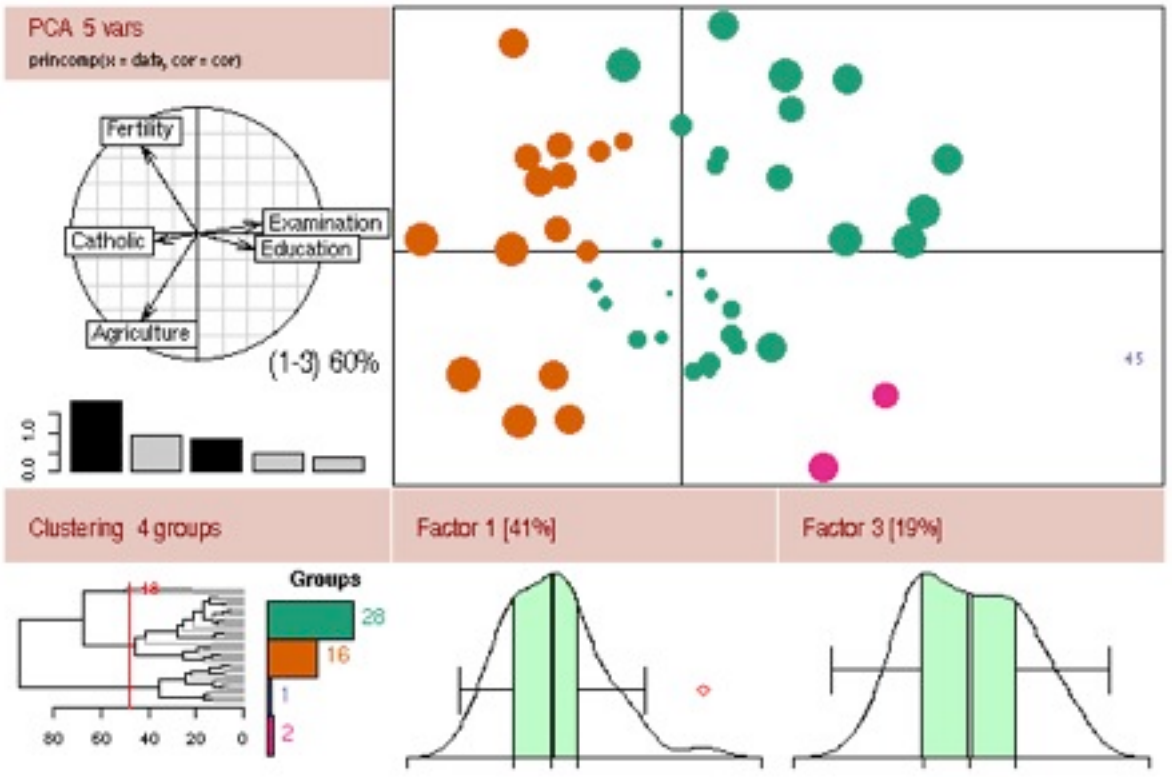


Visualisation

Wolfgang Huber

Overview

Visualisation

- 1-dim. data: distributions
- 2-dim. data: scatterplots
- 3-dim. data: pseudo-3D displays
- a few more than 2-dim: colours, drill-down, lattice, parallel coordinates
- High-dimensional data

Univariate data

Suppose you have samples of univariate measurements:

Set 1: `0.81, 3.36, 6.84, 9.36, 2.91, 1.81, 5.07, 1.26, 7.89, 9.15, 3.30, 4.35, …`

Set 2: `6.57, 5.92, 5.78, 6.63, 5.38, 5.98, 6.30, 6.34, 6.45, 6.57, 6.40, 5.89, …`

How do you visualize that?

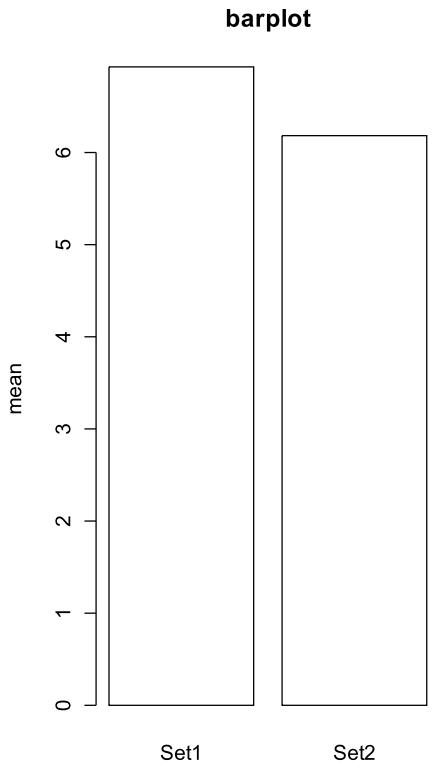
Univariate data

Suppose you have samples of univariate measurements:

Set 1: `0.81, 3.36, 6.84, 9.36, 2.91, 1.81, 5.07, 1.26, 7.89, 9.15, 3.30, 4.35, …`

Set 2: `6.57, 5.92, 5.78, 6.63, 5.38, 5.98, 6.30, 6.34, 6.45, 6.57, 6.40, 5.89, …`

How do you visualize that?



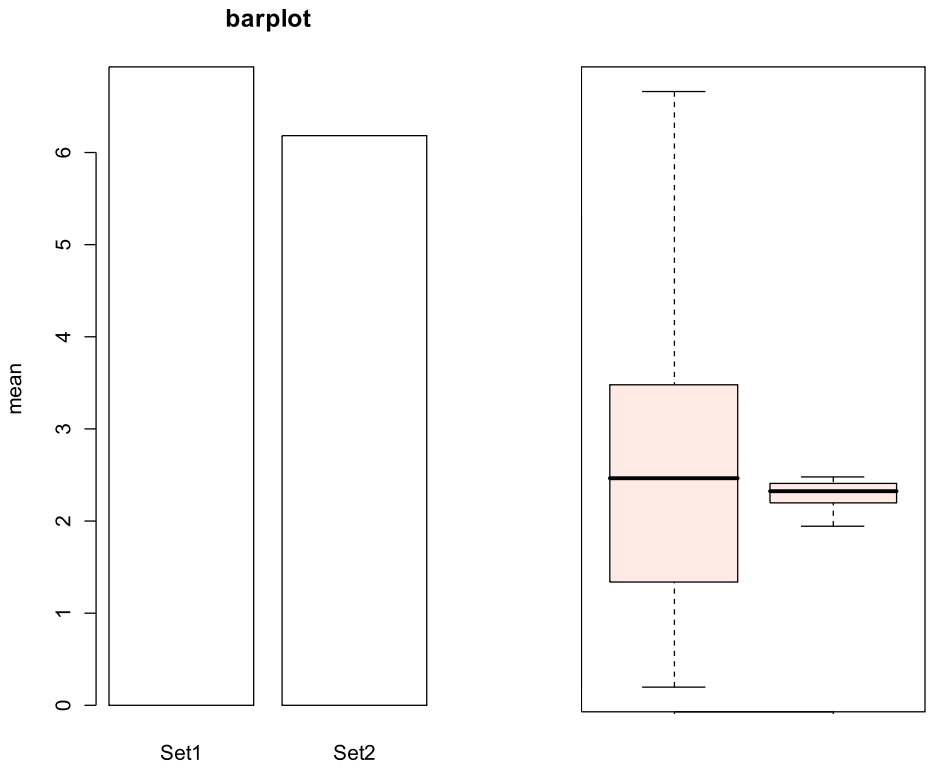
Univariate data

Suppose you have samples of univariate measurements:

Set 1: `0.81, 3.36, 6.84, 9.36, 2.91, 1.81, 5.07, 1.26, 7.89, 9.15, 3.30, 4.35, …`

Set 2: `6.57, 5.92, 5.78, 6.63, 5.38, 5.98, 6.30, 6.34, 6.45, 6.57, 6.40, 5.89, …`

How do you visualize that?



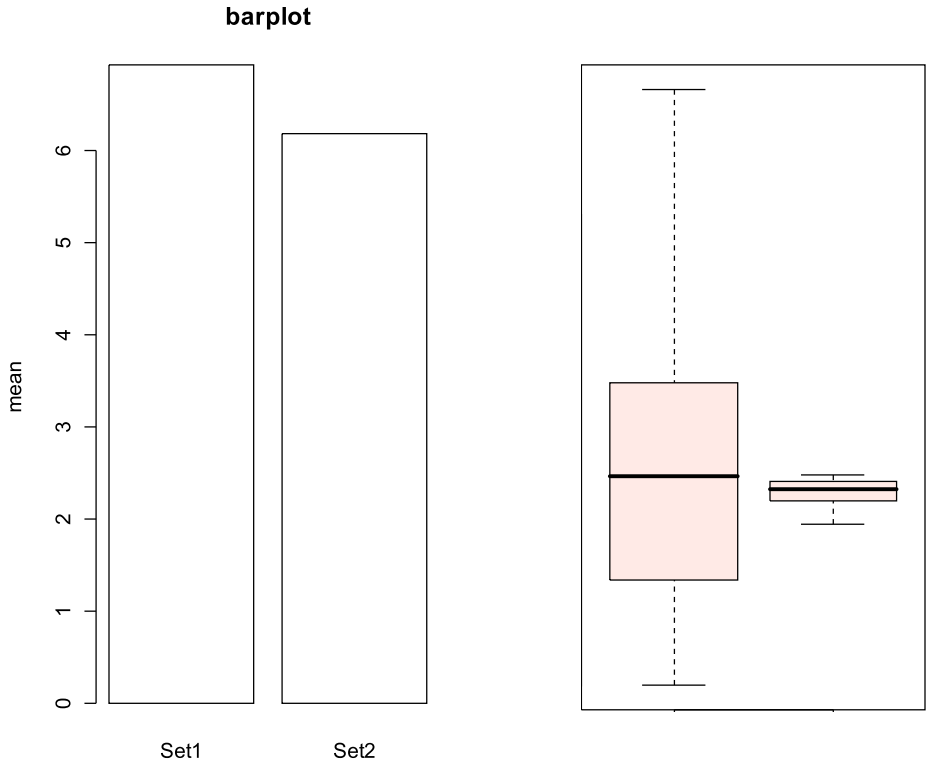
Univariate data

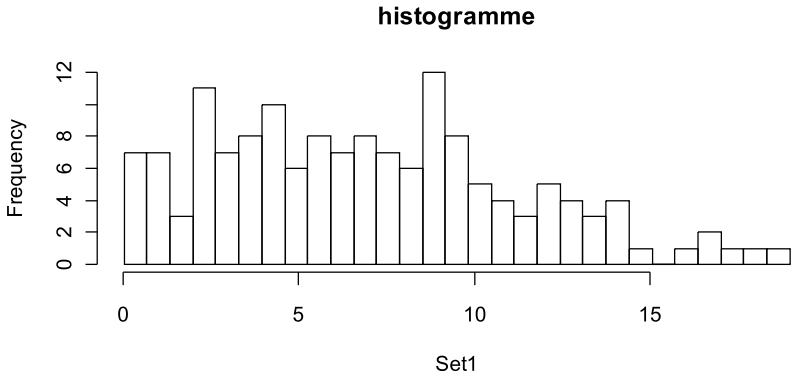
Suppose you have samples of univariate measurements:

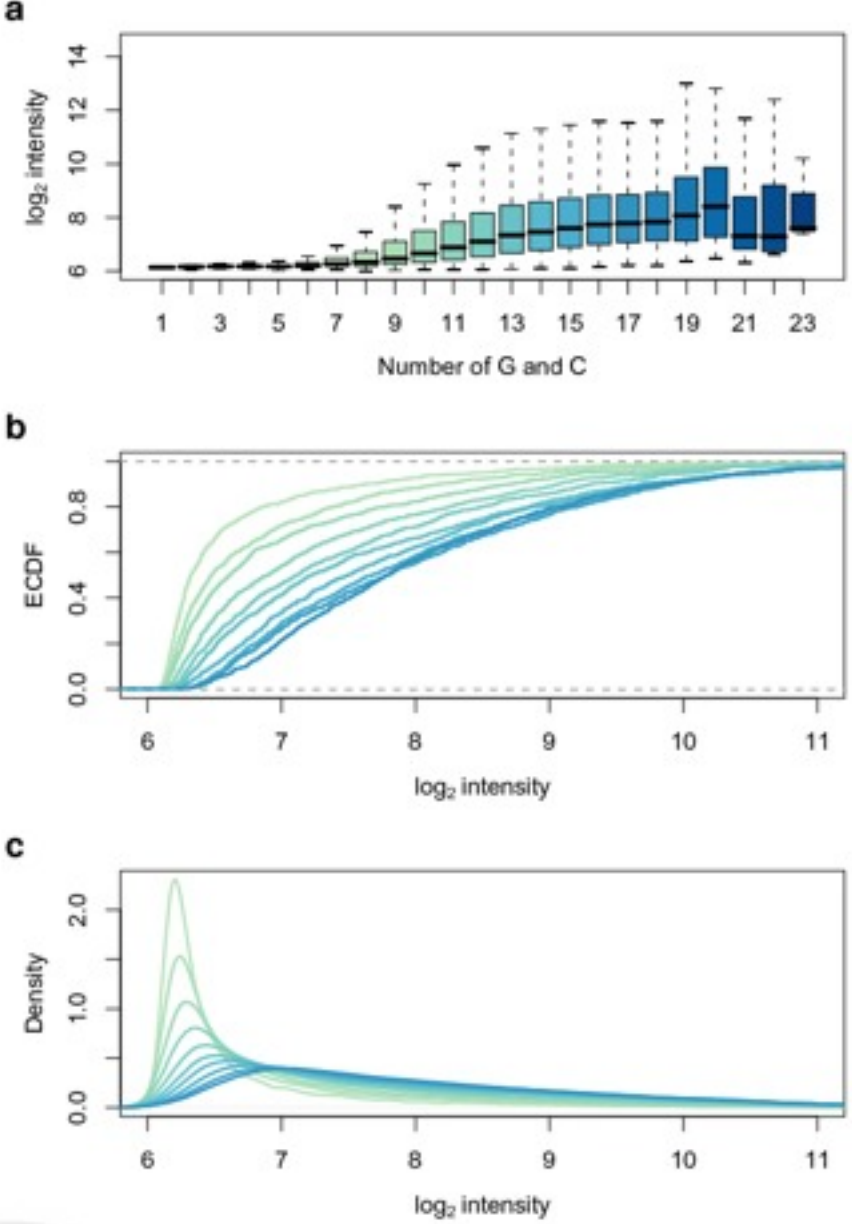
Set 1: `0.81, 3.36, 6.84, 9.36, 2.91, 1.81, 5.07, 1.26, 7.89, 9.15, 3.30, 4.35, …`

Set 2: `6.57, 5.92, 5.78, 6.63, 5.38, 5.98, 6.30, 6.34, 6.45, 6.57, 6.40, 5.89, …`

How do you visualize that?







ECDF(x) = fraction of data with values ≤x

Figure 7: Distributions of the $\log_2$–intensities from the CLL dataset (see Section 2) grouped by the number of cytosines (C) and guanines (G) among the 25 nucleotides in each probe.

Density estimation

If $x_1, x_2, \ldots, x_N \sim f$ is an IID sample of a random variable, then the kernel density approximation of its probability density function is

$$\widehat{f}_h(x) = \frac{1}{Nh} \sum_{i=1}^{N} K\left(\frac{x - x_i}{h}\right)$$

where K is some kernel and h is the bandwidth (smoothing parameter). Quite often K is taken to be a standard Gaussian function with mean zero and variance 1:

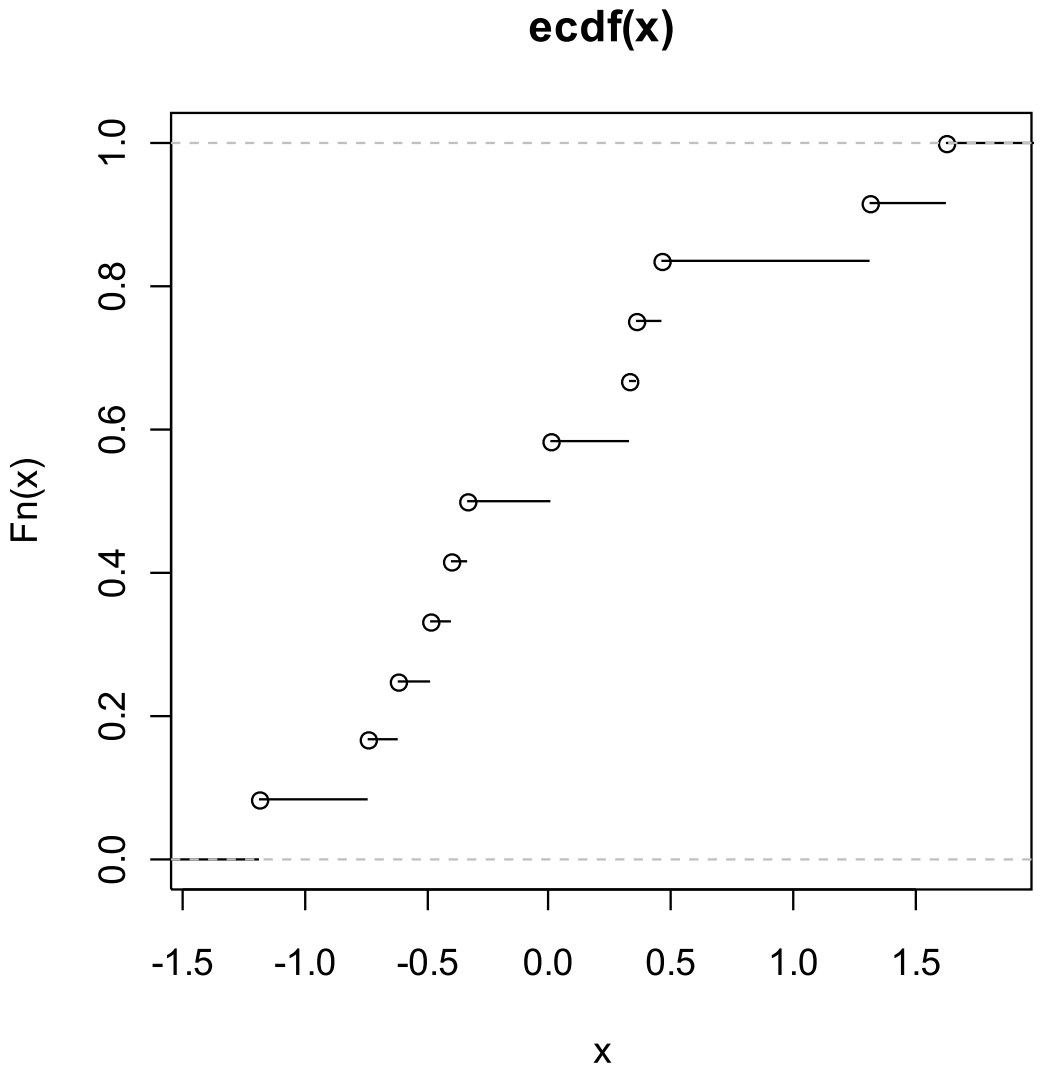
$$K(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}.$$

R function **`density`**:
(i) disperses the mass of the empirical distribution over a regular grid of >= 512 points,
(i) uses the fast Fourier transform to convolve this approximation with a discretized version of the kernel,
(iii) uses linear approximation to evaluate the density at the specified points.

