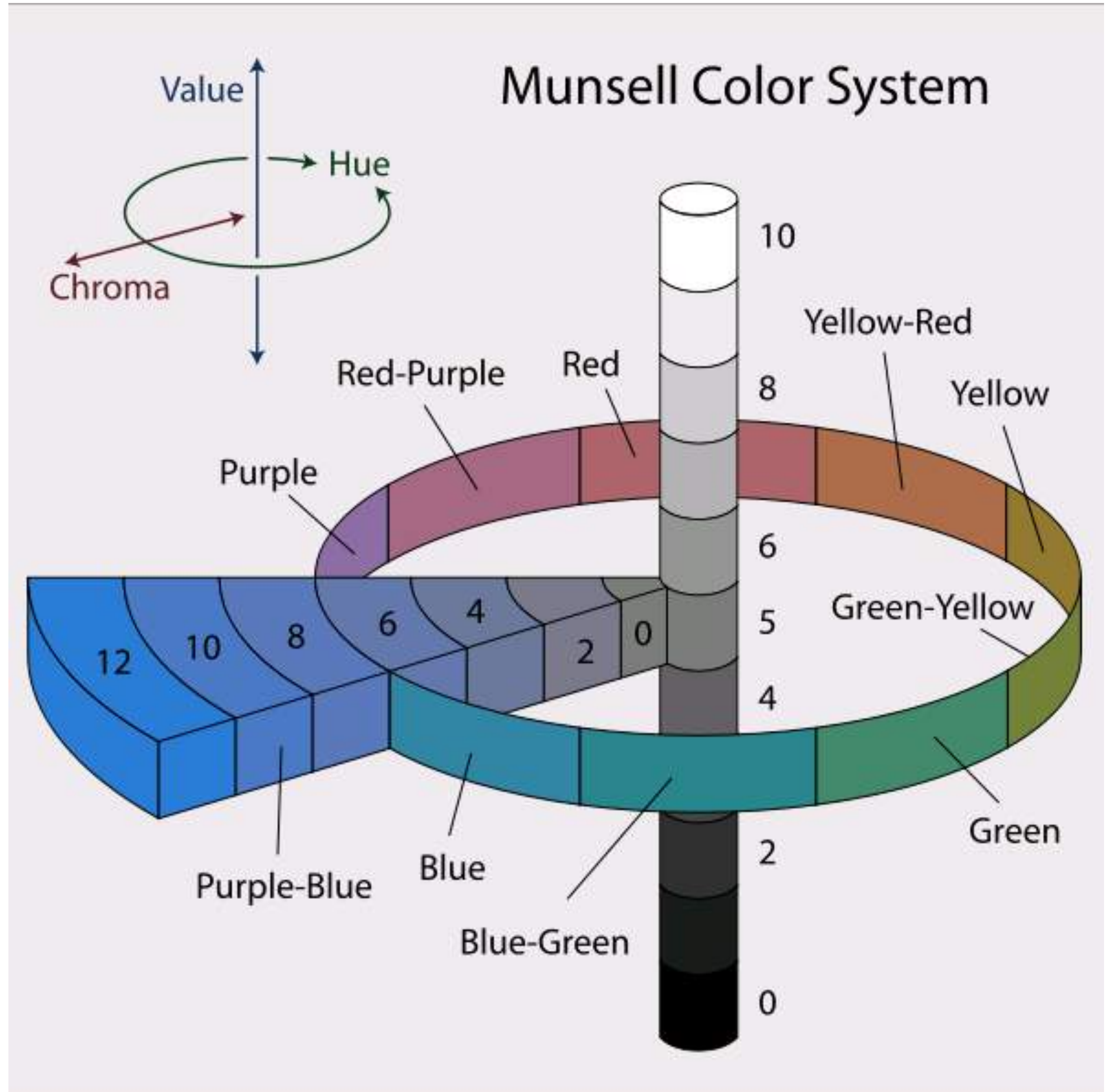


From A. Zeileis, Reisensburg 2007

Albert Munsell (1858-1918) divided the circle of hues into 5 main hues — R, Y, G, B, P (red, yellow, green, blue and purple).

Value, Chroma: ranges divided into 10 equal steps.

E.g. R 4/5 = hue of red with a value of 4 and a chroma of 5.

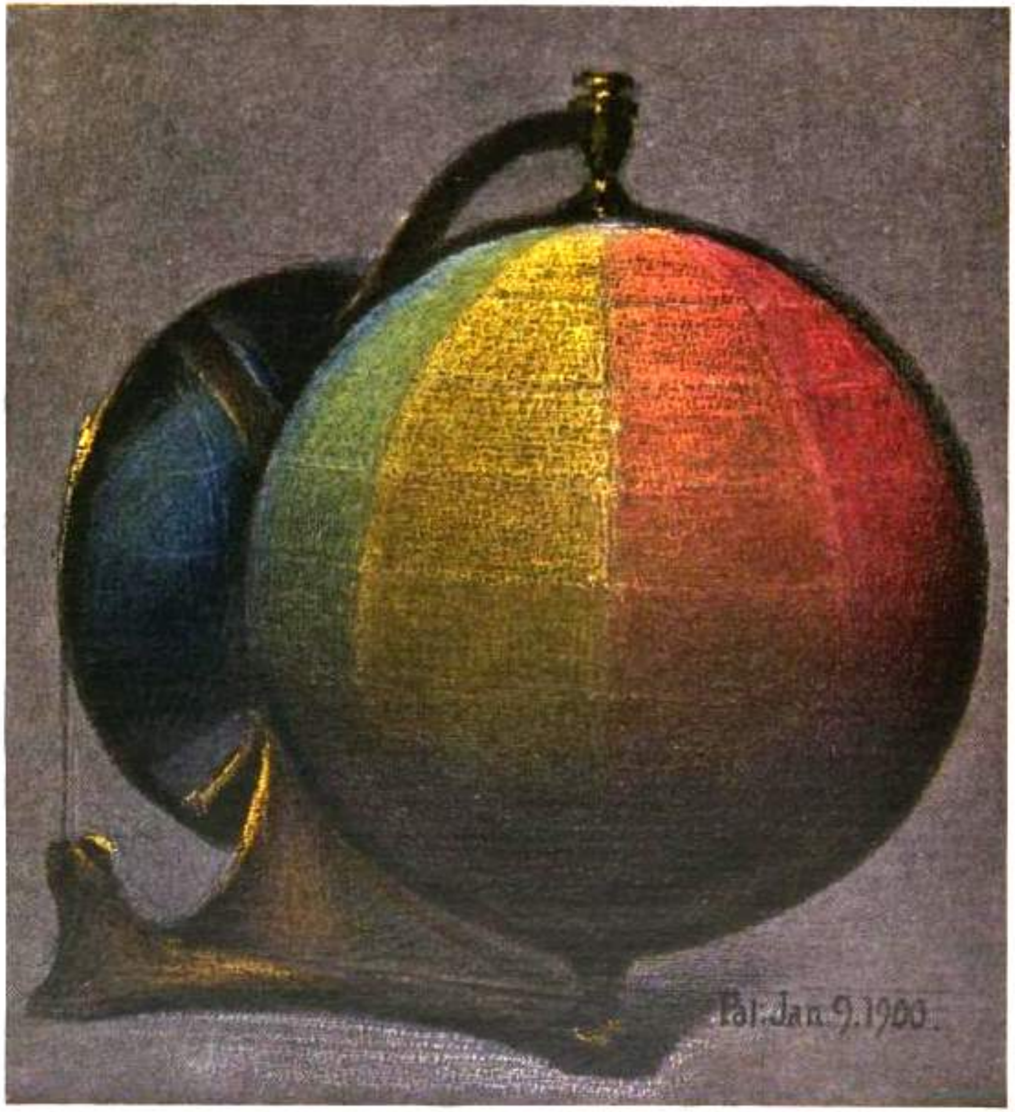


Munsell Colour System

Albert Munsell (1858-1918) divided the circle of hues into 5 main hues — R, Y, G, B, P (red, yellow, green, blue and purple).