Empirical Cumulative Distribution Function: ecdf

```
x = rnorm(12)
Fn = ecdf(x)
plot(Fn)
```

Fn(x) is the fraction of data points with a value ≤ *x*.



Discussion: boxplot, histogramme, density, ecdf

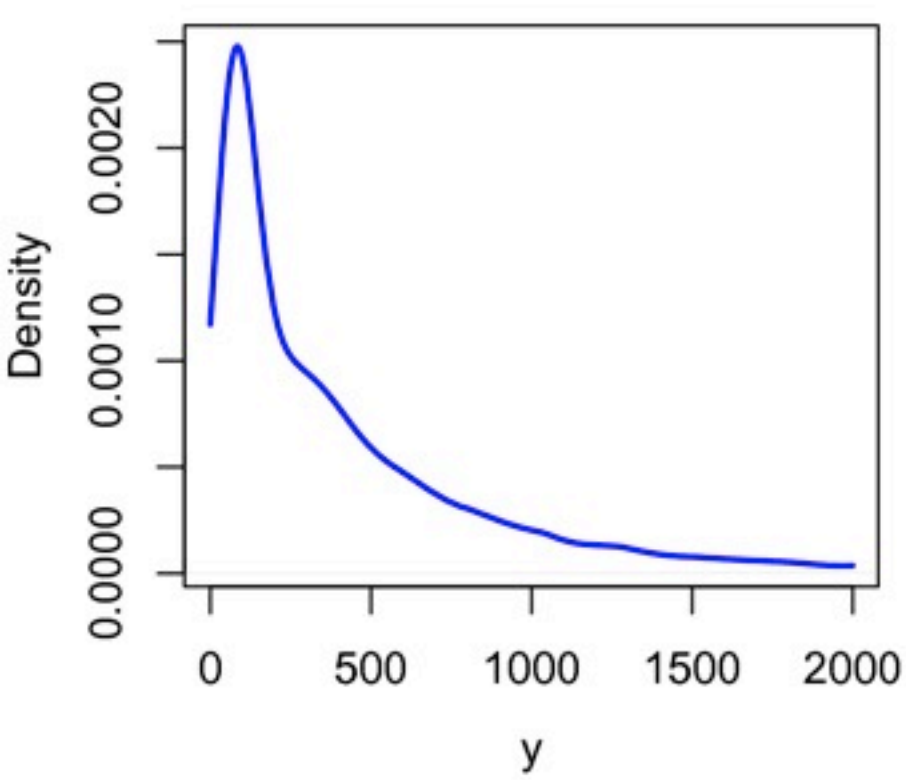
Boxplot makes sense for unimodal distributions

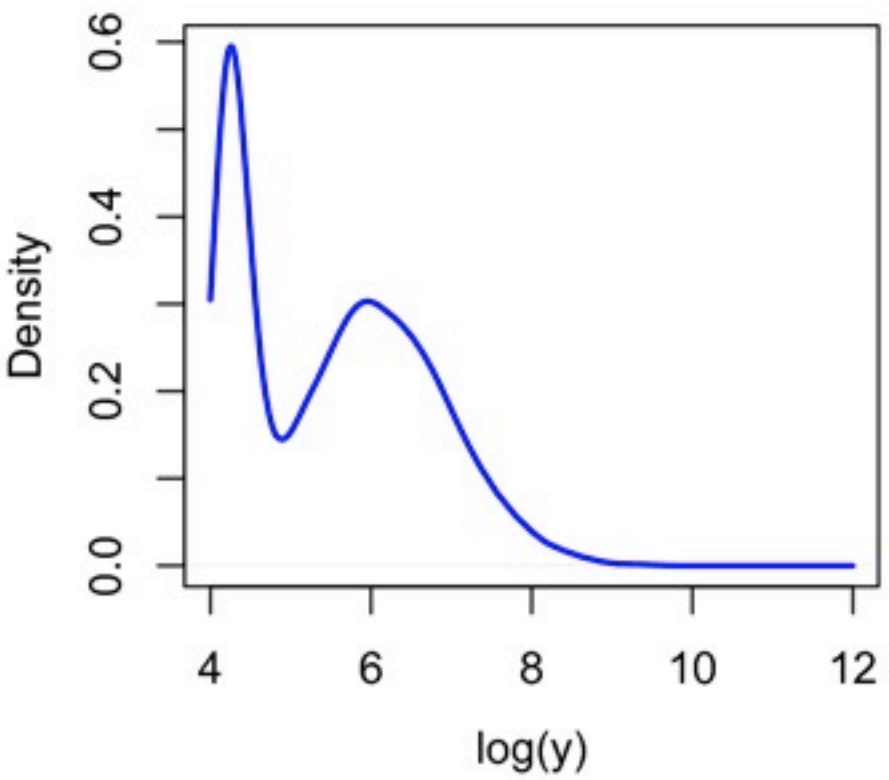
Histogram requires definition of bins (width, positions) and can create visual artifacts esp. if the number of data points is not large

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 its 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.

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





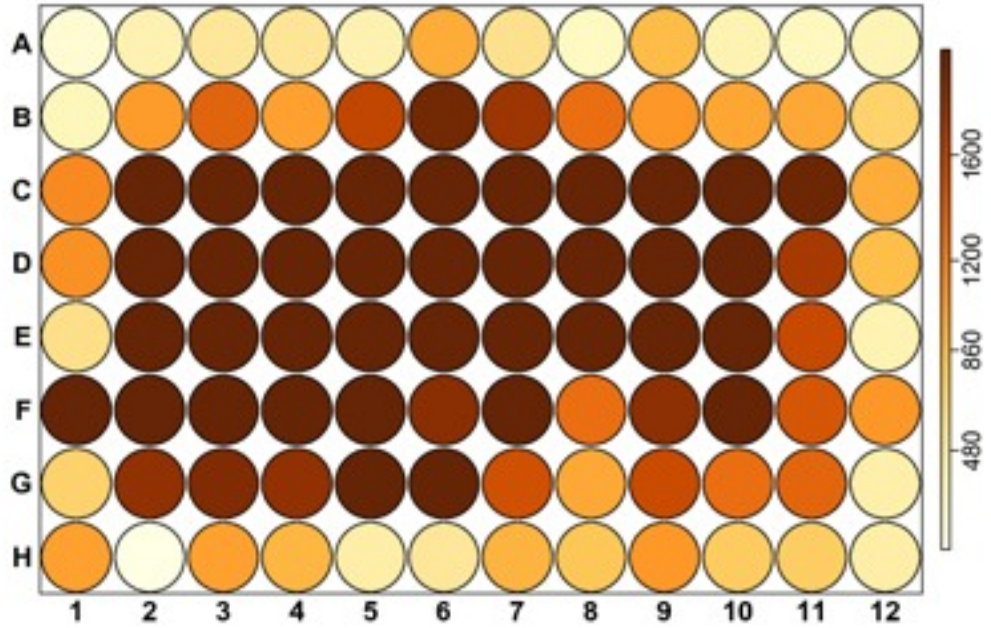
y: sample from a mixture of two log-normal distributions
kernel density estimates

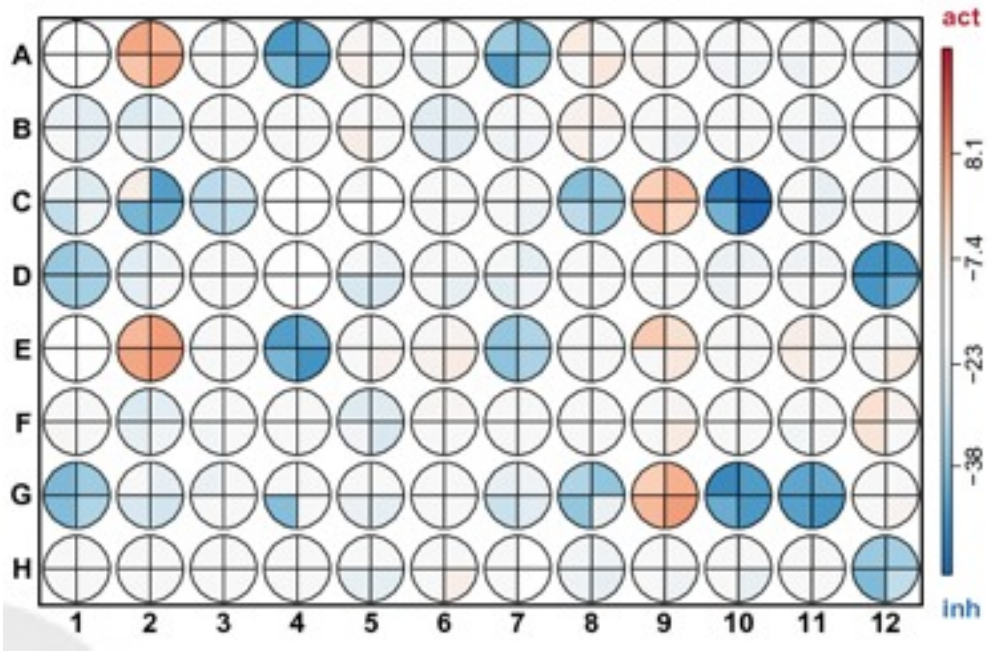
Univariate measurements from a 96-well microtitre plate experiment