Value, Chroma: ranges divided into 10 equal steps.

E.g. R 4/5 = hue of red with a value of 4 and a chroma of 5.



A BALANCED COLOR SPHERE

Colour Harmony

Figure 3: The principal Munsell 5/5 colours. From the top these are R 5/5, Y 5/5, G 5/5, B 5/5 and P 5/5. This figure is redrawn from Birren (1969).

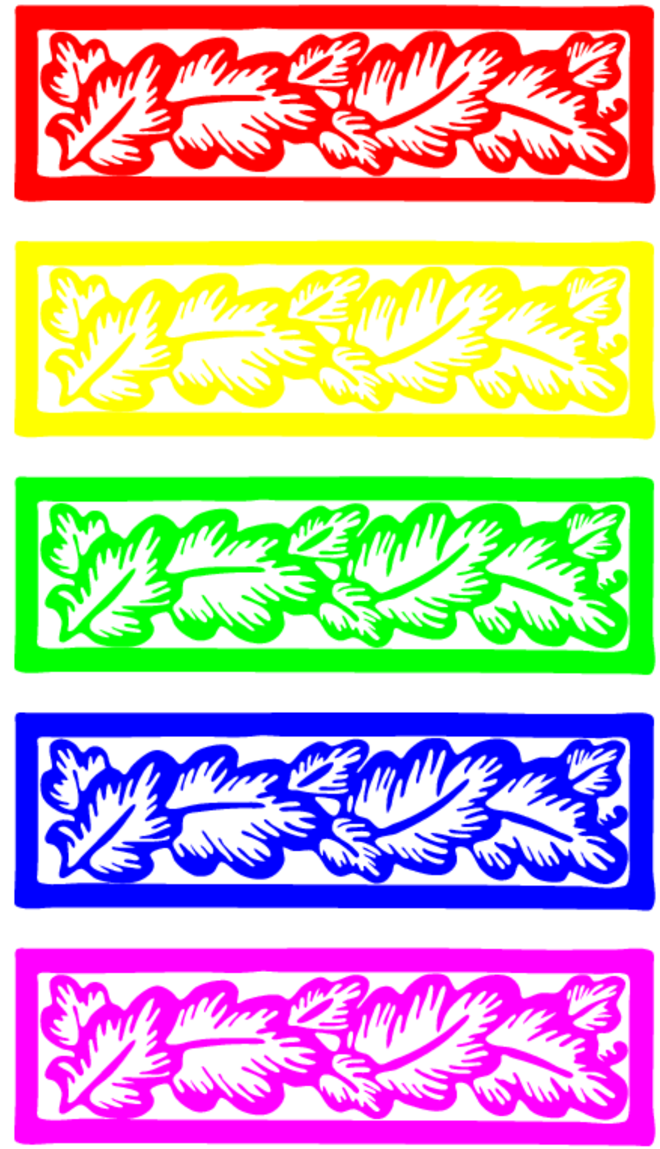


Figure 4: The same images as Figure 3, but drawn with full saturation HSV colours.

Balance

The intensity of colour which should be used is dependent on the area that that colour is to occupy. Small areas need to be more colourful than larger ones.

Choose colours centered on a mid-range or neutral value, or;

Choose colours at equally spaced points along smooth paths through (perceptually uniform) colour space: equal luminance and chroma and correspond to set of evenly spaced hues.

Acknowledgements

Susan Holmes

Robert Gentleman

Florian Hahne

Hadley Wickham

Ross Ihaka

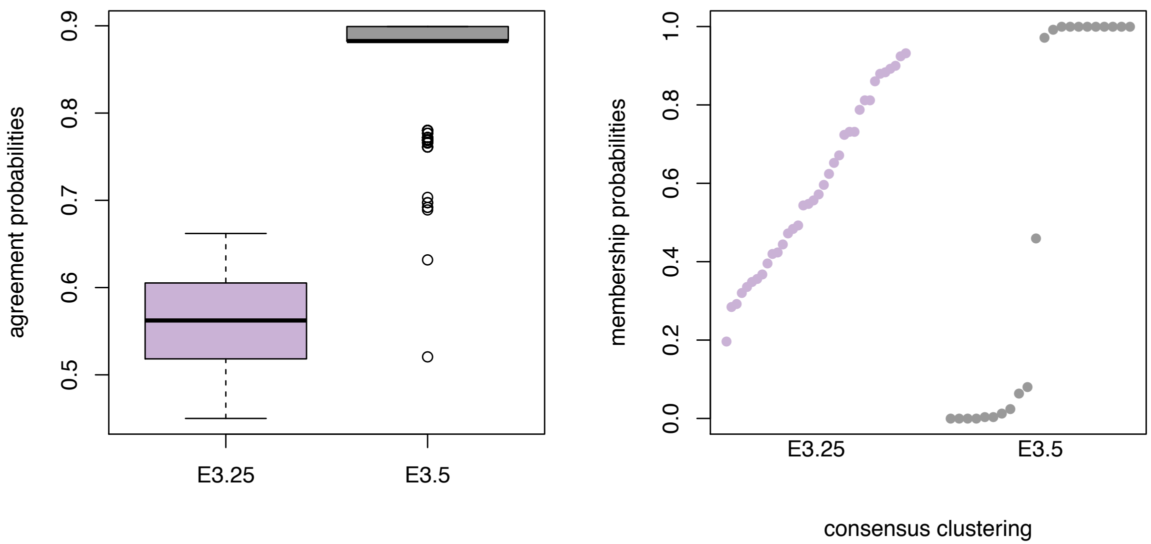
Achim Zeileis

Kurt Hornik

Cluster stability analysis

1. Draw random subset of the full data (e.g. 67% of the samples)
2. Apply clustering method of choice. Predict cluster memberships of the samples not in the subset with cl_predict - through their proximity to the cluster centres
3. Repeat 1.+2. for B = 250 times
4. Apply consensus clustering (cl_consensus)
5. For each of the B clusterings, measure agreement with consensus (cl_agreement)
6. If the agreement is generally high, then the clustering into k classes can be considered stable and reproducible; inversely, if it is low, then no stable partition of the samples into k clusters is evident