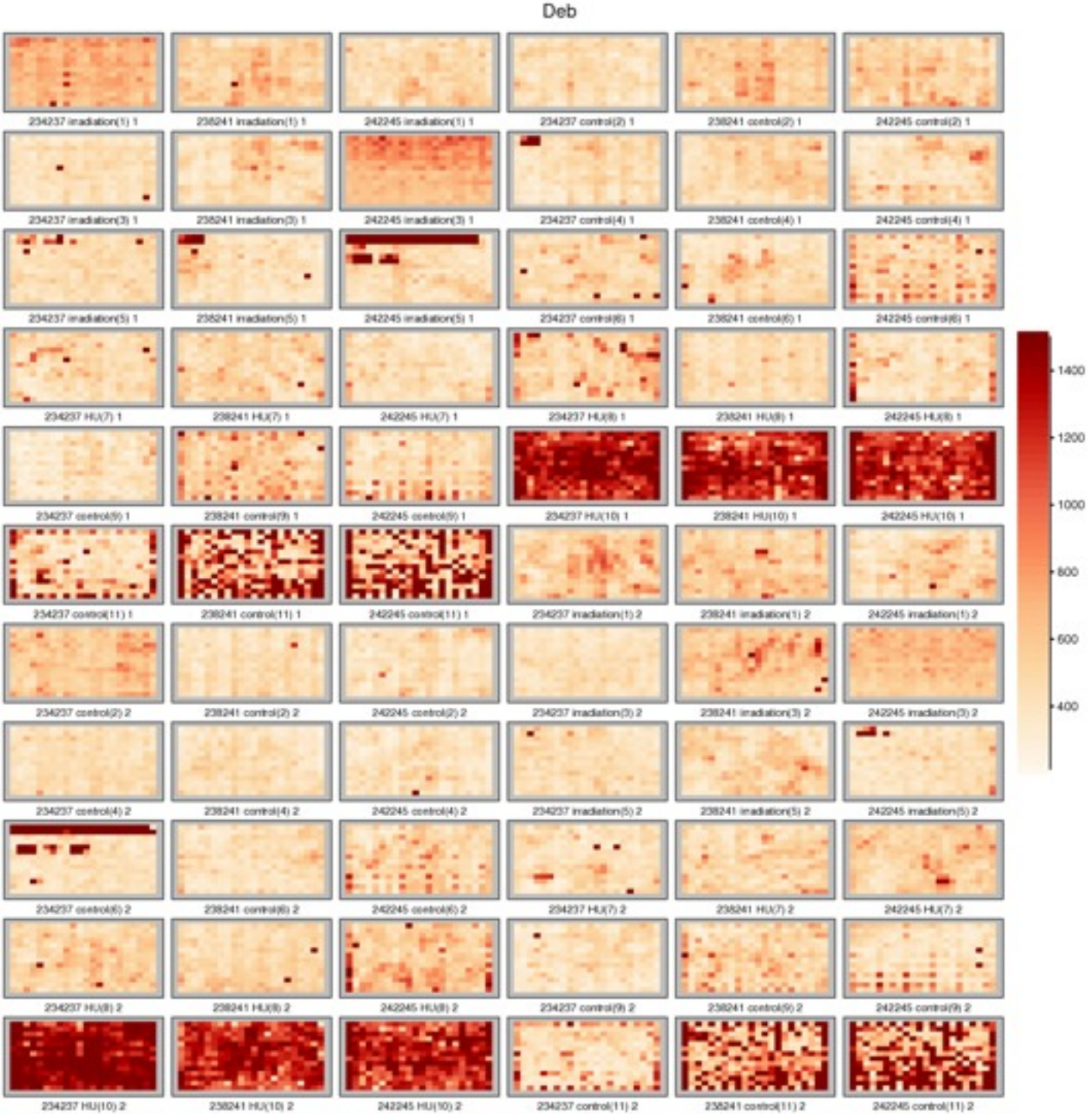
Detection of edge effects

Replicate reproducibility

package
prada

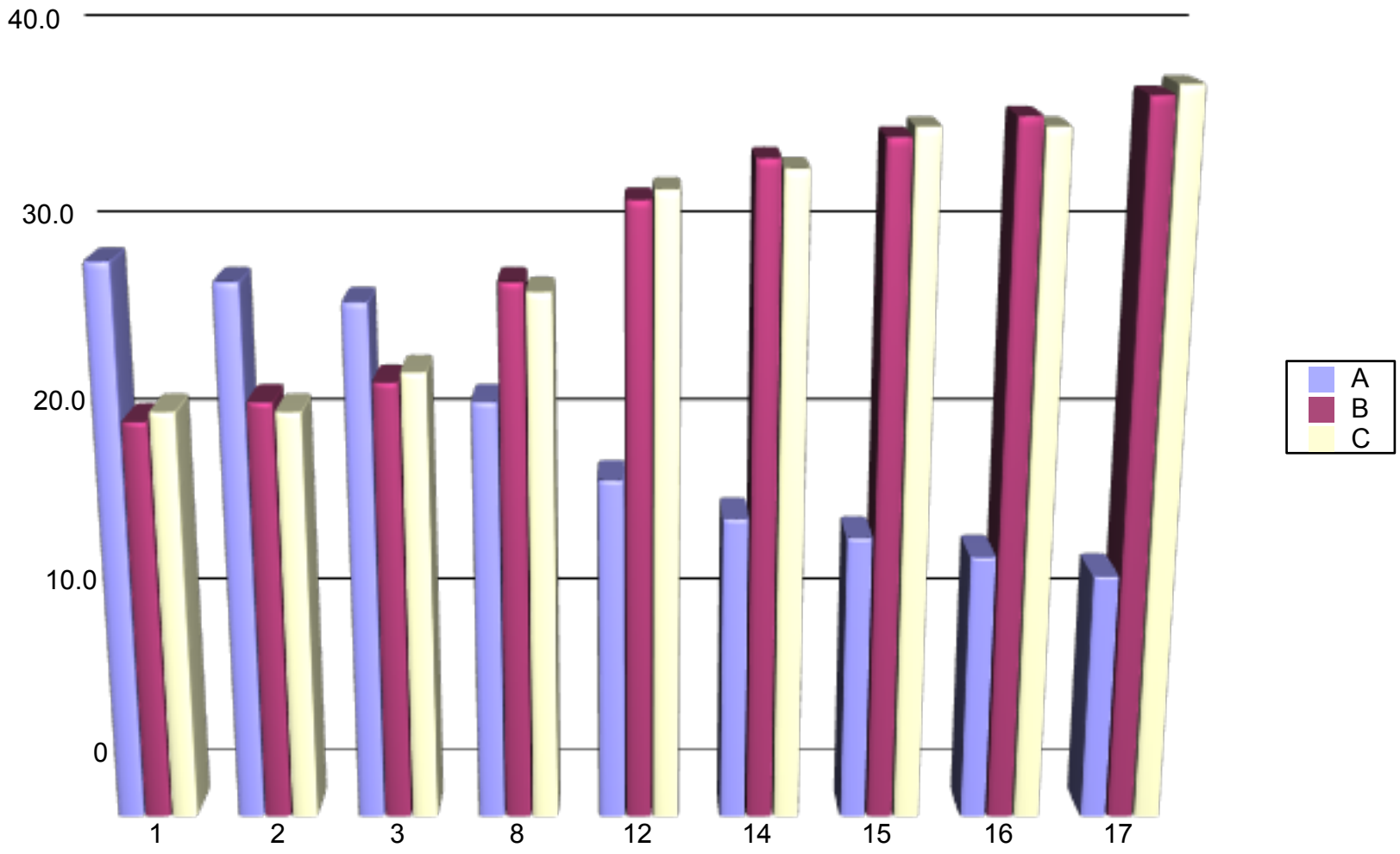


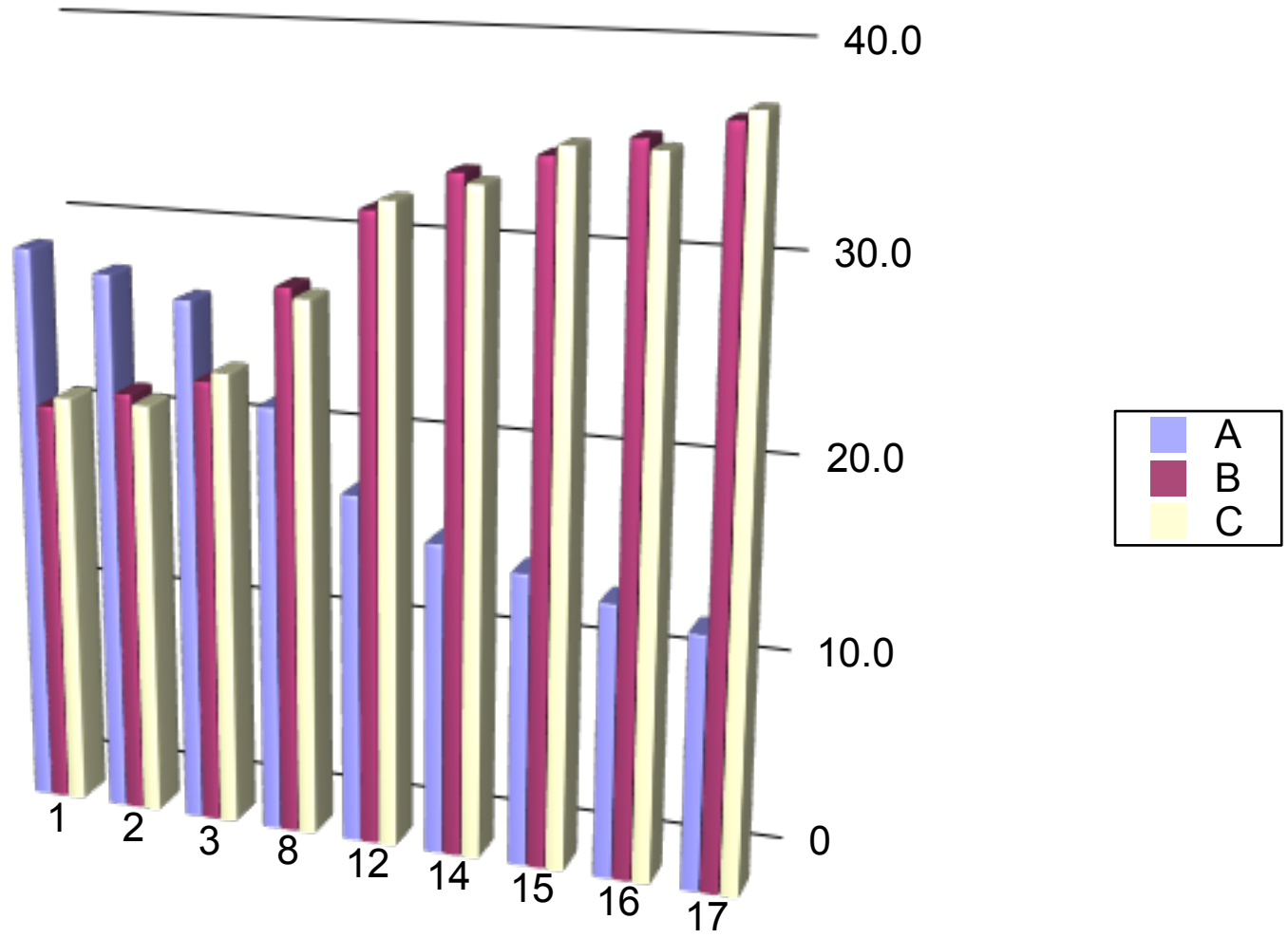


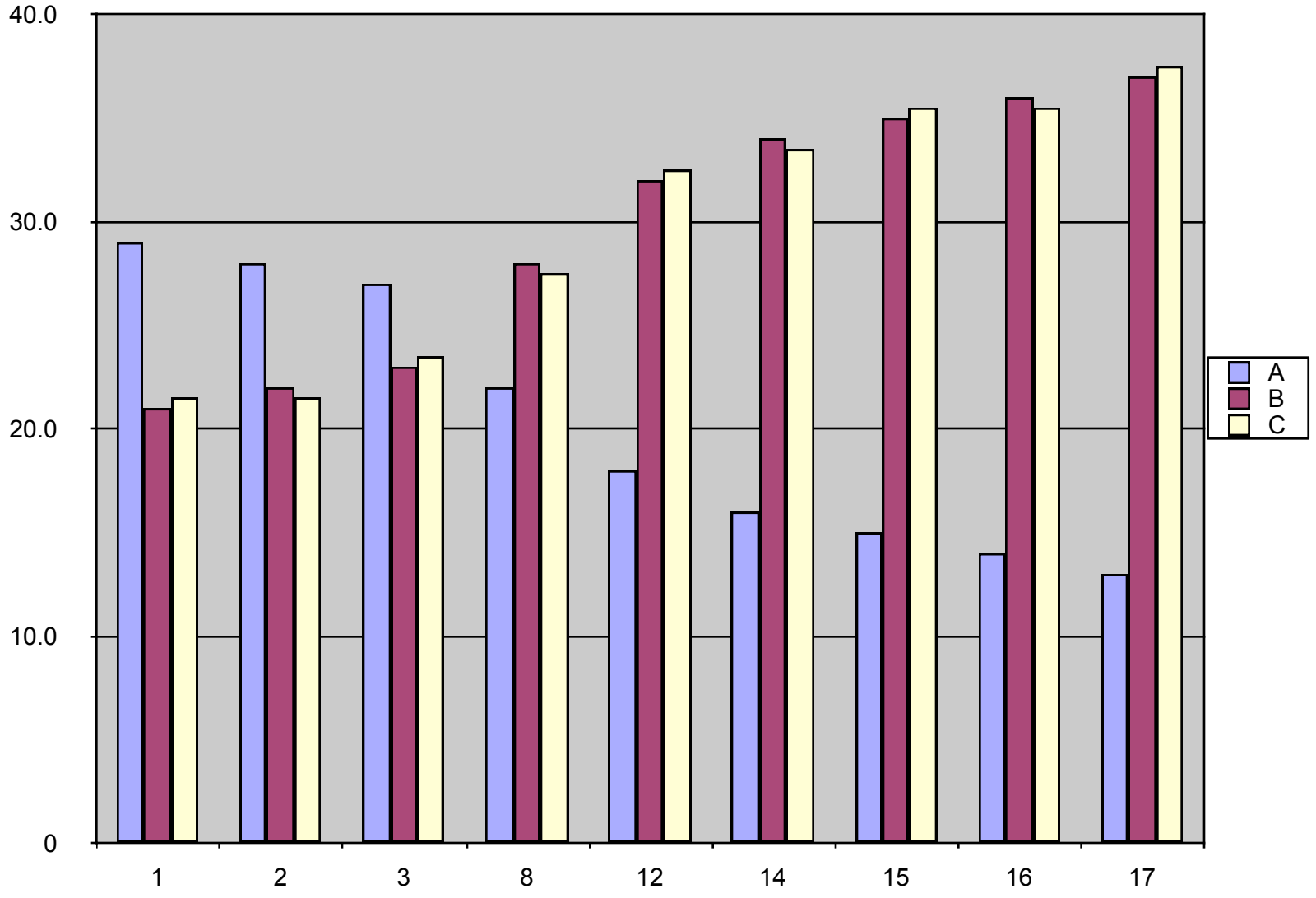


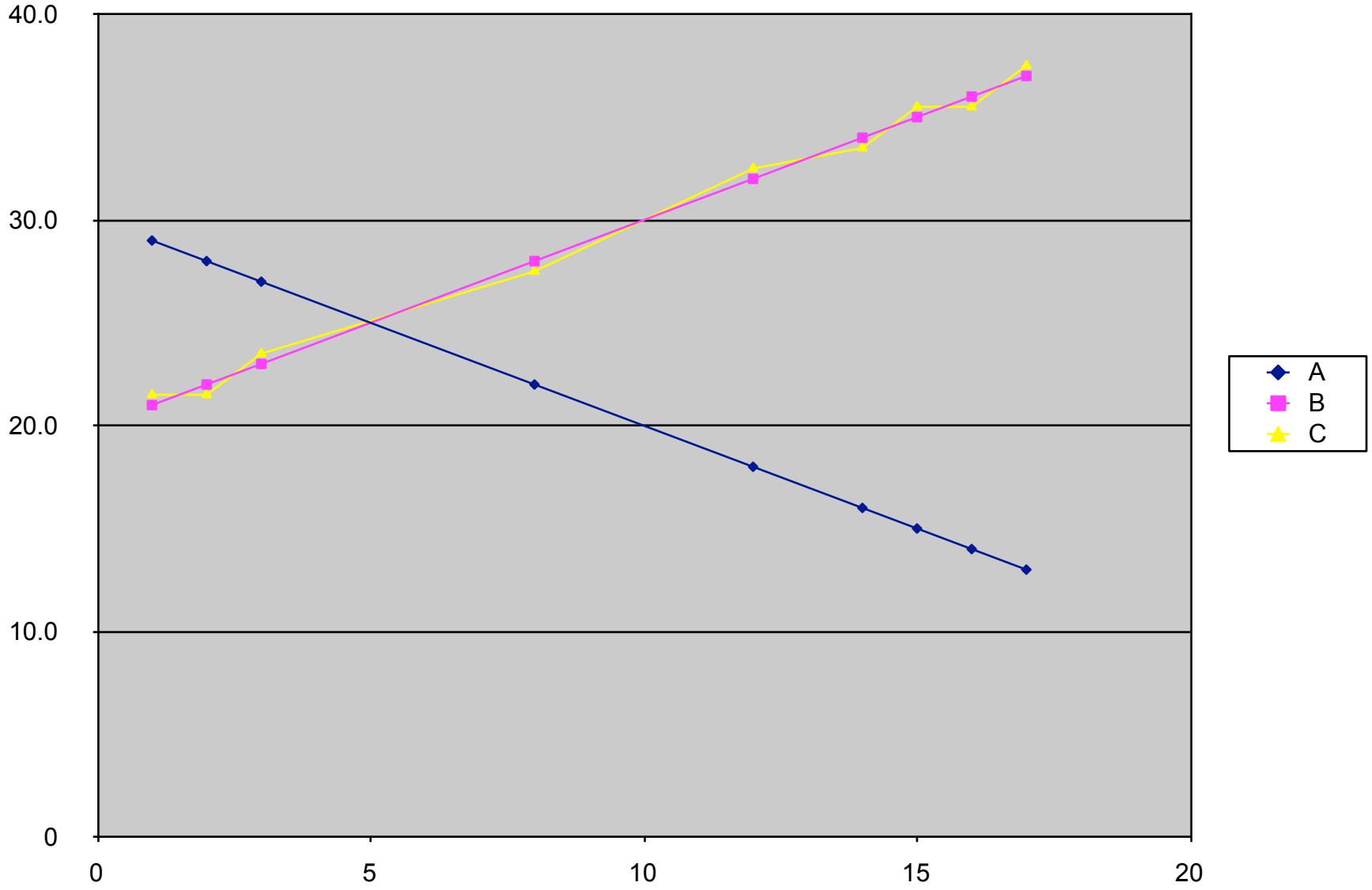
package
splots

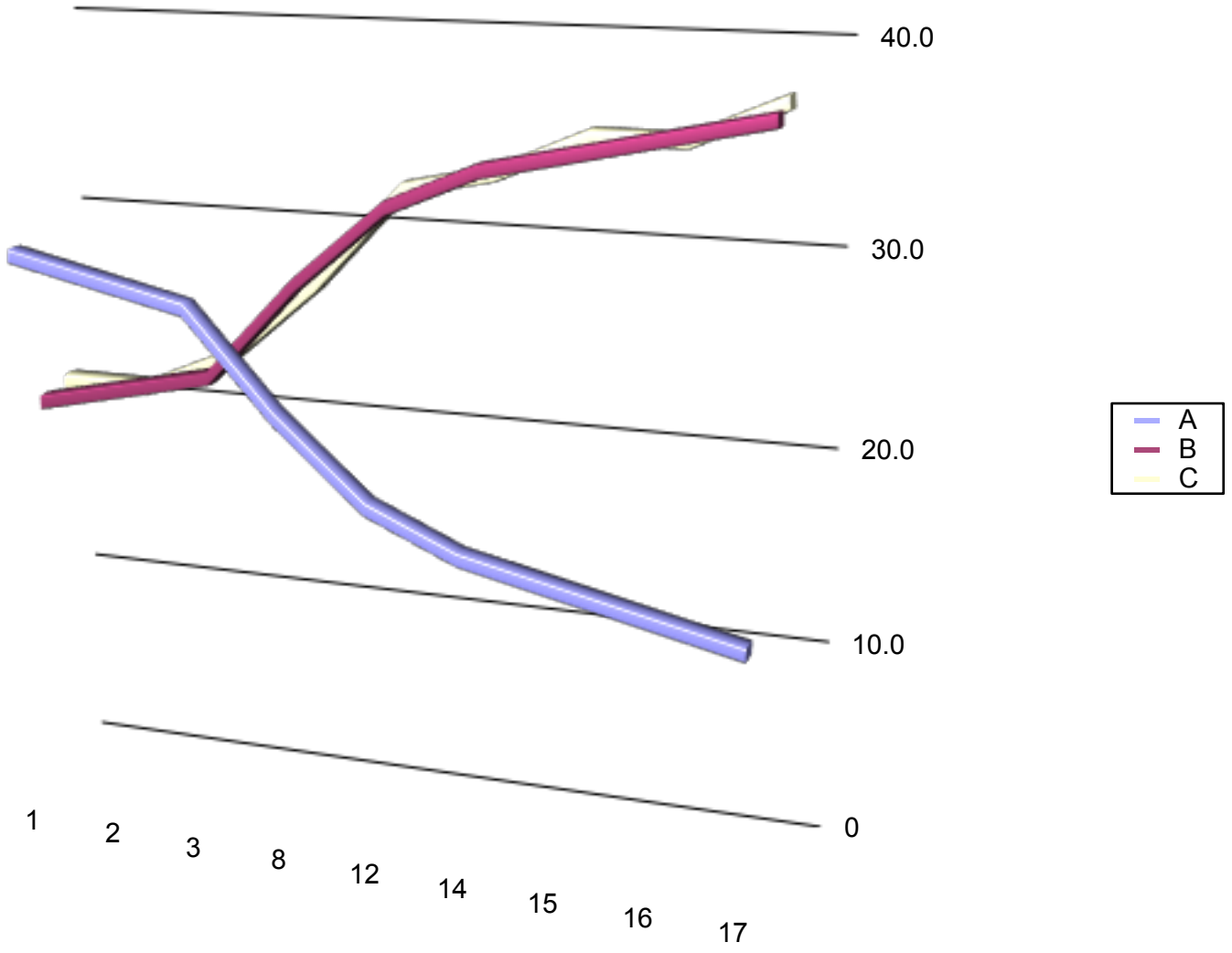
Horror Picture Show


40.0
30.0
20.0
10.0
0
1
2
3
8
12
14
15
16
17
A
B
C

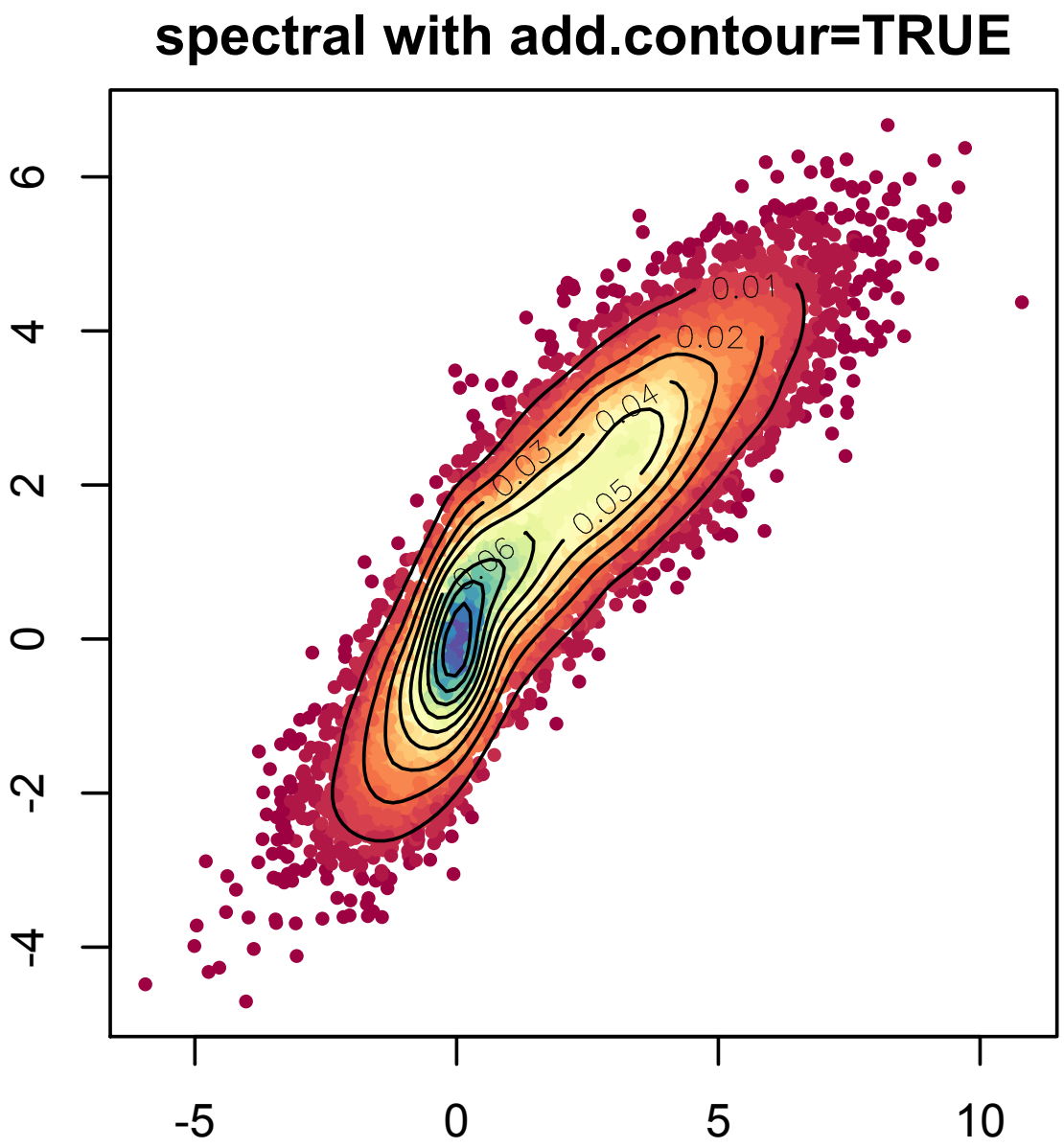

40.0
30.0
20.0
10.0
0
1
2
3
8
12
14
15
16
17
A
B
C


40.0
30.0
20.0
10.0
0
1
2
3
8
12
14
15
16
17
A
B
C


40.0
30.0
20.0
10.0
0
0
5
10
15
20
A
B
C


40.0
30.0
20.0
10.0
0
1
2
3
8
12
14
15
16
17
A
B
C

2D



function heatscatter

package LSD

Banking

Yearly sunspot numbers 1849-1924

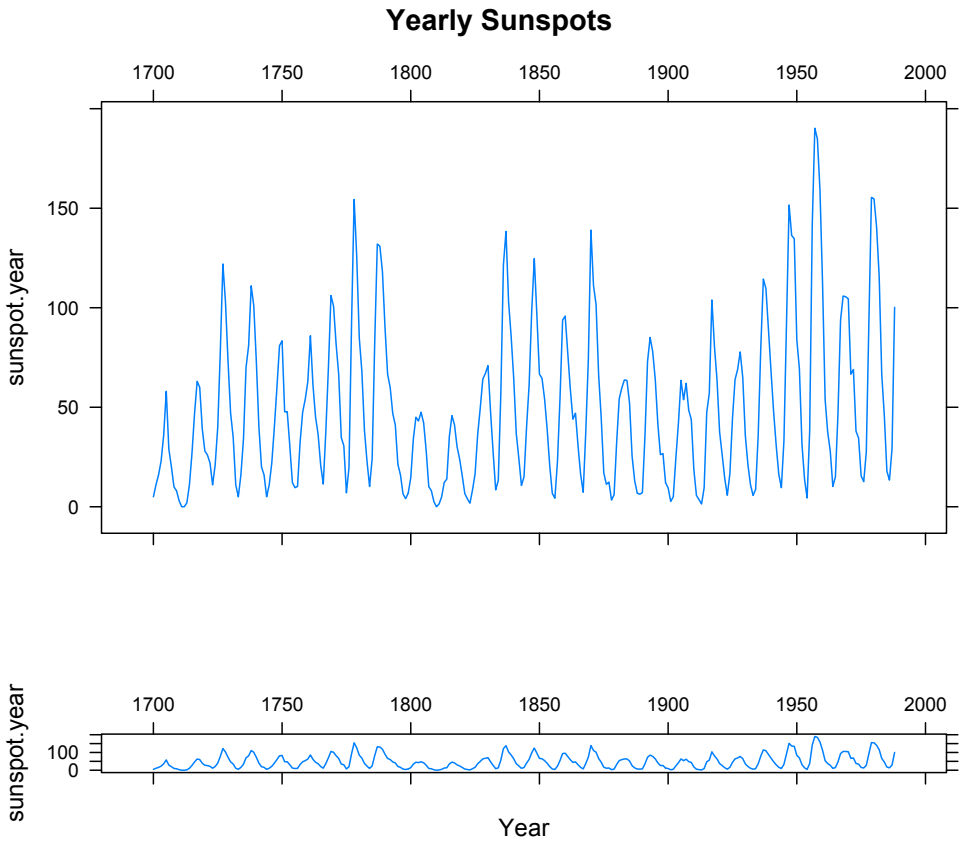
Upper panel: aspect ratio is 1.0, seems a reasonable default. But the graph fails to reveal an important property of the cycles.

Bottom panel: aspect ratio chosen by trellis algorithm *banking to 45 degrees:*

Sunspot cycles typically rise more rapidly than they fall.

This behavior is pronounced for high peaks, less pronounced for medium peaks and disappears for the lowest peaks.

Banking to 45 degrees chooses the aspect ratio to center the absolute values of the slopes of selected line segments on 45 degrees.



3D

3D

rgl package demo

3-12 D

Trellis graphics and the lattice package

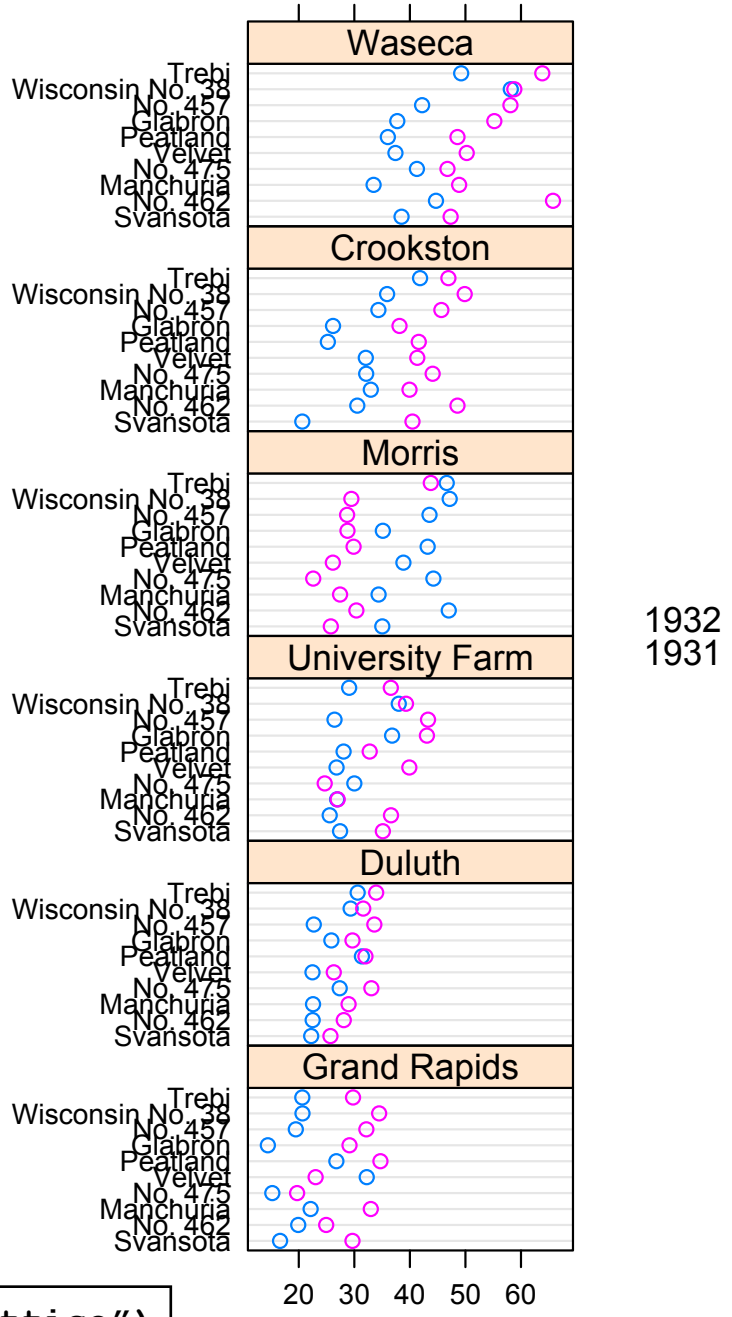
Trellis graphics

- a framework for the visualization of multivariable data. Its implementation for R is in the package lattice.
- Panels are laid out into rows, columns, and pages (reminiscent of a garden trelliswork). On each panel of the trellis, a subset of the data is graphed by a display method such as a scatterplot, curve plot, boxplot, 3-D wireframe, normal quantile plot, or dot plot. Each panel shows the relationship of certain variables conditional on the values of other variables.

Trellis

frame or structure of latticework used as a support for growing vines or plants.

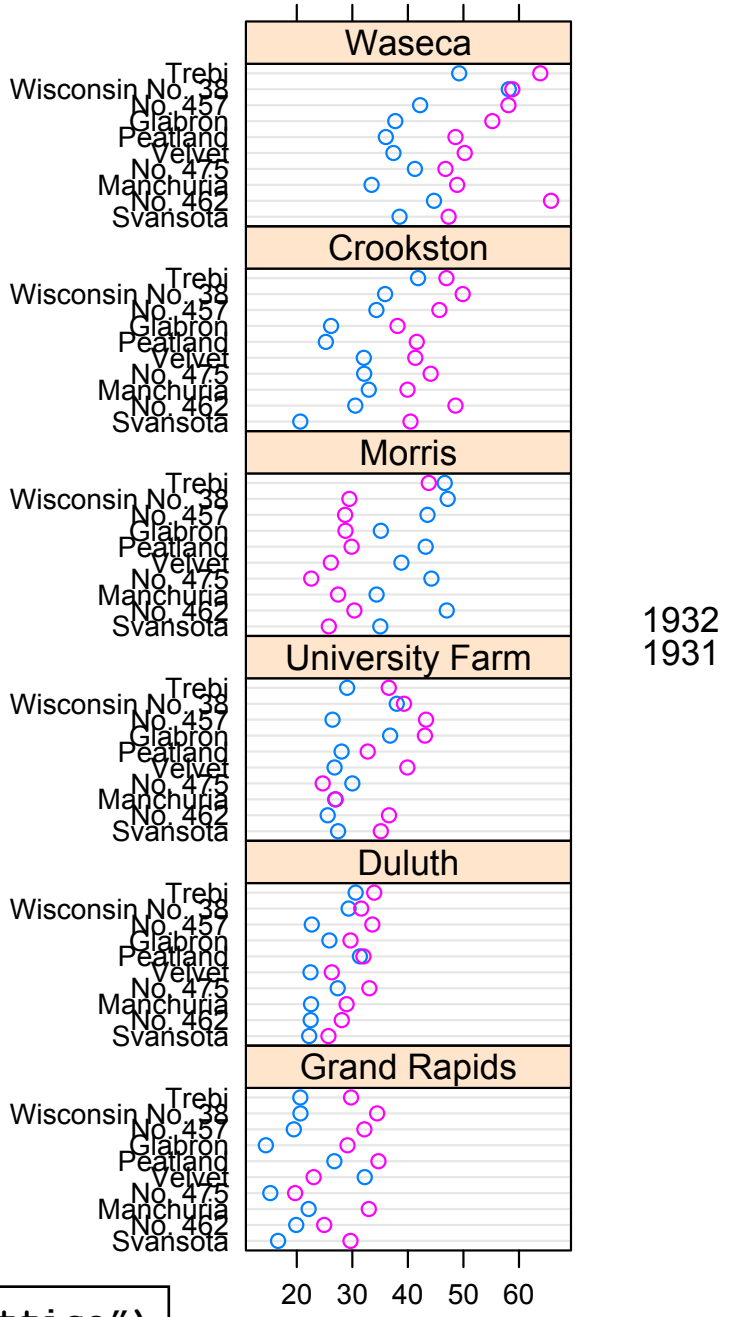
- Data from an agricultural field trial to study the crop barley.
- At six sites in Minnesota, ten varieties of barley were grown in each of two years.
- The data are the yields for all combinations of site, variety, and year, so there are 6 x 10 x 2 = 120 observations.
- Each panel in the figure displays the 20 yields at a single site.




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

- Data from an agricultural field trial to study the crop barley.
- At six sites in Minnesota, ten varieties of barley were grown in each of two years.
- The data are the yields for all combinations of site, variety, and year, so there are 6 x 10 x 2 = 120 observations.
- Each panel in the figure displays the 20 yields at a single site.

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




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

Trellis Graphics

- **Initial ideas in the 1993 book *Visualizing Data* by Bill Cleveland - for up to two conditioning variables.**
- **Extension to many explanatory variables required a new approach to conditioning, and new display technology for multipanel display.**
- **1993-1996 Rick Becker and Bill Cleveland further developed the framework.**

Trellis Graphics

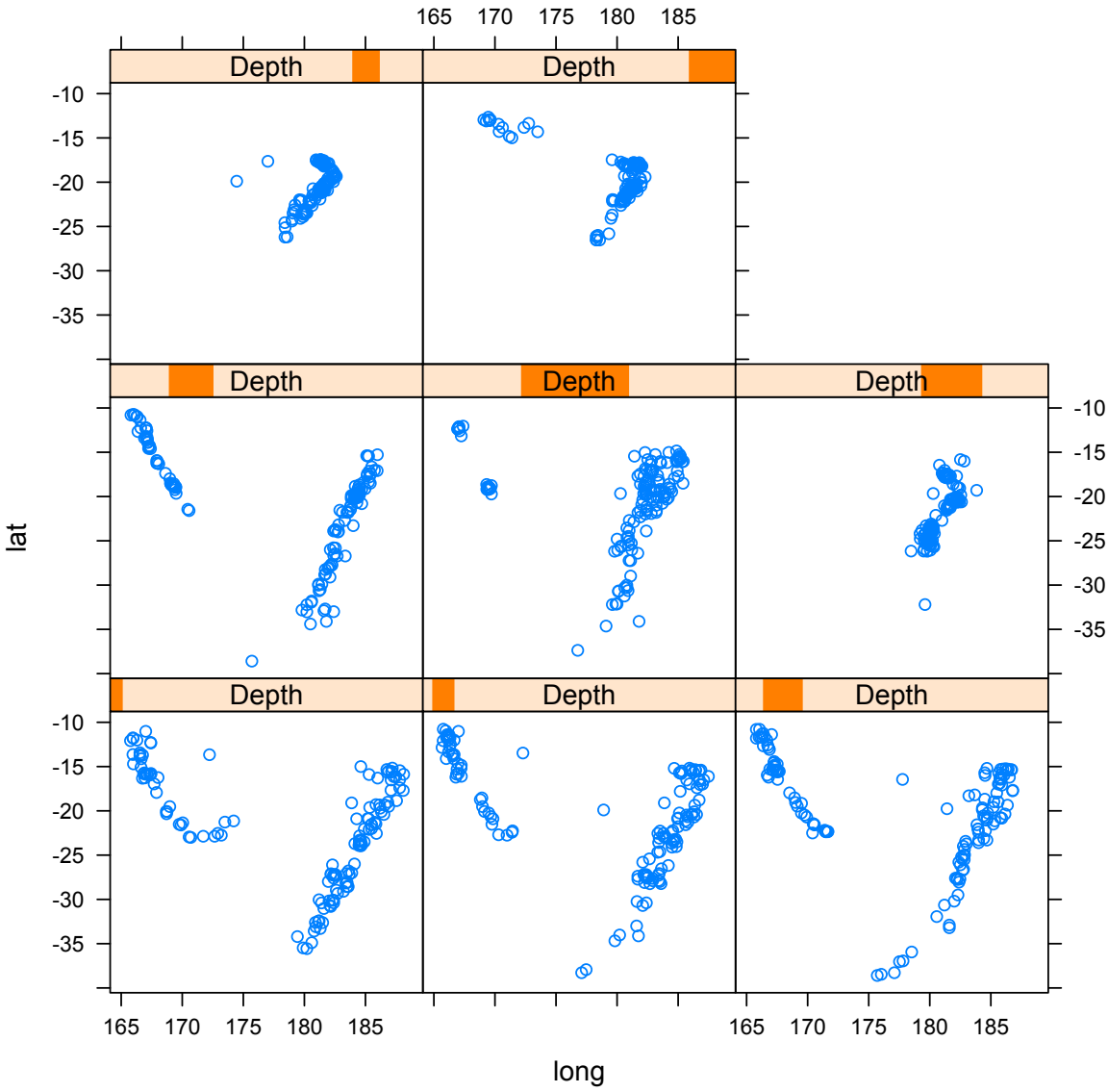
- Two primary variables are selected for display on the common axes of the panels. Conditioning variables are also selected. For example, suppose there are four variables: blood pressure, weight, sex, and race. Each panel might be a scatterplot of blood pressure (primary variables) against weight for one combination of race and sex (conditioning variables).
- Shingle: numerical variable together with a set of intervals. Allows to use it as a conditioning variable. Intervals are allowed to overlap.

Tonga Trench earthquakes

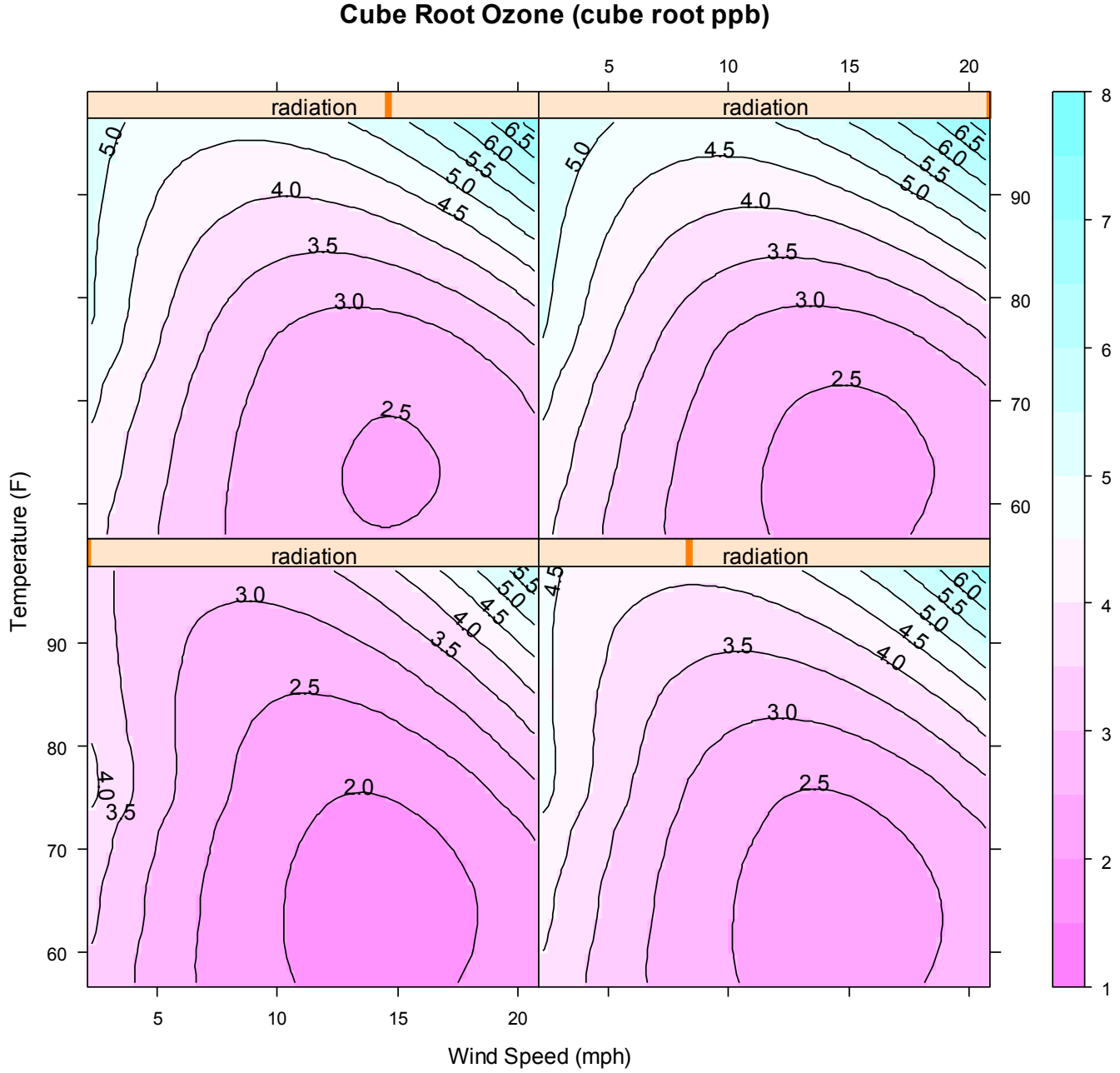
- **Depth made into a shingle and used as conditioning variable**

```
Depth =
equal.count(quakes
$depth,number=8,
overlap=.1)

xyplot(lat ~ long |
Depth, data =
quakes)
```