Cluster stability analysis



Ohnishi et al.,
Nature Cell Biology
doi: 10.1038/ncb2881

Bioc package:
Hiiragi2013

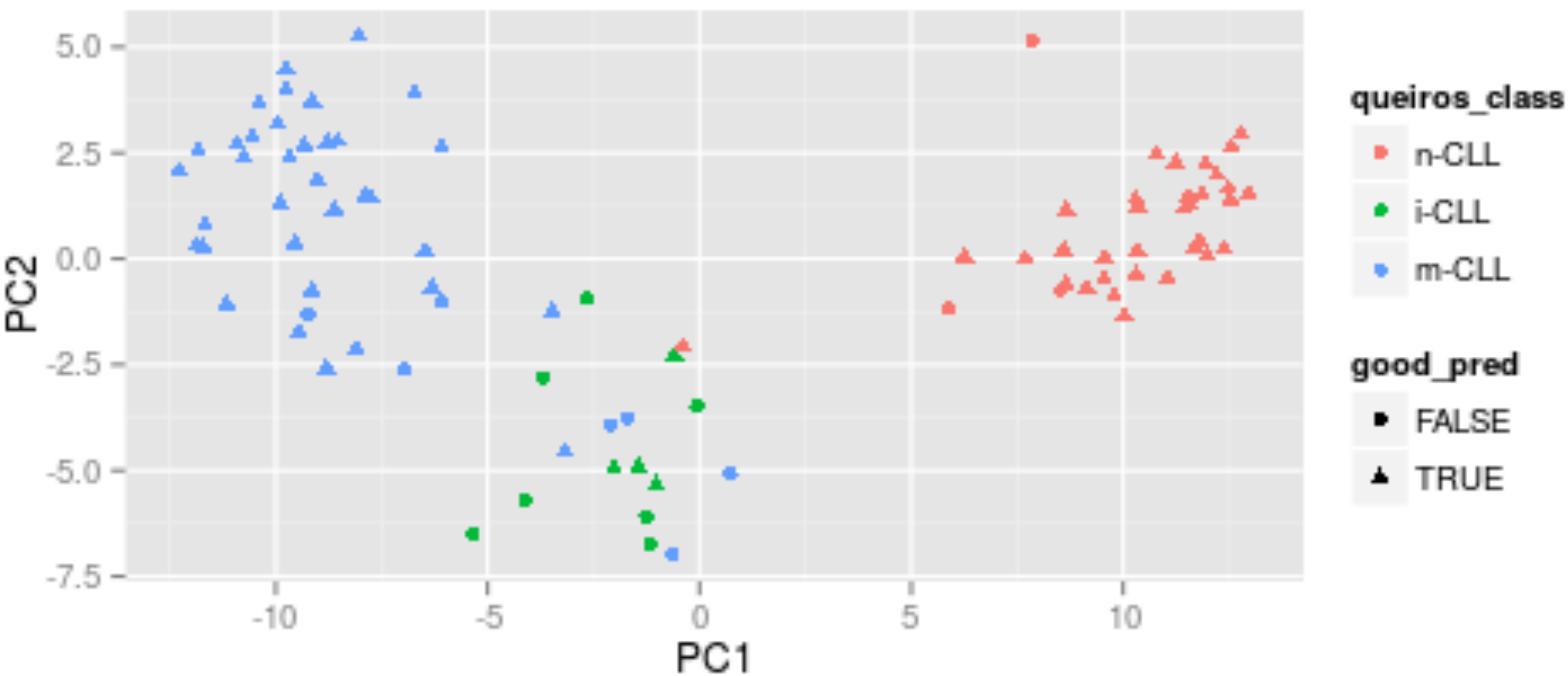
Figure 3: Cluster stability analysis with E3. 25 and E3.5 WT samples. Left: boxplot of the cluster agreements with the consensus, for the `B`=250 clusterings; 1 indicates perfect agreement, and the value decreases with worse agreement. The statistical significance of the difference is confirmed by a Wilcoxon test in the main text. Right: membership probabilities of the consensus clustering; colours are as in the left panel. For E3.25, the probabilities are diffuse, indicating that the individual (resampled) clusterings disagree a lot, whereas for E3.5, the distribution is bimodal, with only one ambiguous sample.

CRAN package: clue

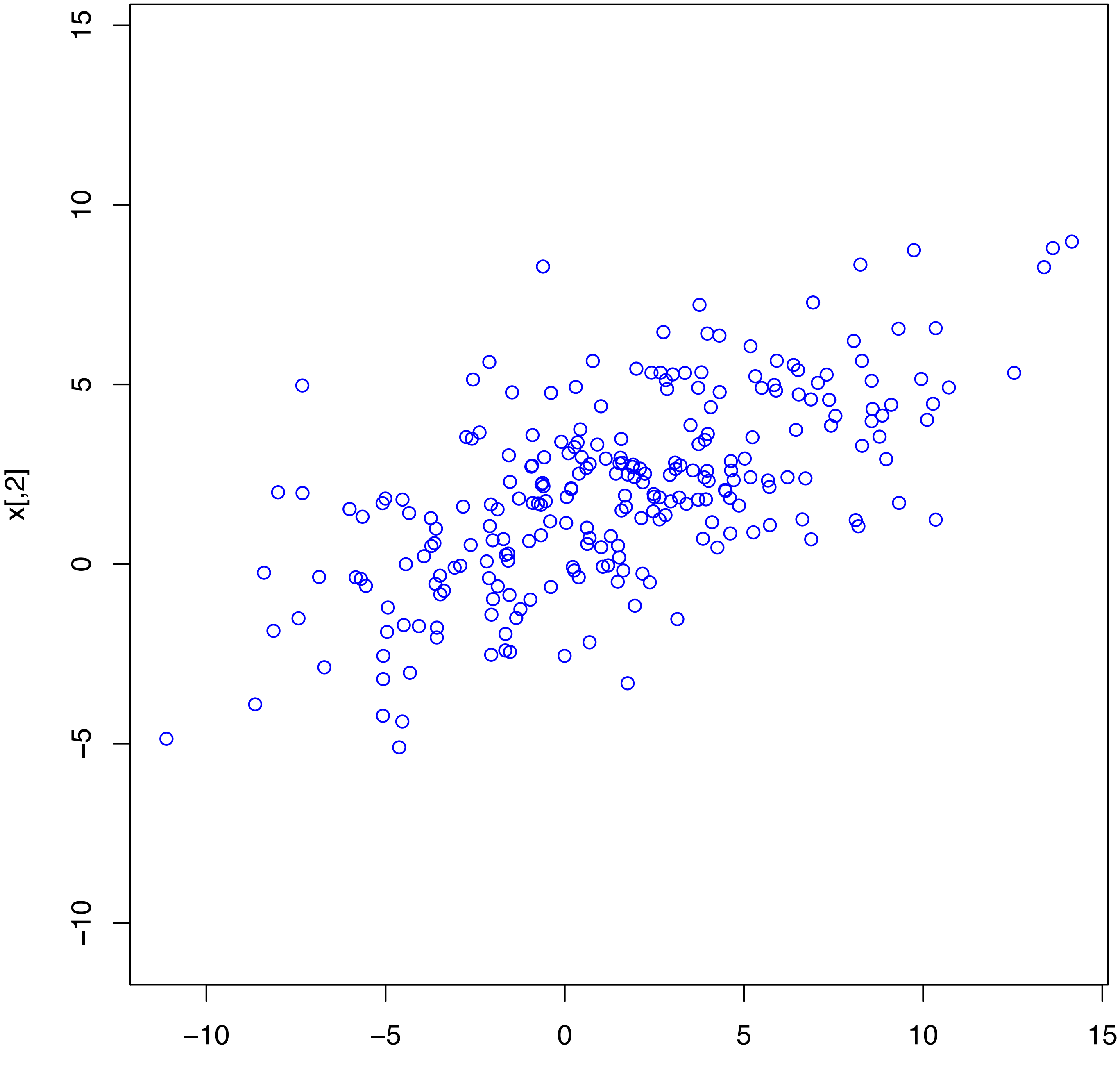
References

Visualizing Genomic Data, R. Gentleman, F. Hahne, W. Huber (2006), Bioconductor Project Working Papers, Paper 10