Levelplot (trivariate) for primaries



Irises

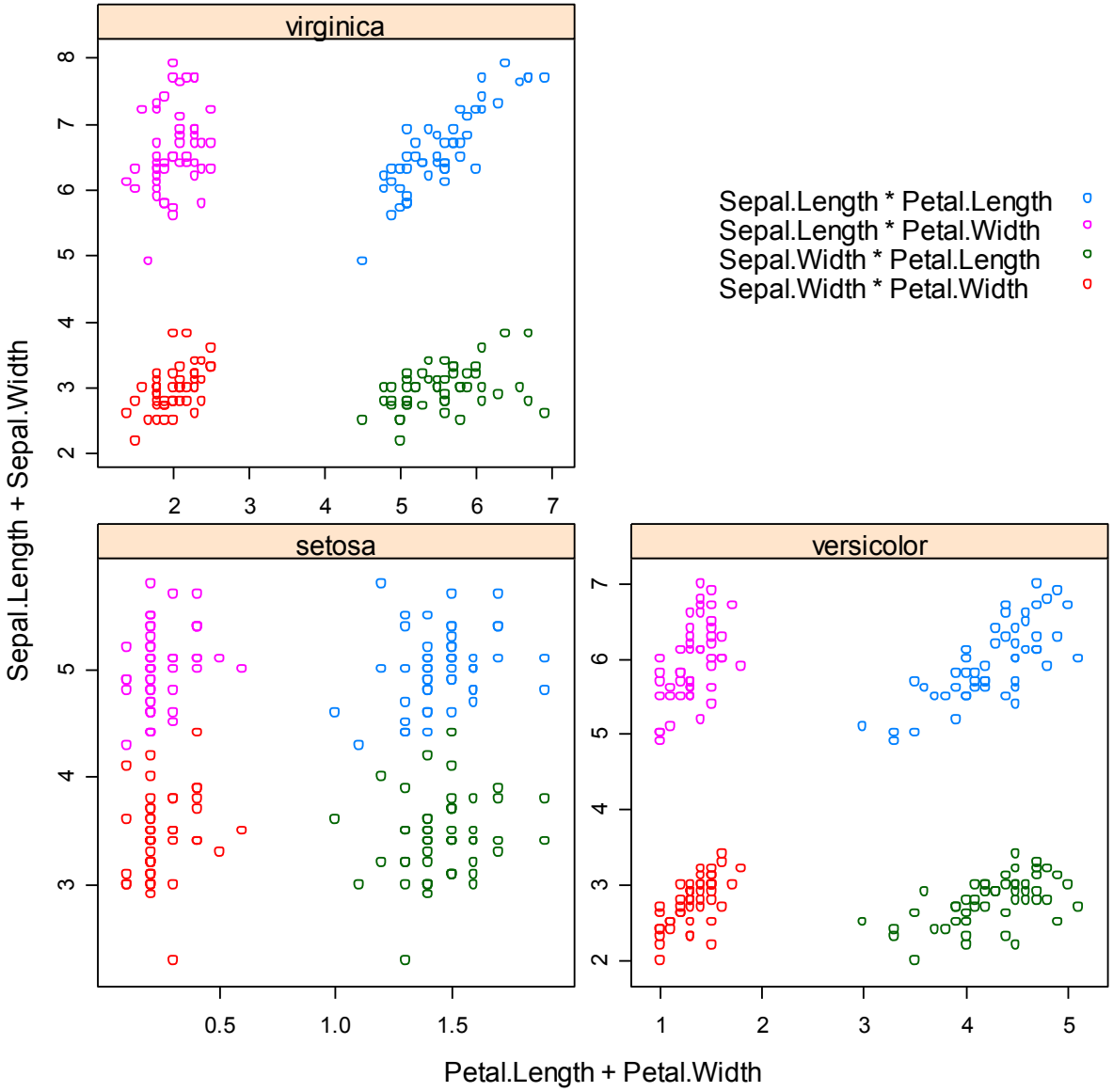
Iris virginica

Iris setosa

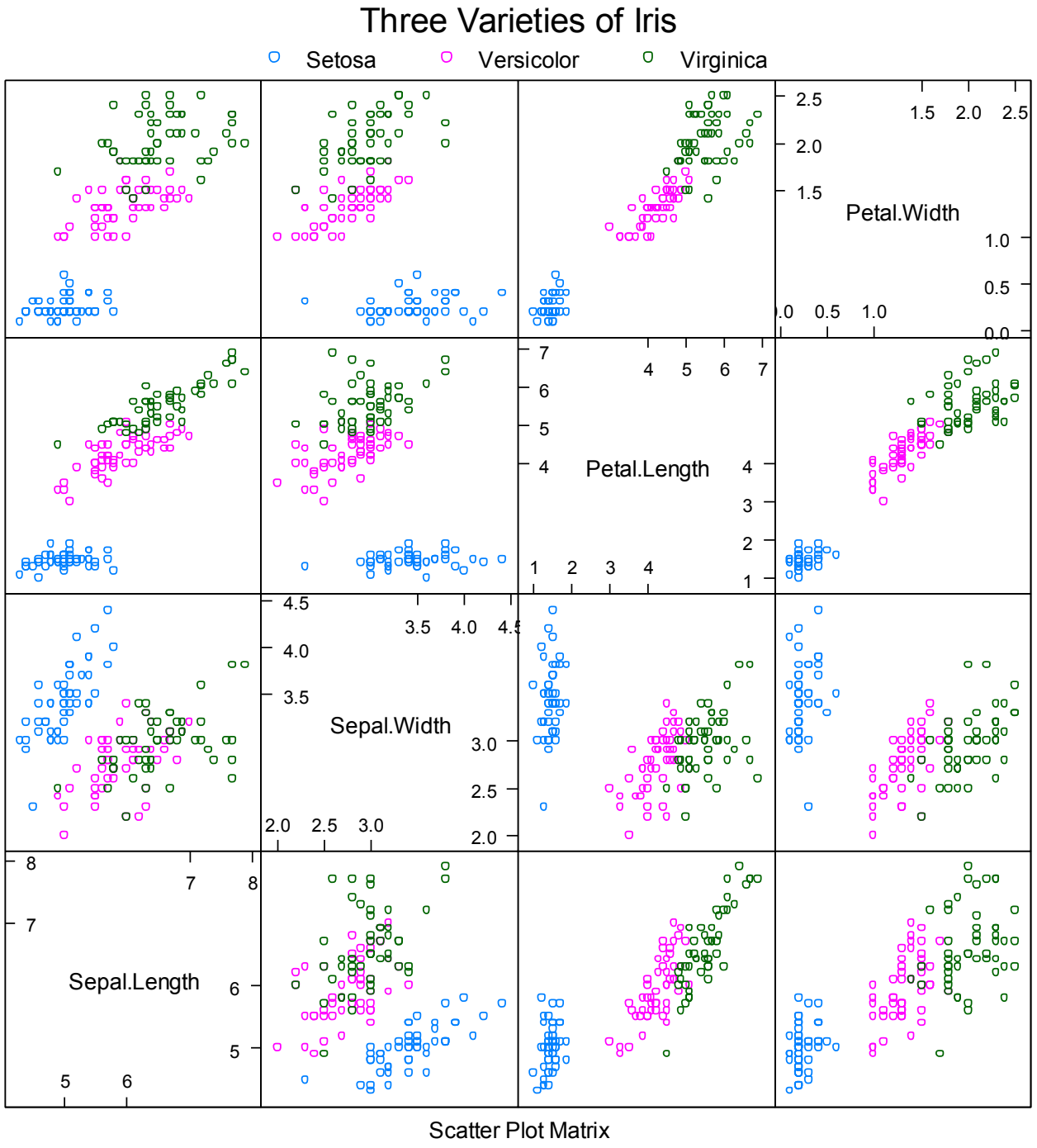
Iris versicolor

5 dimensions

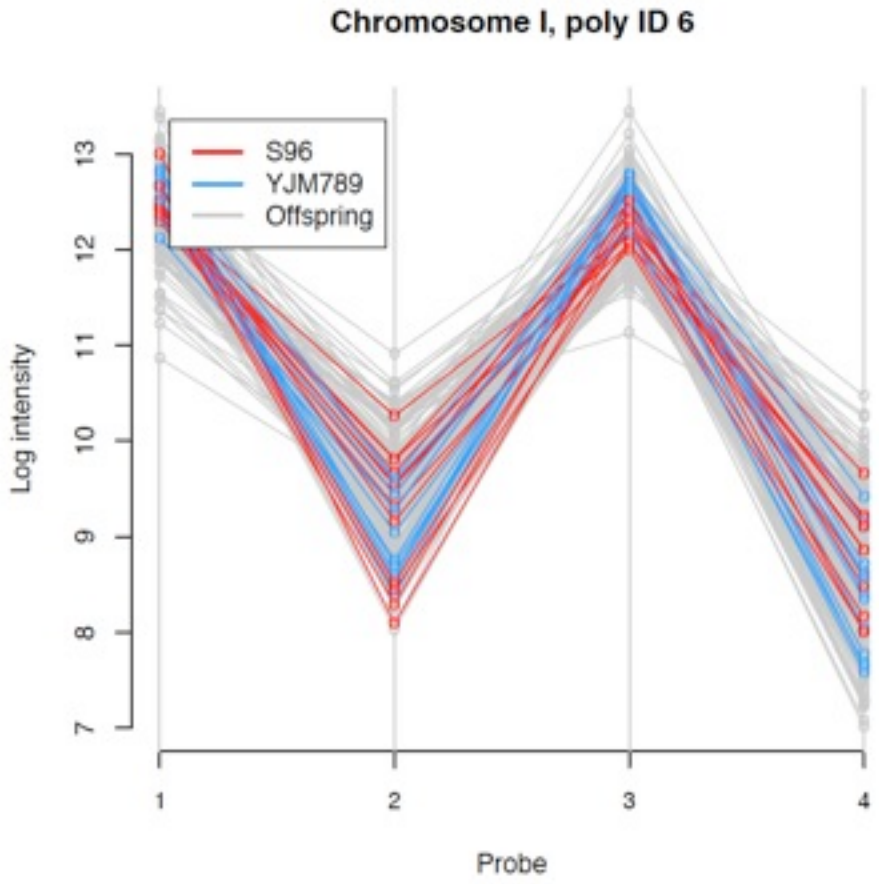
- **Iris data:**
- **sepal length and width**
- **petal length and width**
- **species**

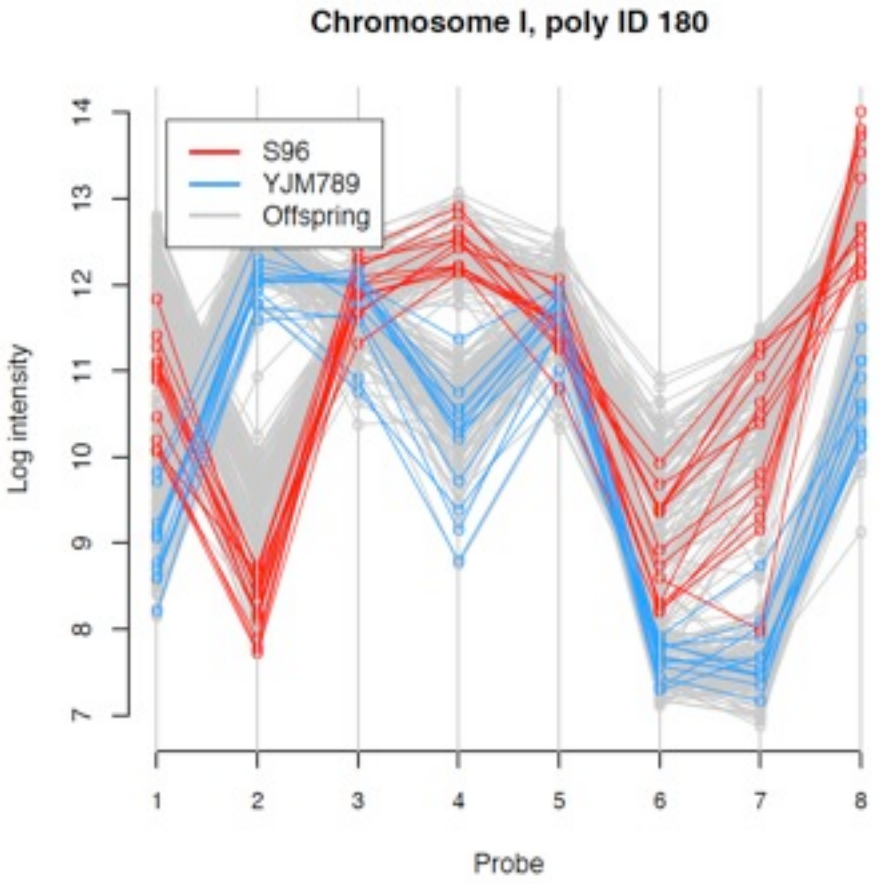


Scatterplot matrix



parallel coordinate plots





high-dimensional data

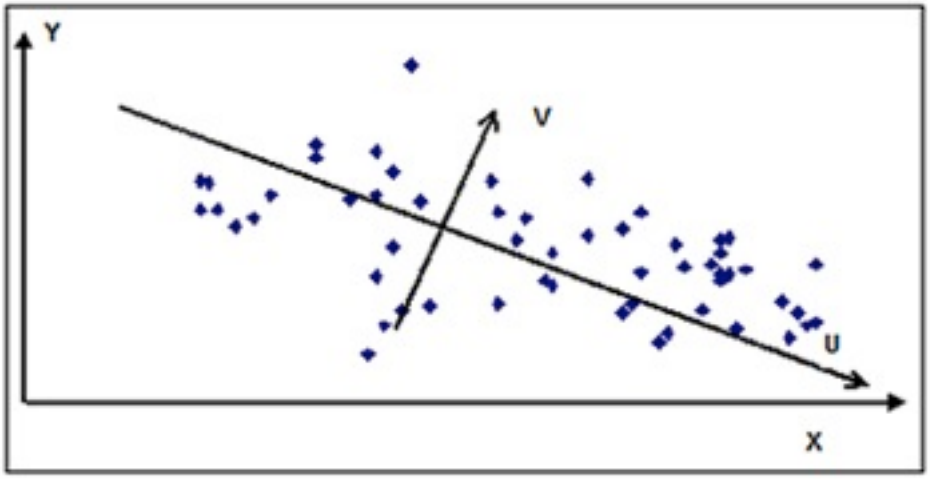
Principal Component Analysis

- **Orthogonal linear transformation of the data to a new coordinate system such that the greatest variance comes to lie on the first coordinate (first principal component), the second greatest variance on the second coordinate, and so on.**

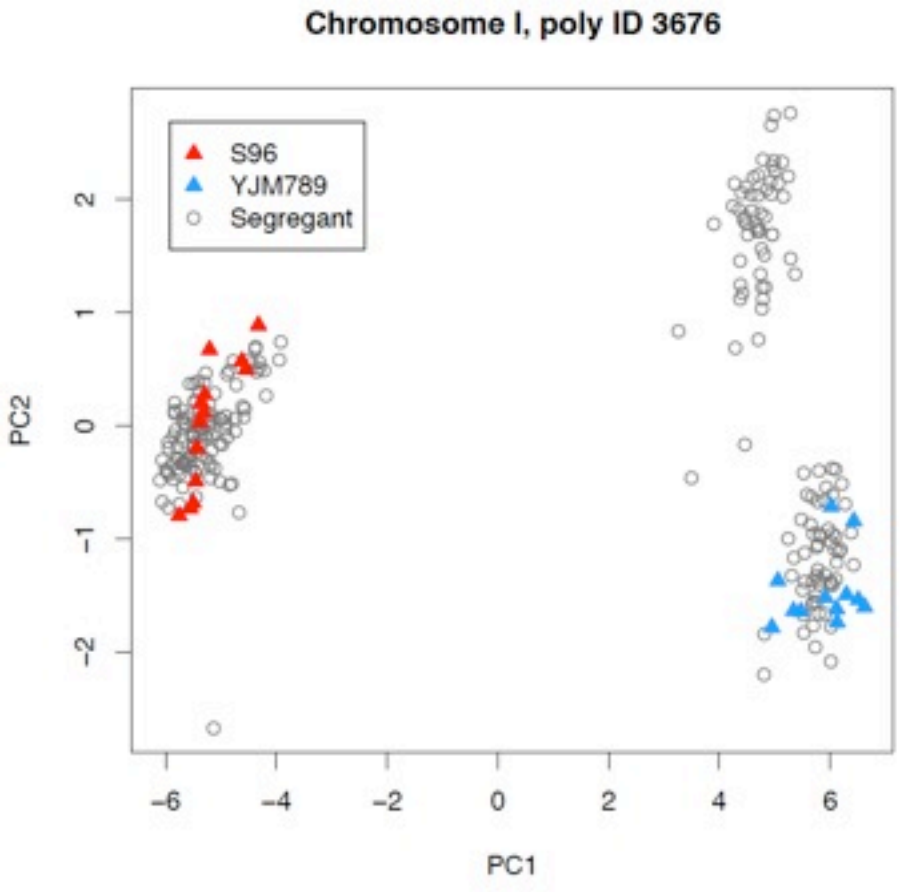
- **Principal components = Eigenvectors of covariance matrix**
- **Amount of contributed variance = Eigenvalues**

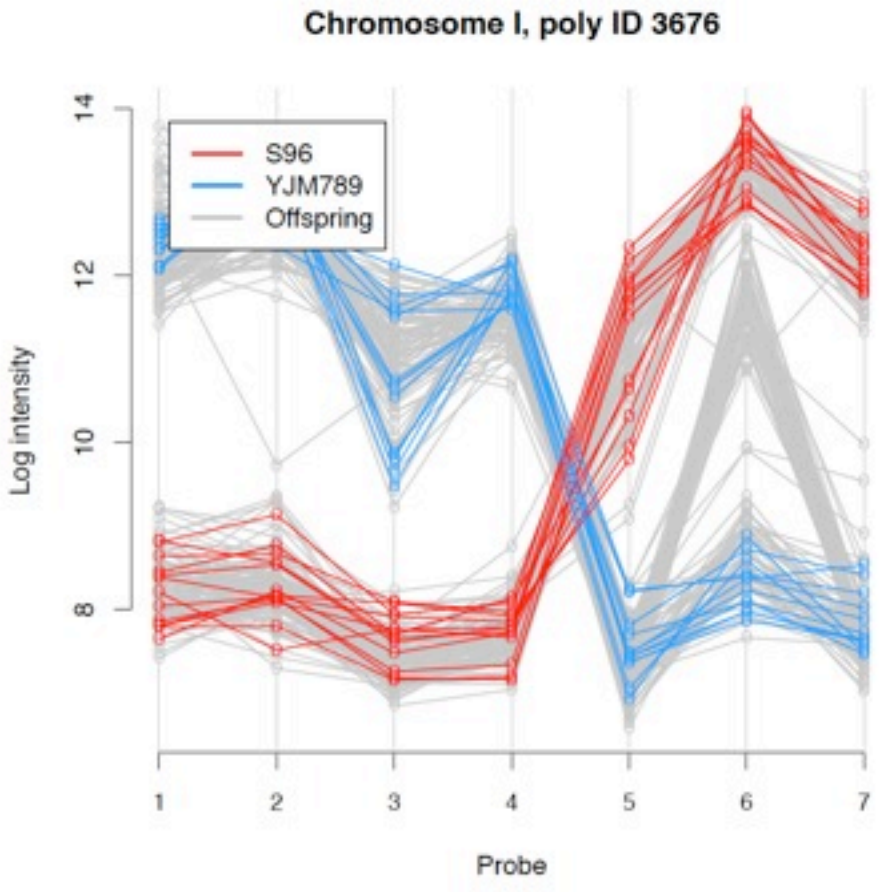
Principal Component Analysis

- **Orthogonal linear transformation of the data to a new coordinate system such that the greatest variance comes to lie on the first coordinate (first principal component), the second greatest variance on the second coordinate, and so on.**