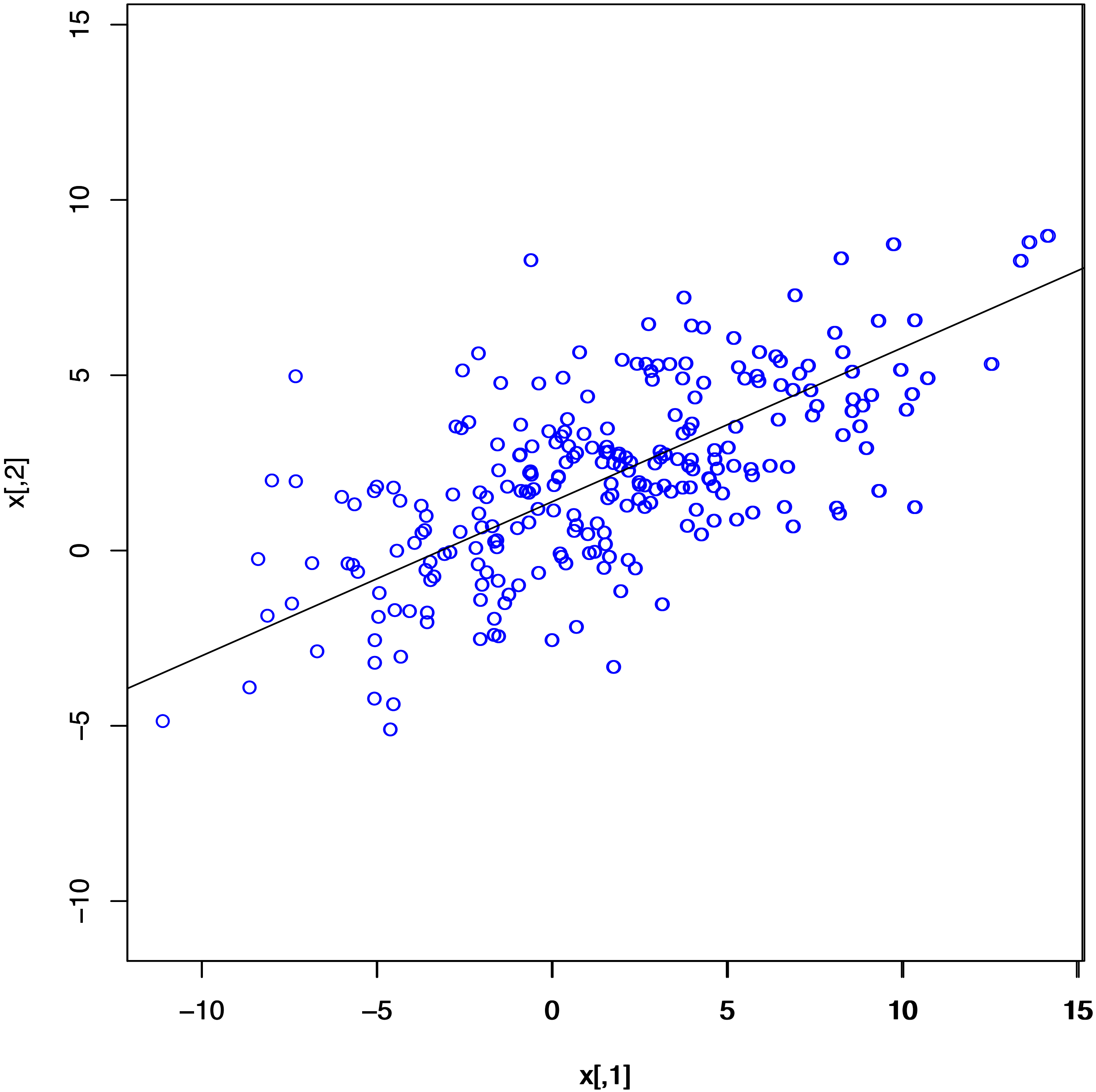
Choosing Color Palettes for Statistical Graphics, A. Zeileis, K. Hornik (2006), Department of Statistics and Mathematics, Wirtschaftsuniversität Wien, Research Report Series, Report 41

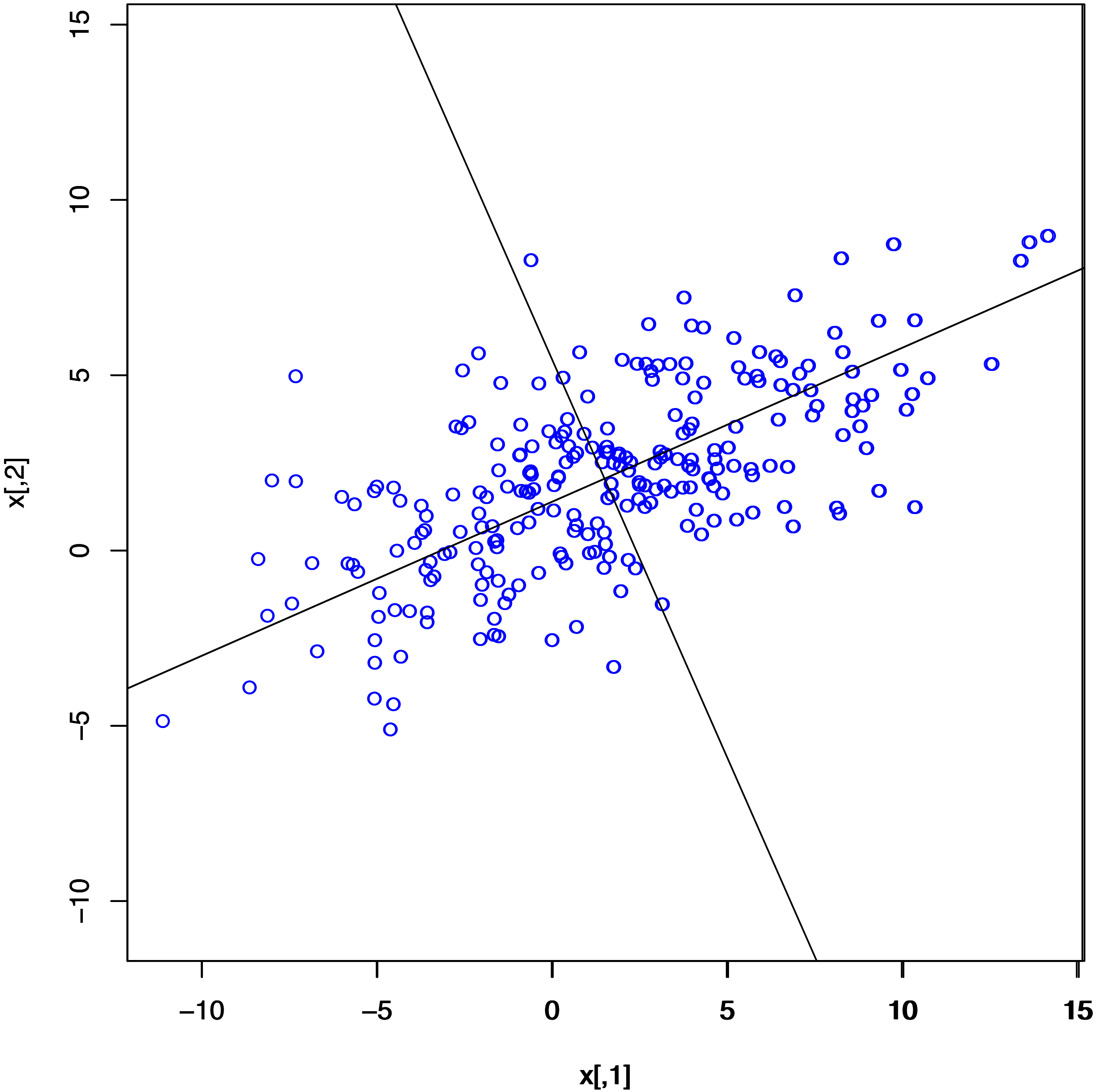


PC2
5.0
2.5
0.0
-2.5
-5.0
-7.5
-10
-5
0
5
10
PC1
queiros_class
n-CLL
i-CLL
m-CLL
good_pred
FALSE
TRUE



15
10
5
0
−5
−10
x[,2]
−10
−5
0
5
10
15
x[,1]


15
10
5
0
−5
−10
x[,2]
−10
−5
0
5
10
15
x[,1]



x[,2]
x[,1]
15
10
5
0
−5
−10
−10
−5
0
5
10
15

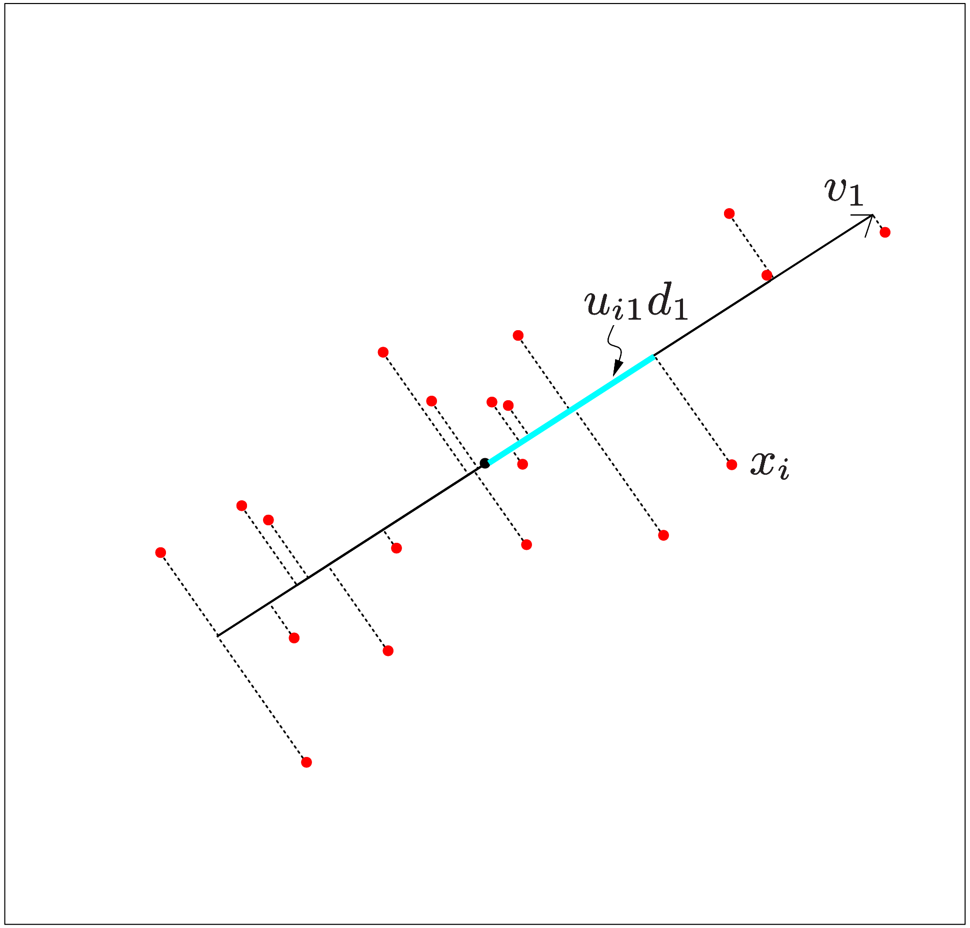


FIGURE 14.20. *The first linear principal component of a set of data. The line minimizes the total squared distance from each point to its orthogonal projection onto the line.*

Hastie, Tibshirani, Friedman

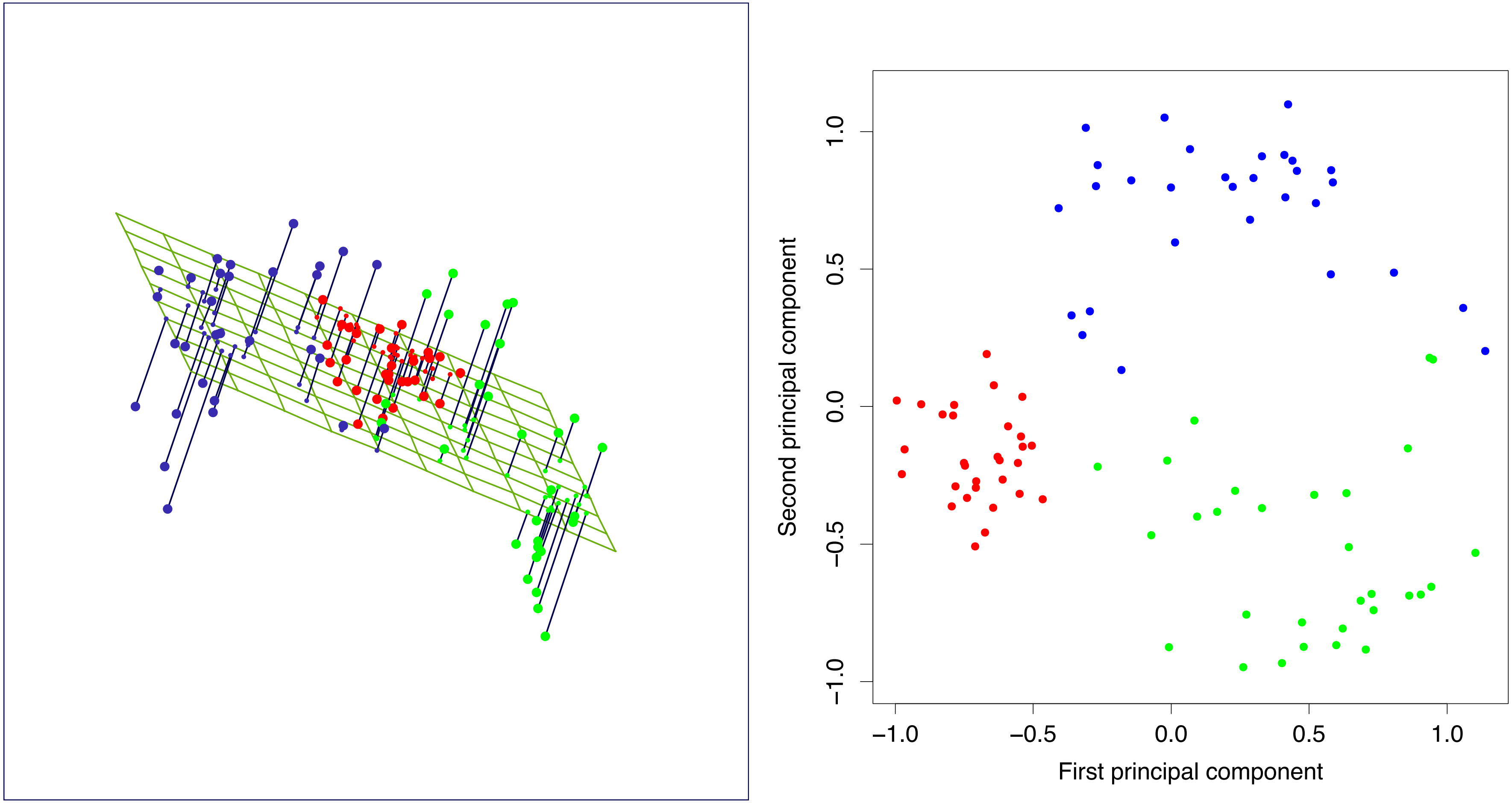


FIGURE 14.21. *The best rank-two linear approximation to the half-sphere data. The right panel shows the projected points with coordinates given by* $\mathbf{U}_2\mathbf{D}_2$*, the first two principal components of the data.*