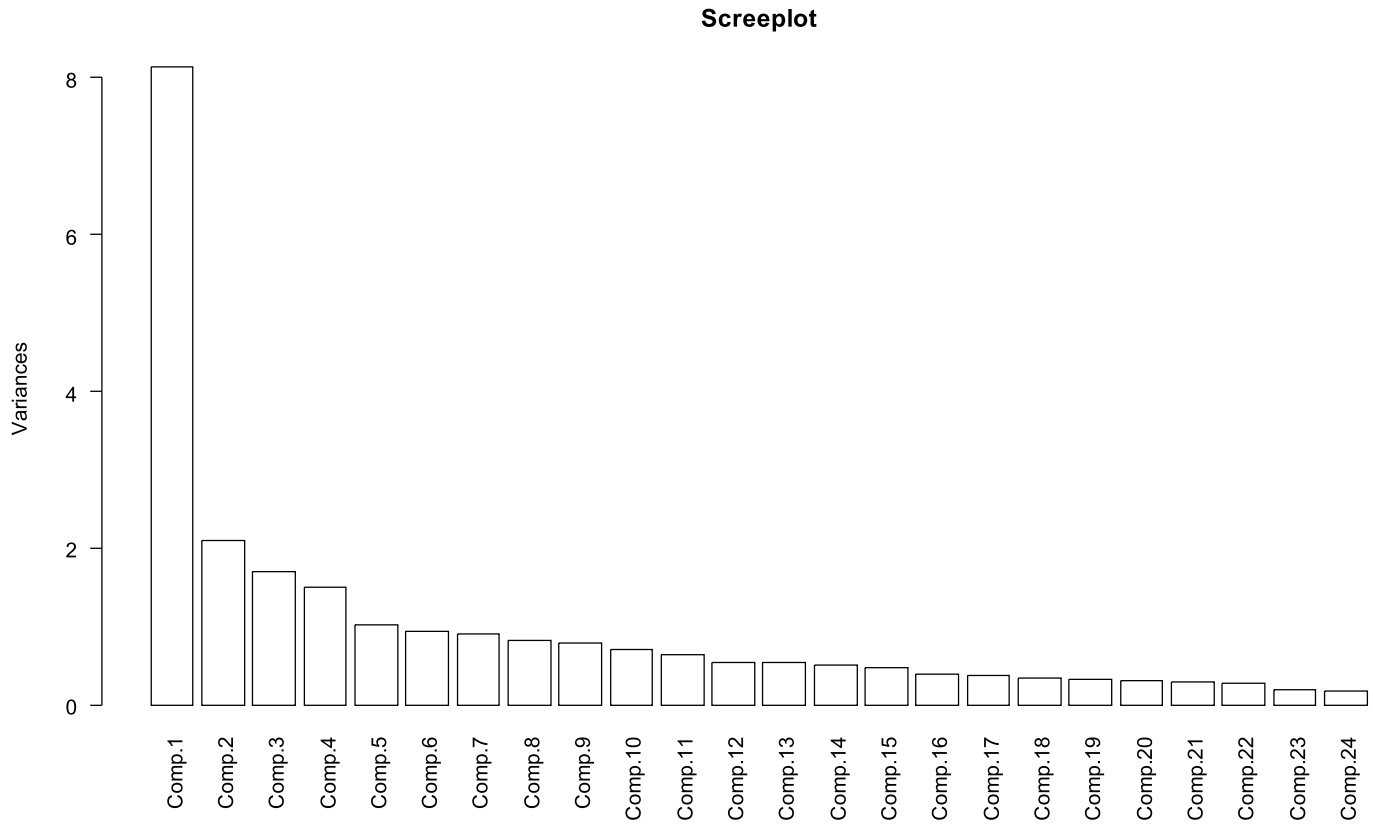


Principal component analysis





Screeplot



- `fit = princomp(covmat=Harman74.cor)`
- `sum(diag(Harman74.cor$cov))`
- `##   Trace = 24`
- `s=screeplot(fit, npcs=24, main="Screeplot", las=2)`

Non-linear low-dimensional embeddings of high-dimensional data

- **PCA is a linear method for finding a projection**
  P: $R^n \rightarrow R^d$ (e.g. $d=2$),
- **based on data $x_1,...,x_k$ *with coordinates in* R^n.**
- **Generalisations:**
  - ***P* non-linear**
  - ***k x k* distance matrix instead of coordinates**

Multidimensional scaling

- **Starting again from *k x k* distance matrix *D*, arrange points in a *d*-dimensional Euclidean space (e.g. *d=2*) such that the distances between the points are as much like the given distances as possible.**
- **Different flavors of MDS use different interpretations of "like".**
- **cmdscale: classical metric MDS uses a least-squares definition of "like." Its solution can be found by computing the eigendecomposition of a suitably defined matrix, the so-called doubly centered matrix of squared distances. A nice property of classical MDS is that the dimensions are nested, that is, the first two dimensions of the *d=2* solution are the same as the *k=2* solution.**

Multidimensional scaling

- **isoMDS minimizes the loss-function ("stress")**

$$s^2 = \min_{f \text{ monotonous}} \frac{\sum_{i \neq j} \left(f(D_{ij}) - d_{ij} \right)^2}{\sum_{i \neq j} d_{ij}^{\,2}}$$

- **where *f* is a monotonic transformation and d_{ij} are the distances between the points in the low-dimensional space.**
- **another way of saying this is that the d_{ij} are asked to preserve the *order* of the input distances D_{ij}.**

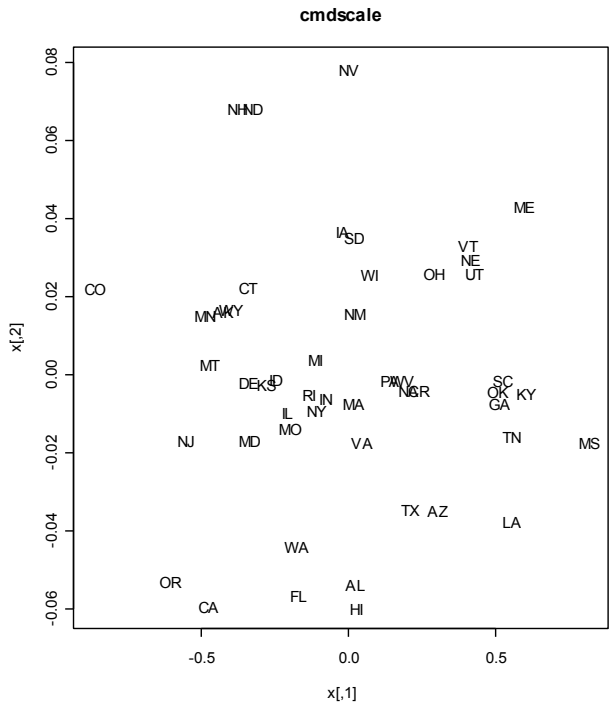
Multidimensional scaling

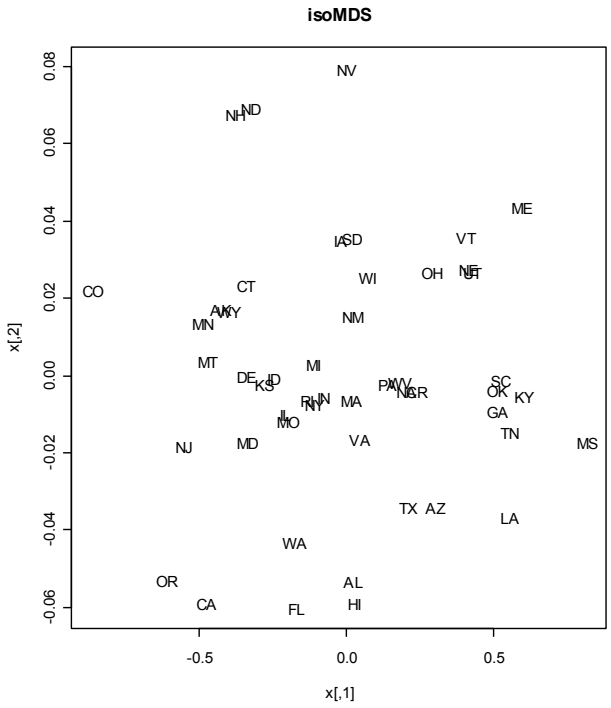
- **sammon minimizes the loss-function ("stress")**

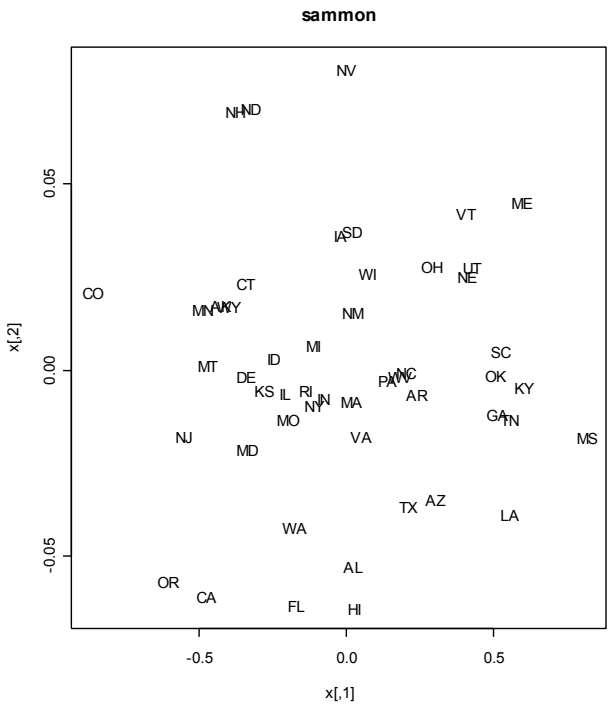
$$s^2 = \frac{\sum_{i \neq j} \frac{\left(D_{ij} - d_{ij}\right)^2}{D_{ij}}}{\sum_{i \neq j} D_{ij}}$$

- **where d_{ij} are the distances between the points in the low-dimensional space.**
- **compared classical metric MDS:**
  - **non-linear**
  - **weighting of difference terms by D_{ij} → emphasizes preservation of short distances**

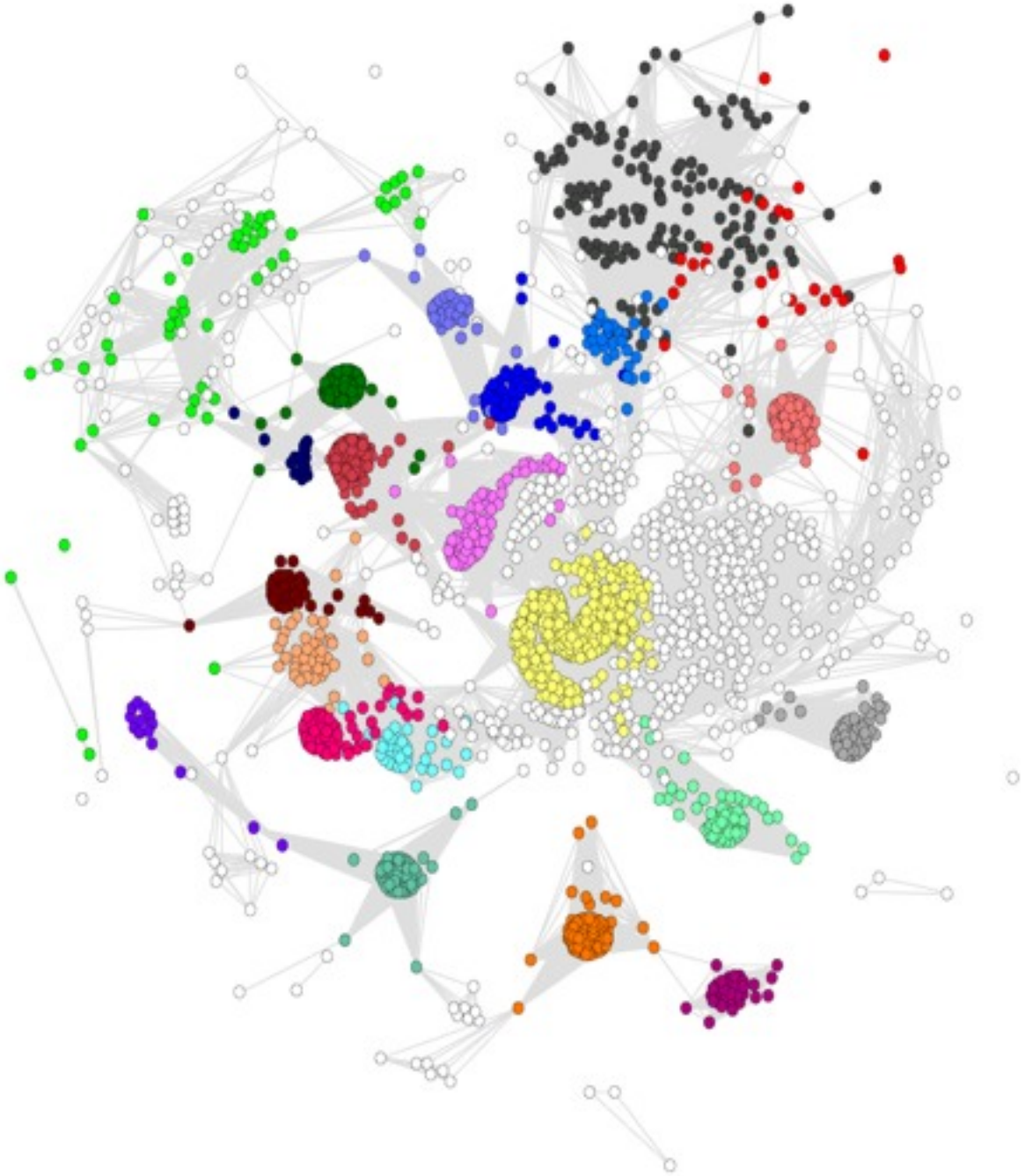
Multidimensional scaling



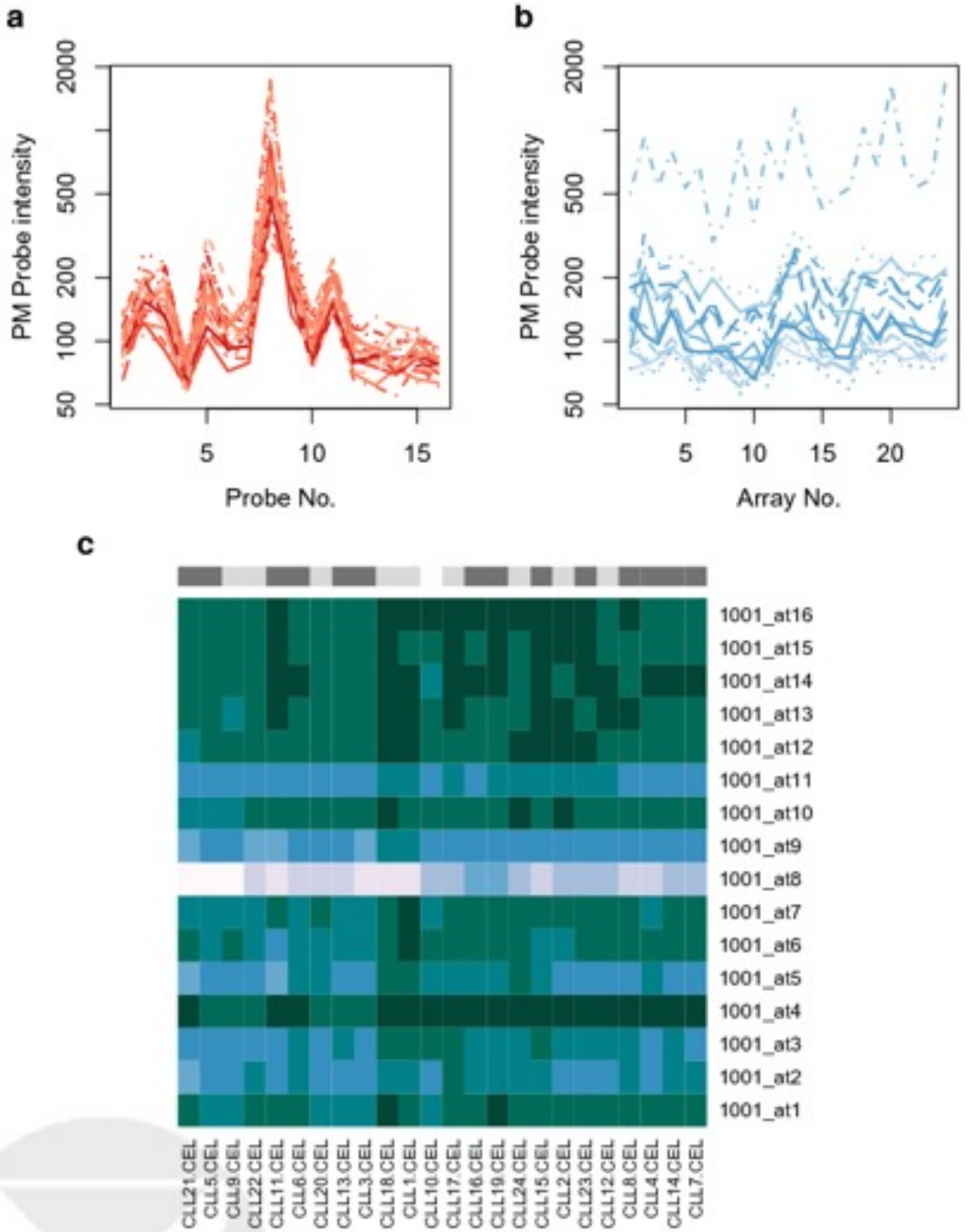




	Income	Illiteracy	Life Exp	Murder	HS Grad	Frost
Alabama	3624	2.1	69.05	15.1	41.3	20
Alaska	6315	1.5	69.31	11.3	66.7	152
Arizona	4530	1.8	70.55	7.8	58.1	15
Arkansas	3378	1.9	70.66	10.1	39.9	65
California	5114	1.1	71.71	10.3	62.6	20
Colorado	4884	0.7	72.06	6.8	63.9	166
...						

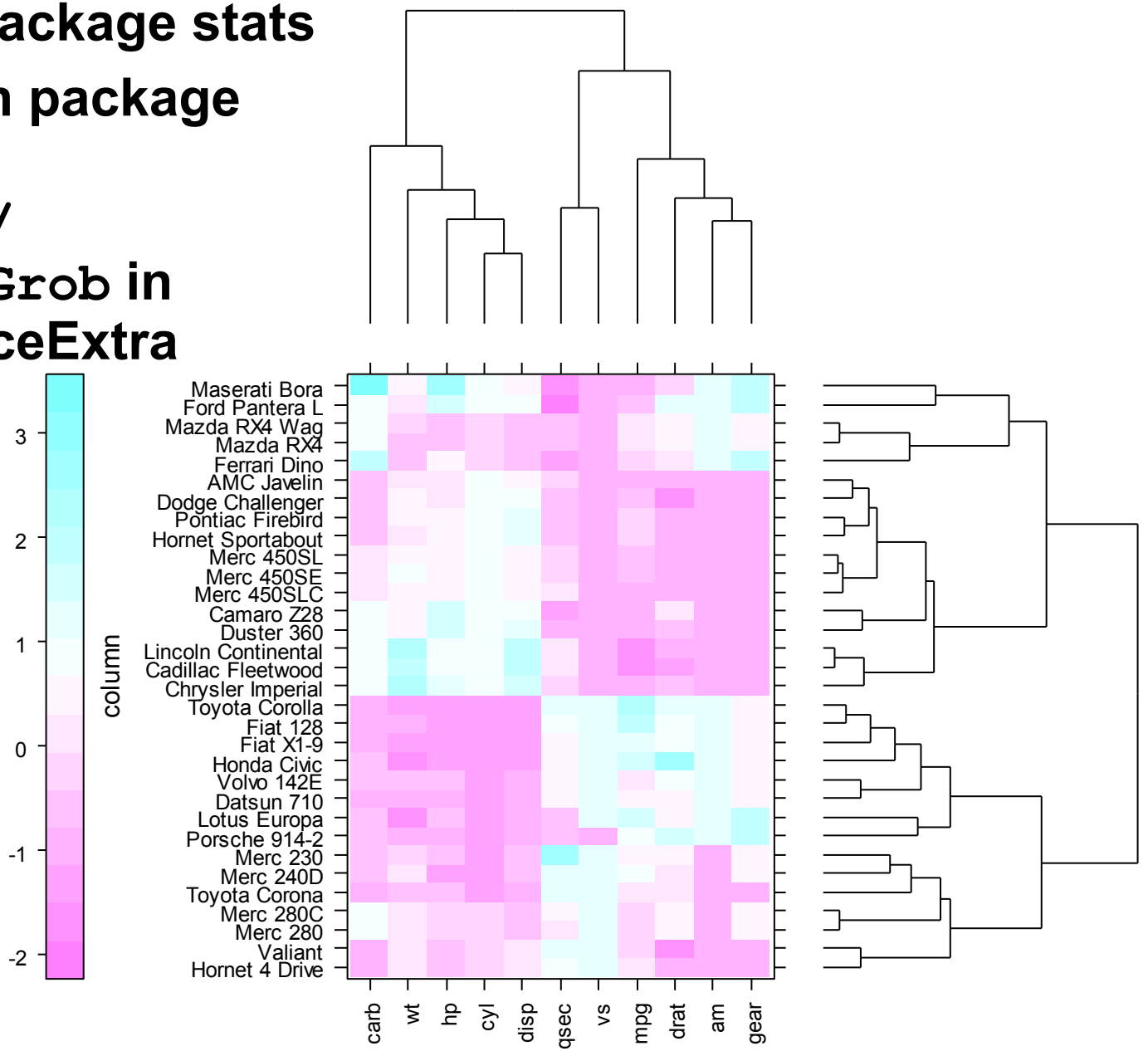


Heatmaps for matrix-like data



Software for drawing heatmaps

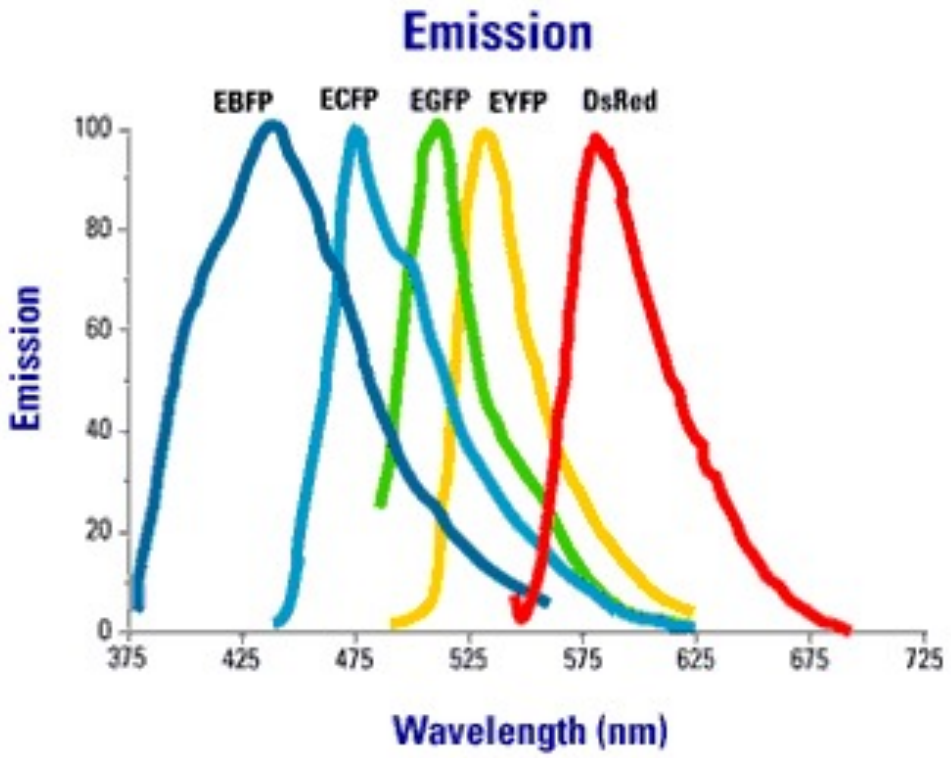
- `heatmap` in package stats
- `heatmap.2` in package gplots
- `levelplot / dendrogramGrob` in package latticeExtra



Using colours

- Different requirements for line colours than for area colours
- Avoid artefacts related to human perception
- Many people are red-green colour blind
- Lighter colours tend to make areas look larger than darker colors, thus colors of equal luminance should be chosen for graphics with large filled areas or where perception of area is important.

Light Emission Spectra



The spectral density of light waves is a function of wavelength λ.
This function space is infinite dimensional.

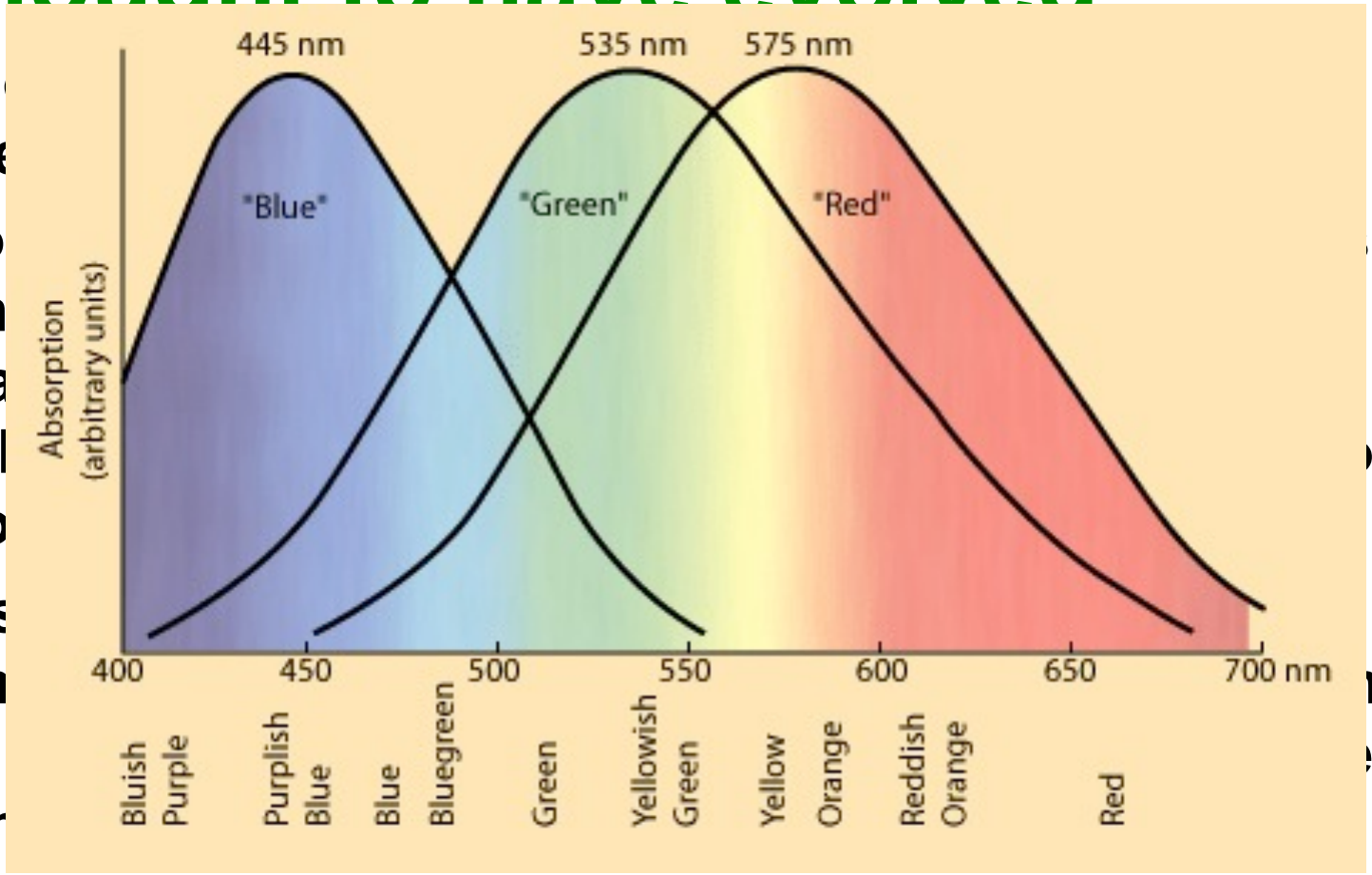
Spectrometers measure such densities on a dense sampling grid.
But our eyes are not a spectrometer.

How human colour vision is thought to have evolved

1. perception of light/dark by cone cells (monochrome; sensitive to yellow and green wavelengths)
2. Evolution (pre-mammal) of a second class of cone cells with sensitivity for blue-violet wavelengths. In combination with 1, allows to see contrasts along a "yellow/blue" axis (usually associated with our notion of warm/cold colors)
3. Primates, 30 Ma ago: specification of the yellow/green cones into two classes: one more sensitive to green, one more to red, allowing to see contrasts in that part of the spectrum (helpful for assessing the ripeness of fruit)
4. Although the space of all possible wavelength spectra is infinite-dimensional, we perceive them as a 3-dimensional signal

How human colour vision is thought to have evolved

1. percepti
   sensitive
2. Evolutio
   with sen
   combina
   "yellow/l
   warm/co
3. Primates
   cones in
   more to
   spectrur
4. Although the space of all possible wavelength spectra is infinite-dimensional, we perceive them as a 3-dimensional signal



letters to nature

Nature **323**, 623 - 625 (16 October 1986); doi:10.1038/323623a0

Polymorphism of the long-wavelength cone in normal human colour vision

JAY NEITZ & GERALD H. JACOBS

Department of Psychology, University of California, Santa Barbara, California 93106, USA

Colour vision is based on the presence of multiple classes of cone each of which contains a different type of photopigment[1]. Colour matching tests have long revealed that the normal human has three cone types. Results from these tests have also been used to provide estimates of cone spectral sensitivities[2]. There are significant variations in colour matches made by individuals whose colour vision is classified as normal[3–6]. Some of this is due to individual differences in preretinal absorption and photopigment density, but some is also believed to arise because there is variation in the spectral positioning of the cone pigments among those who have normal colour vision. We have used a sensitive colour matching test to examine the magnitude and nature of this individual variation and here report evidence for the existence of two different long-wavelength cone mechanisms in normal humans. The different patterns of colour matches made by male and female subjects indicate these two mechanisms are inherited as an X-chromosome linked trait.

letters to nature

Nature **323**, 623 - 625 (16 October 1986); doi:10.1038/323623a0

Polymorphism of the long-wavelength cone in normal human colour vision

JAY NEITZ & GERALD H. JACOBS

Department of Psychology, University of California, Santa Barbara, California 93106, USA

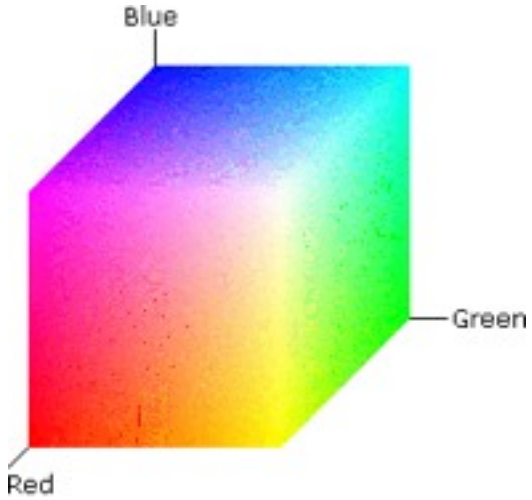
Colour vision is based on the presence of multiple classes of cone each of which contains a different type of photopigment[1]. Colour matching tests have long revealed that the normal human has three cone types. Results from these tests have also been used to provide estimates of cone spectral sensitivities[2]. There are significant variations in colour matches made by individuals whose colour vision is classified as normal[3–6]. Some of this is ...tion and photopigment density, but some is also believed to ...ositioning of the cone pigments among those who have normal ...atching test to examine the magnitude and nature of this ... the existence of two different long-wavelength cone mechanisms in normal humans. The different patterns of colour matches made by male and female subjects indicate these two mechanisms are inherited as an X-chromosome linked trait.

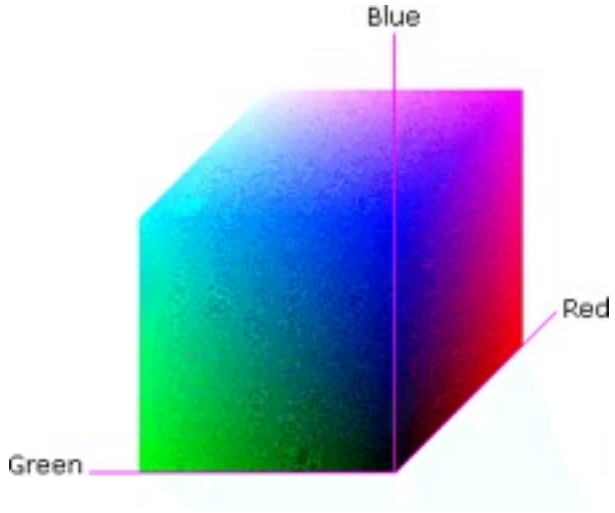
Note: genes for the red and green receptors are on the X-chromosome


VAIO

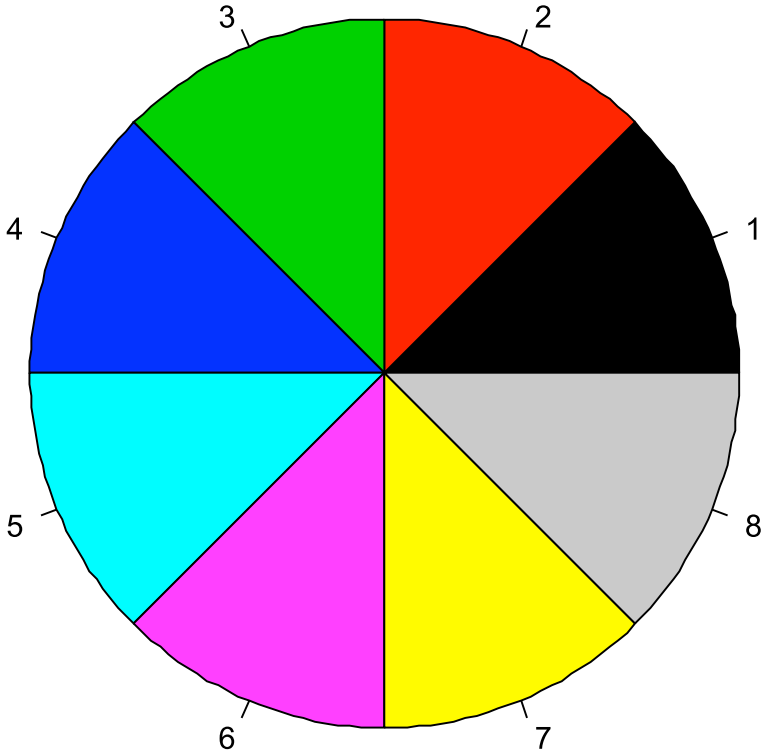
RGB color space

- **Motivated by computer screen hardware**





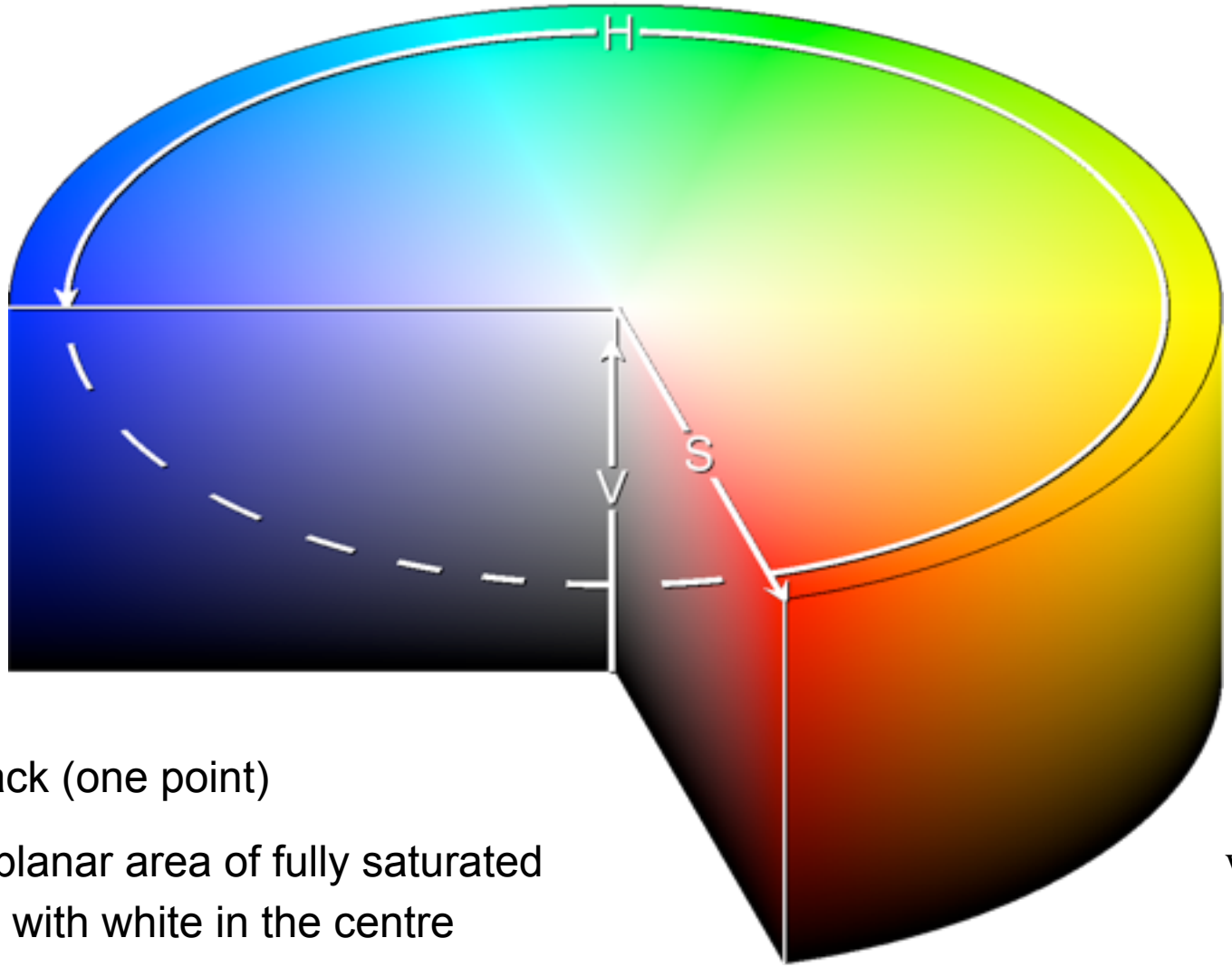
Color palettes based on the extremes of the RGB cube hurt the eyes