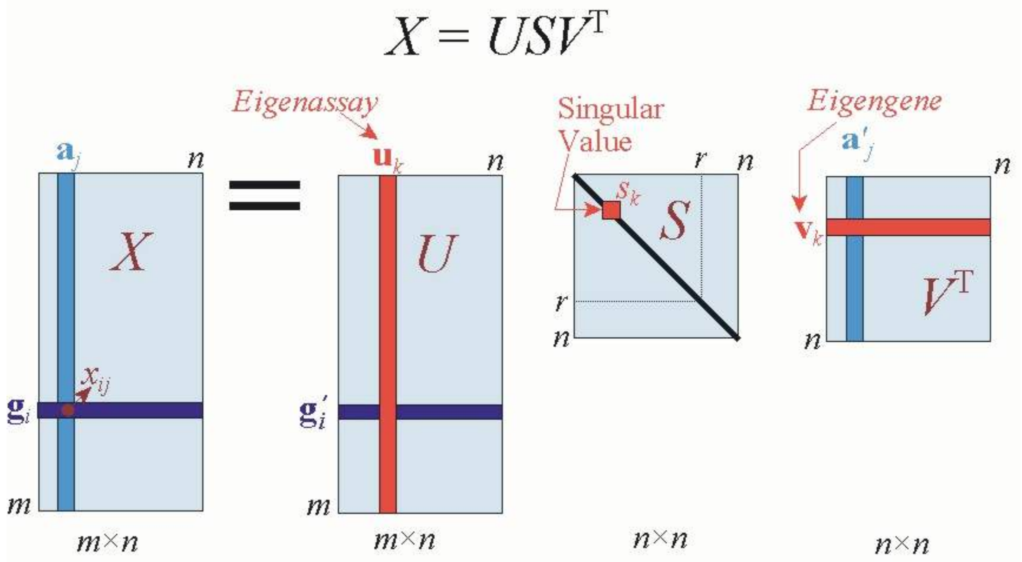
Principal Component Analysis

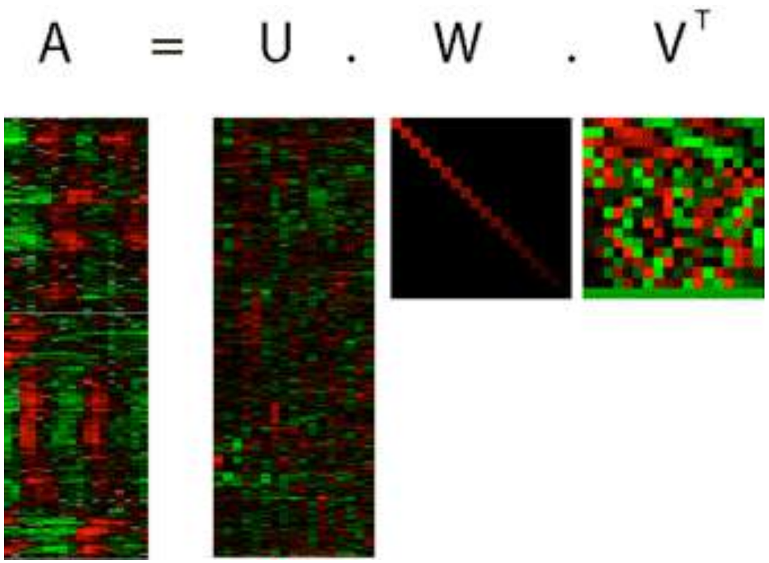
$$\text{Data points: } x_i \in \mathbb{R}^n$$
$$\text{Linear projection: } P : \mathbb{R}^n \mapsto \mathbb{R}^k$$
$$\text{such that}$$

$$\sum_i (x_i - Px_i)^2 \to \min$$

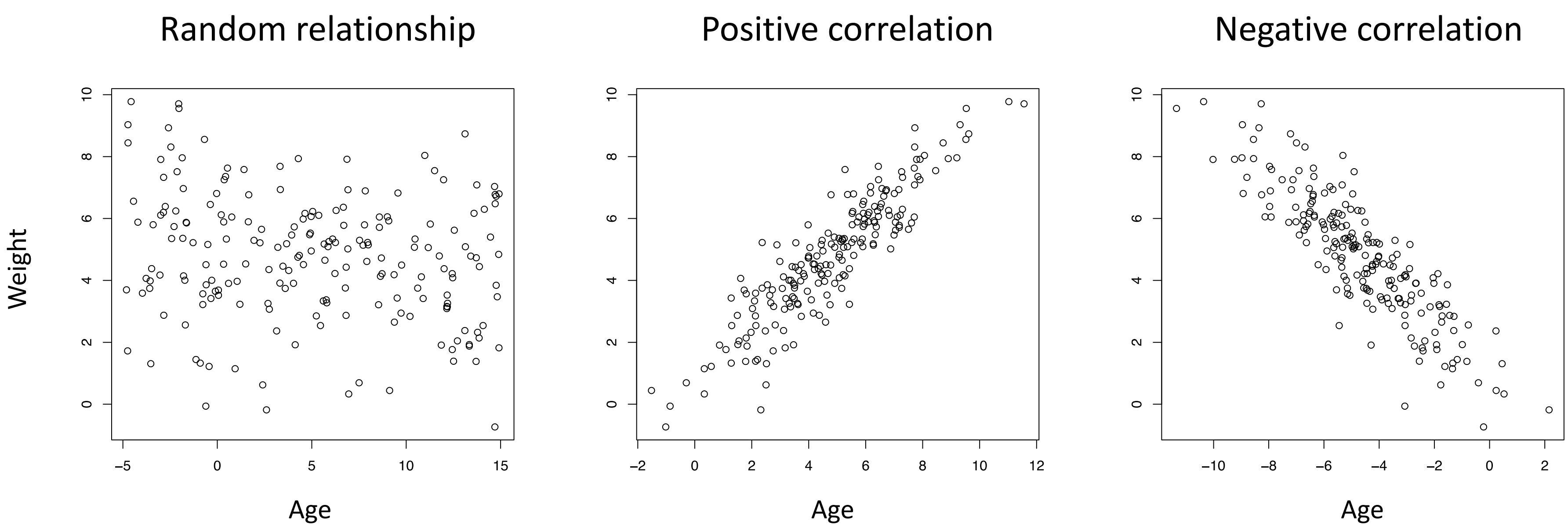
$$\mathrm{Cov}_i \, Px_i \to \max$$

How is the Principal Component Analysis computed?

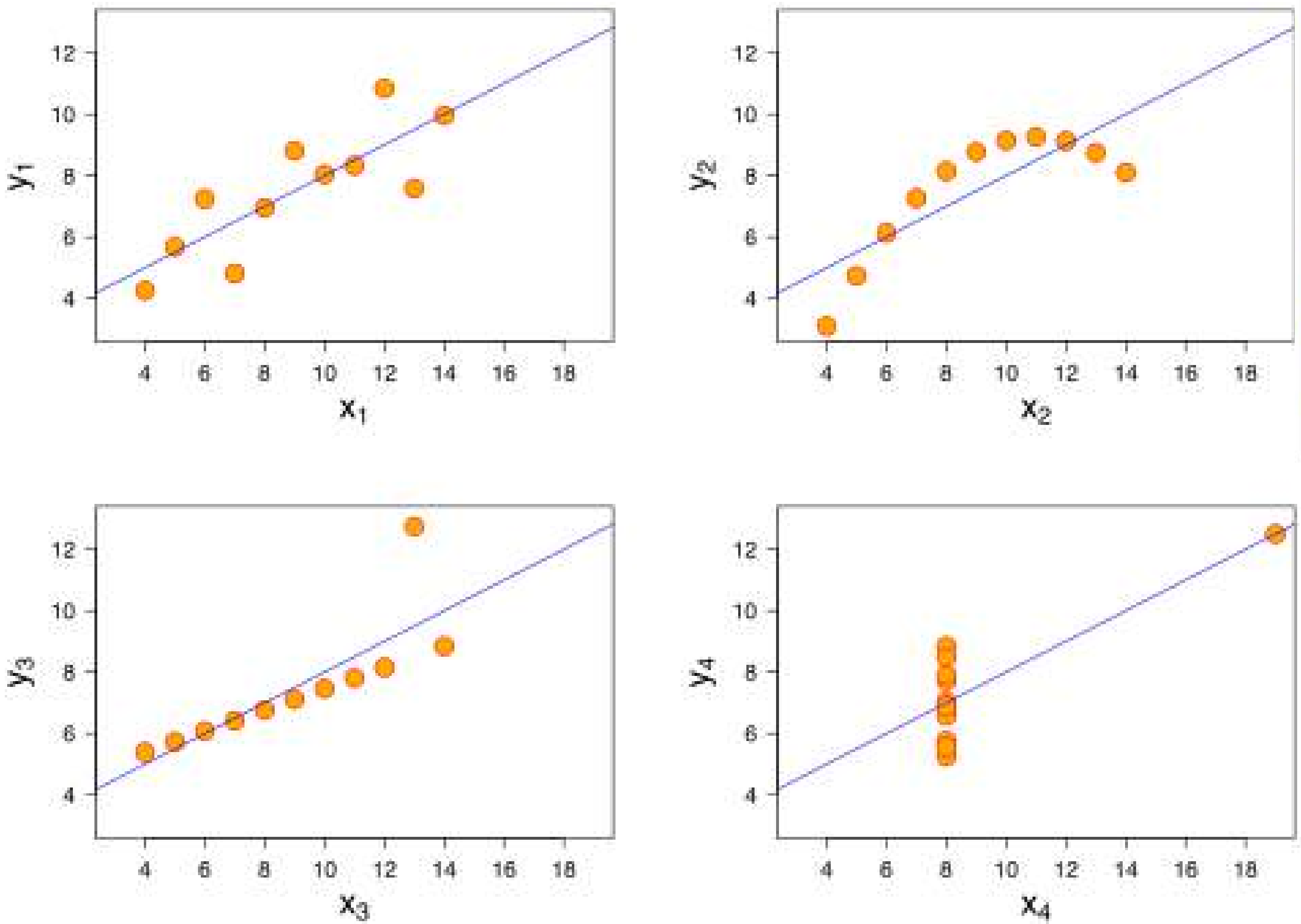




Regression: x vs y



Anscombe’s quartet



Property	Value
Mean of *x* in each case	9 (exact)
Sample variance of *x* in each case	11 (exact)
Mean of *y* in each case	7.50 (to 2 decimal places)
Sample variance of *y* in each case	4.122 or 4.127 (to 3 decimal places)
Correlation between *x* and *y* in each case	0.816 (to 3 decimal places)
Linear regression line in each case	$y = 3.00 + 0.500x$ (to 2 and 3 decimal places, respectively)

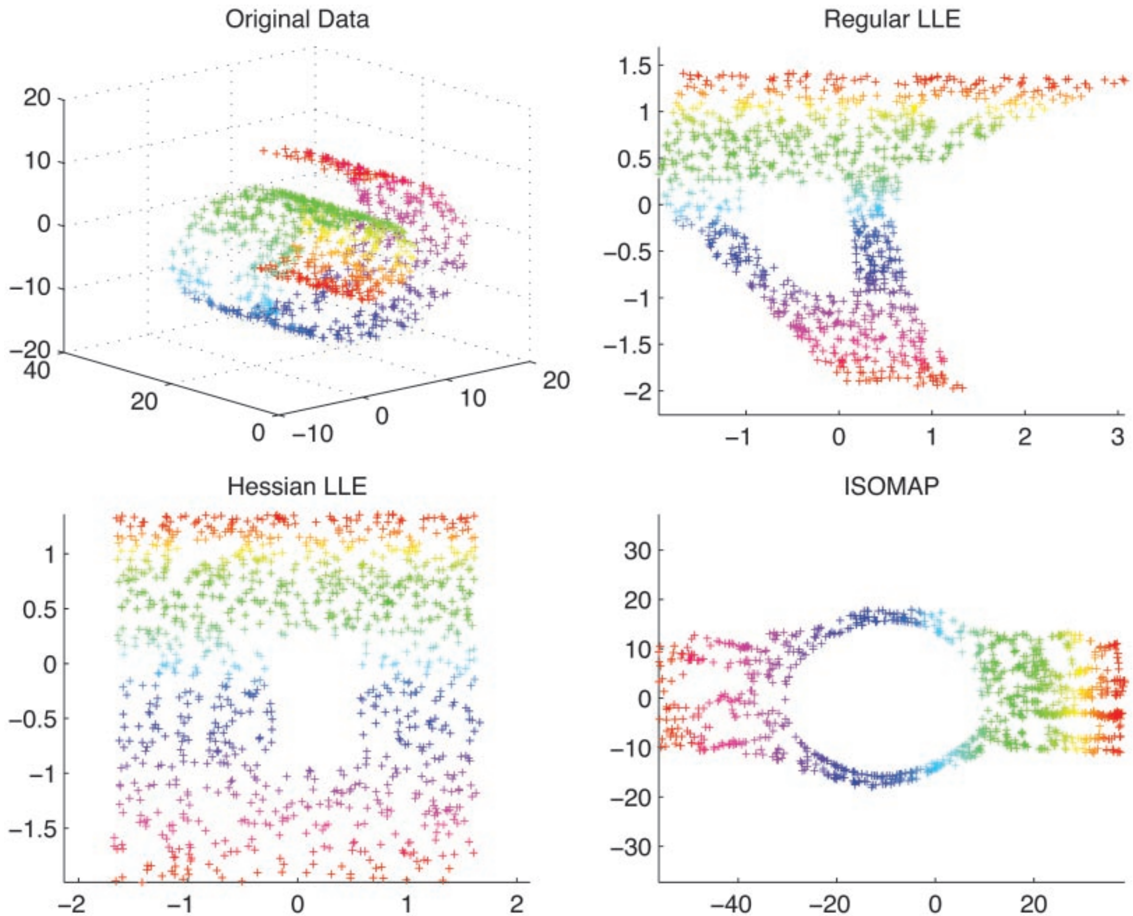


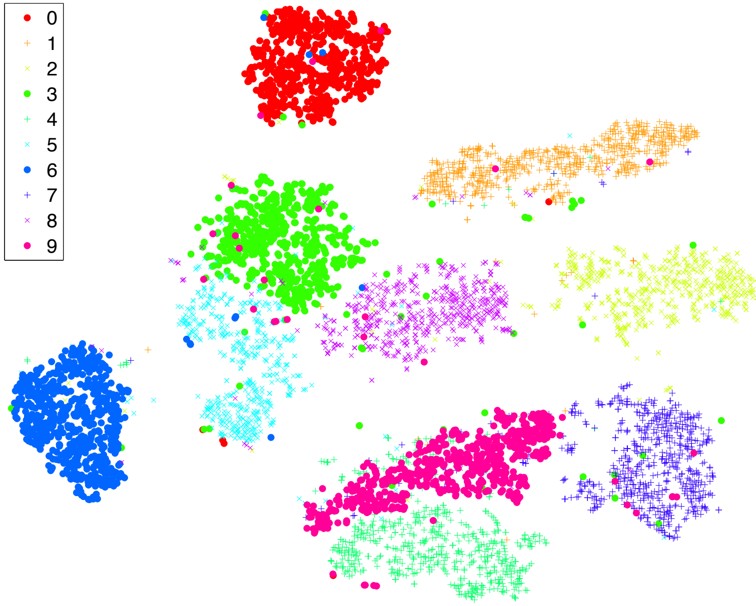
Fig. 1. (*Upper Left*) Original data. (*Upper Right*) LLE embedding (Roweis and Saul code, $k = 12$; ref. 4). (*Lower Left*) Hessian eigenmaps (Donoho and Grimes code, $k = 12$; as described in section 5). (*Lower Right*) ISOMAP (Tenenbaum *et al.* code, $k = 7$; ref. 1). The underlying correct parameter space that generated the data is a square with a central square removed, similar to what is obtained by the Hessian approach (*Lower Left*).

Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data

David L. Donoho* and Carrie Grimes

Department of Statistics, Stanford University, Stanford, CA 94305-4065

PNAS | **May 13, 2003** | vol. 100 | no. 10 | **5591–5596**



(a) Visualization by t-SNE.

(b) Visualization by Sammon mapping.

Figure 2: Visualizations of 6,000 handwritten digits from the MNIST data set.

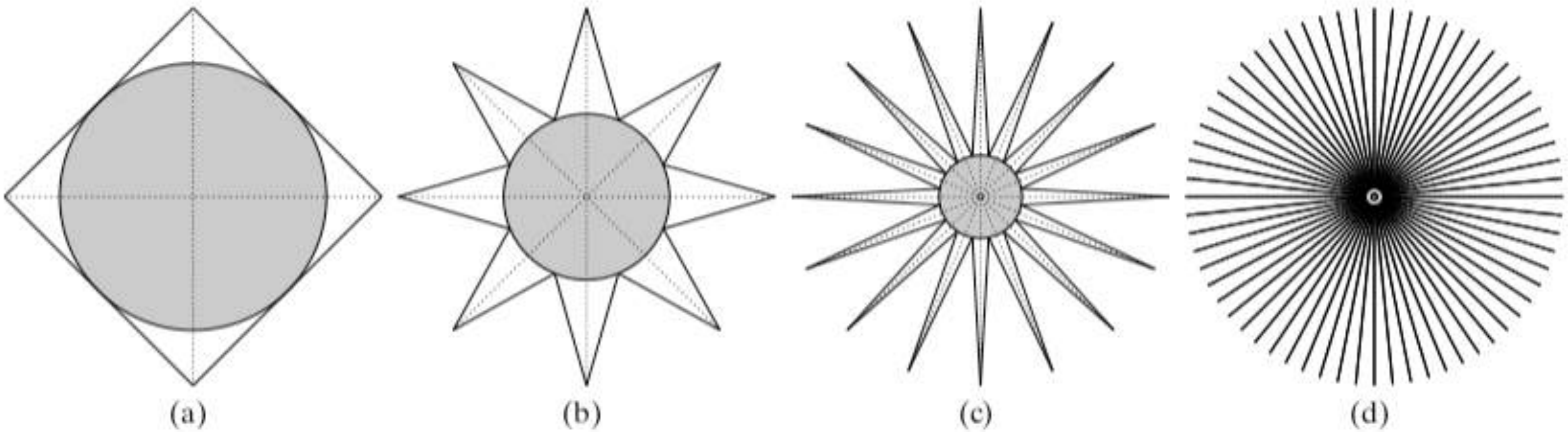
Journal of Machine Learning Research 9 (2008) 2579-2605 Submitted 5/08; Revised 9/08; Published 11/08

Visualizing Data using t-SNE

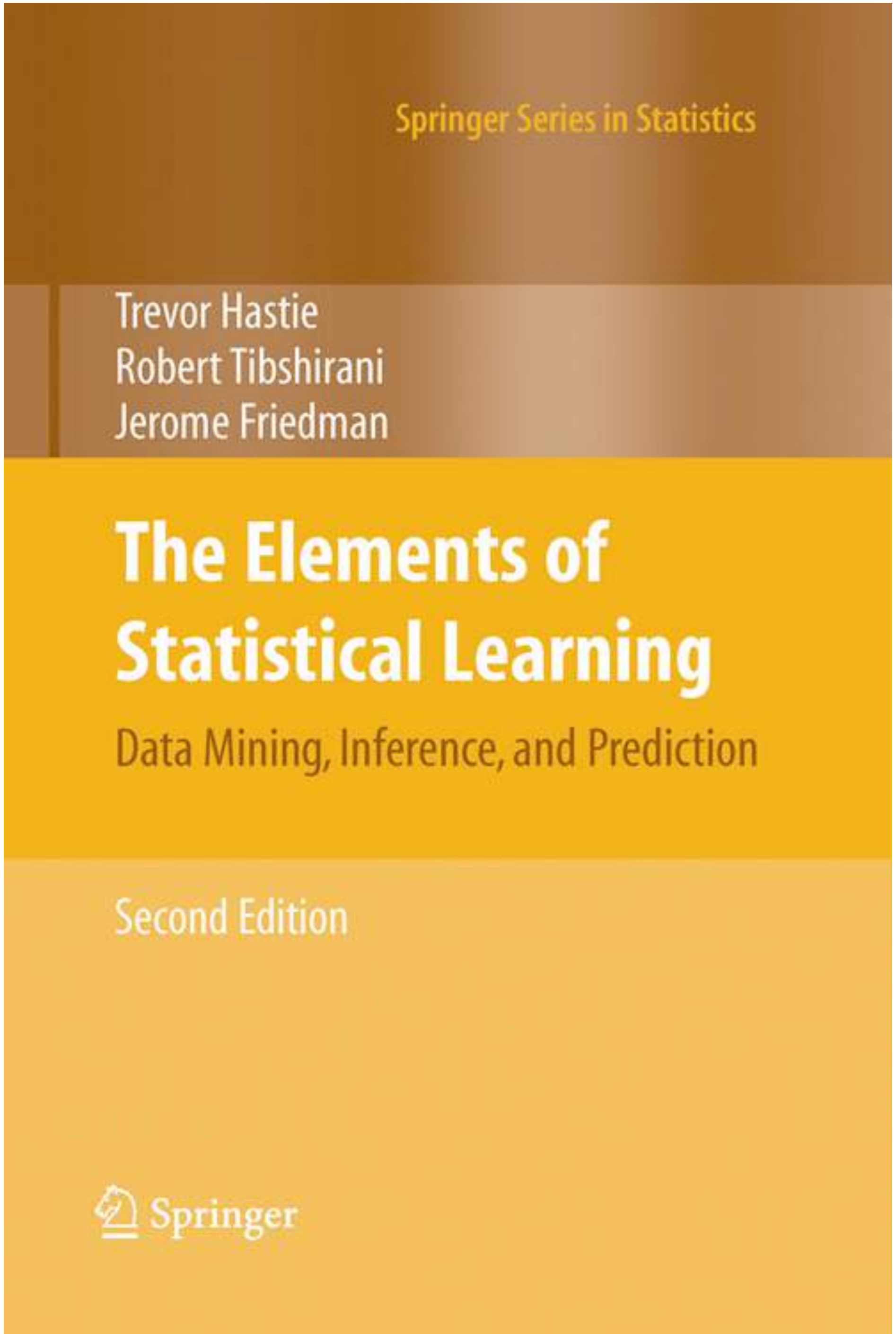
Laurens van der Maaten LVDMAATEN@GMAIL.COM
TiCC
Tilburg University
P.O. Box 90153, 5000 LE Tilburg, The Netherlands

Geoffrey Hinton HINTON@CS.TORONTO.EDU
Department of Computer Science
University of Toronto
6 King's College Road, M5S 3G4 Toronto, ON, Canada

"Curse" of dimensionality



$$\lim_{n\to\infty} \frac{|S_n|}{|C_n|} = \frac{\pi^{\frac{n}{2}}\, r^n}{\Gamma\left(\frac{n}{2}+1\right)} \frac{1}{(2r)^n} \to 0$$


Springer Series in Statistics
Trevor Hastie
Robert Tibshirani
Jerome Friedman
The Elements of Statistical Learning
Data Mining, Inference, and Prediction
Second Edition
Springer