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

HSV color space

- Hue-Saturation-Value (Smith 1978)

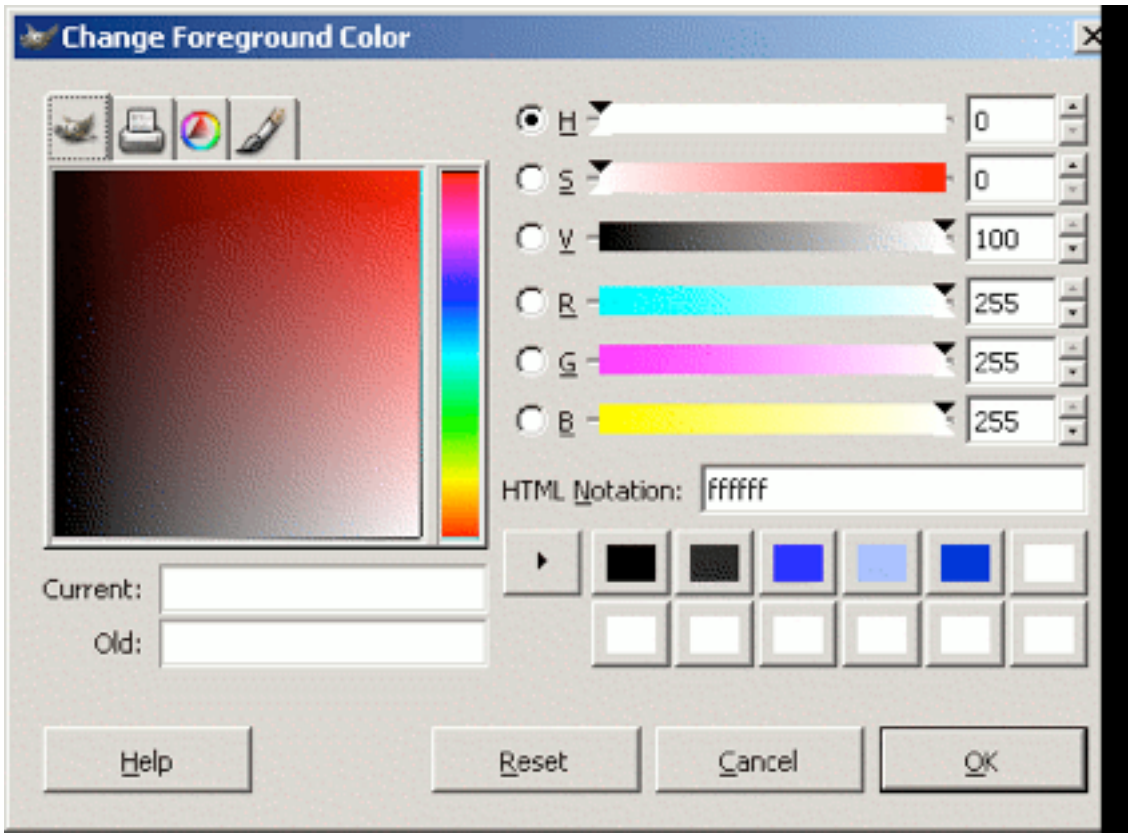


V_{min}: black (one point)

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:

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

$$v = \max$$

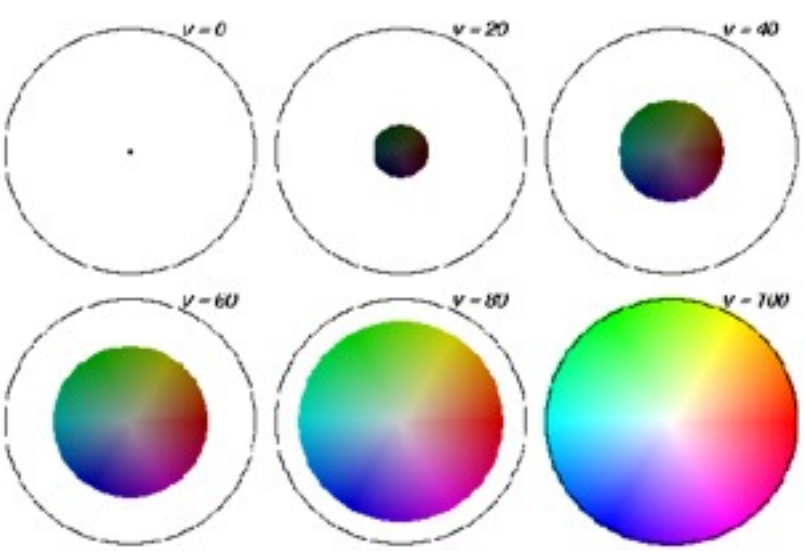
The range of HSV and HSL vectors is a cube in the cartesian coordinate system; but since hue is really a cyclic property, with a cut at red, visualizations of these spaces invariably involve hue circles;[4] cylindrical and conical (bi-conical for HSL) depictions are most popular; Spherical depictions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

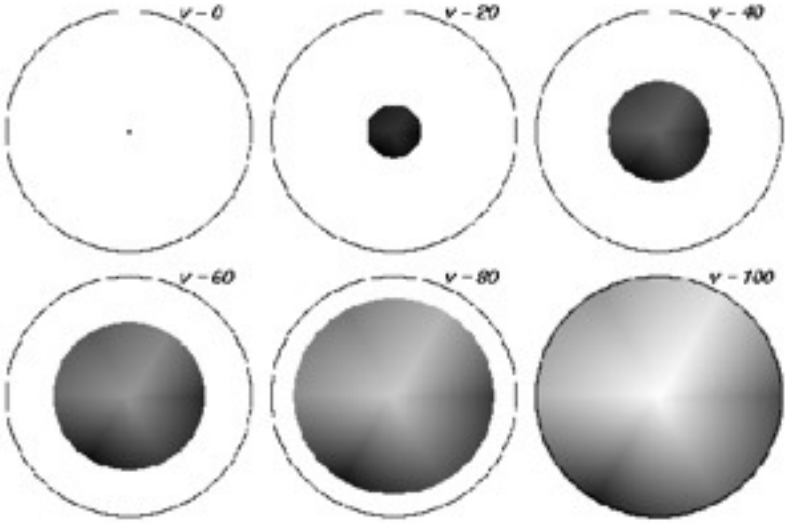
perceptual colour spaces

- However, 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 (colorfulness, intensity of color as compared to gray)
  - 3. luminance (brightness, amount of gray)

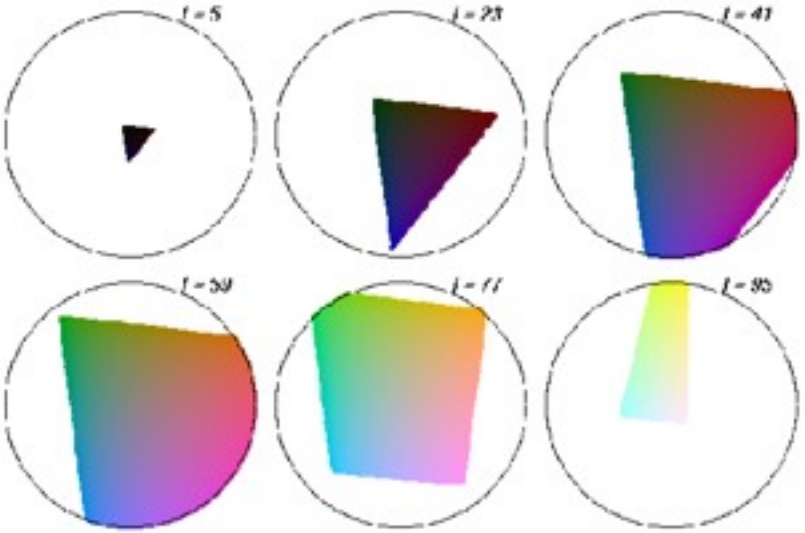
HCL colour coordinates: L is a more useful parameter of brightness

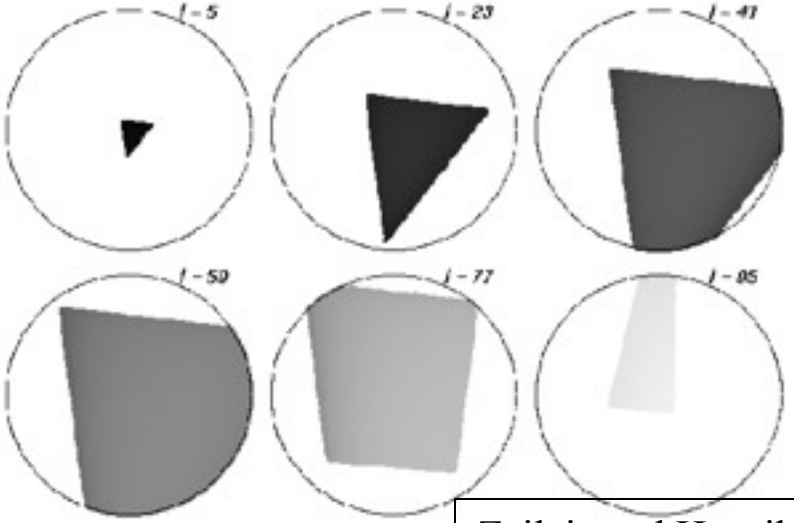
HSV





HCL





Zeileis and Hornik

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)

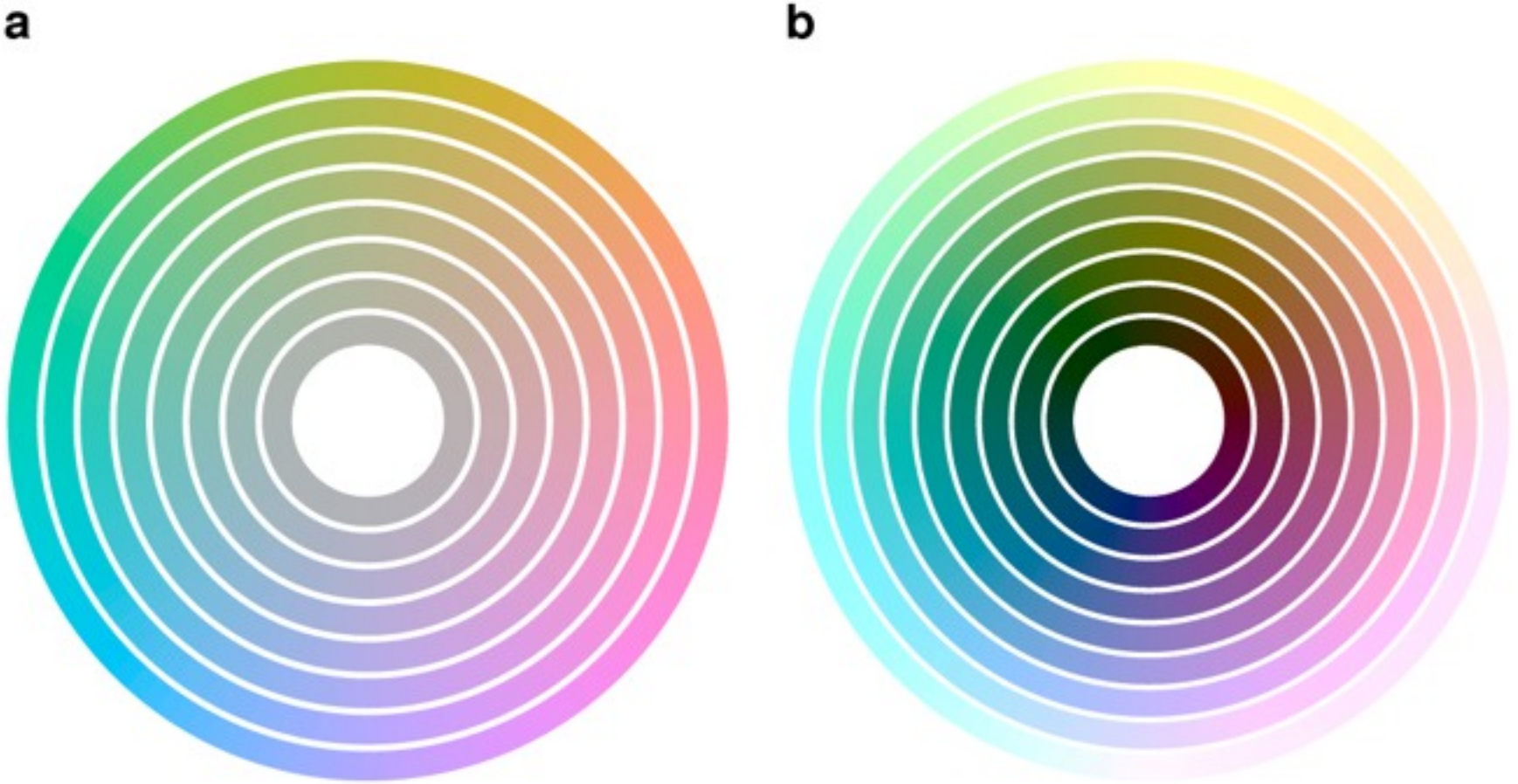
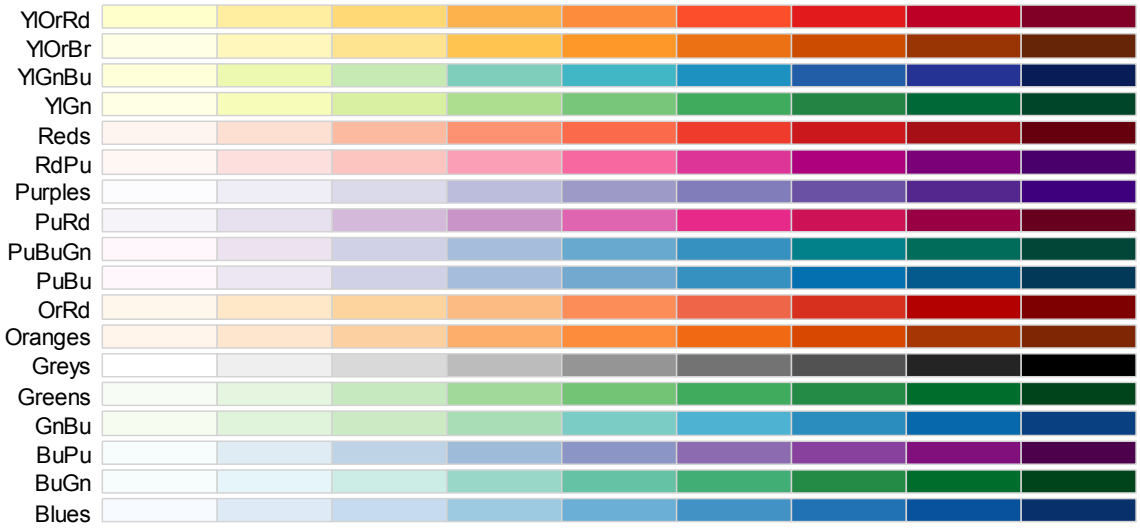


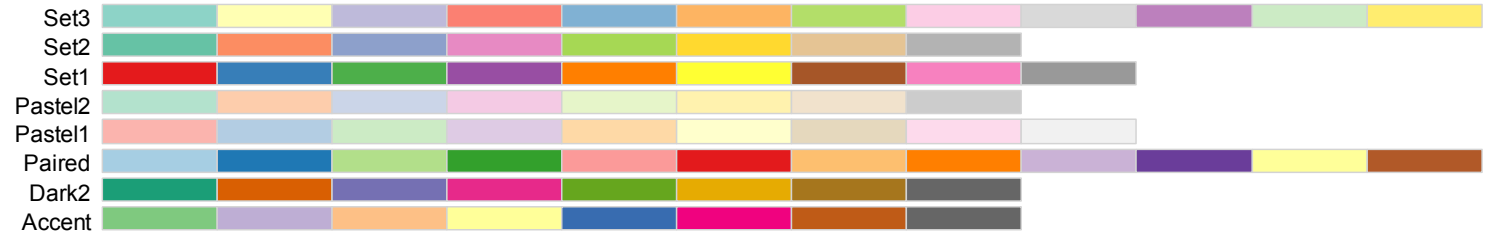
Figure 2: Circles in HCL colorspace. *a*: circles in HCL space at constant $L = 75$, with the angular coordinate H varying from 0 to 360 and the radial coordinate $C = 0, 10, \ldots, 60$. *b*: constant $C = 50$, and $L = 10, 20, \ldots, 90$.

Software

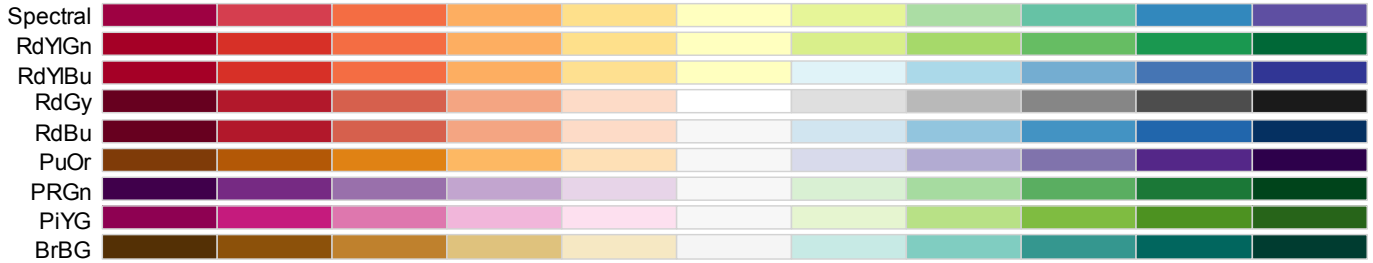
sequential



qualitative

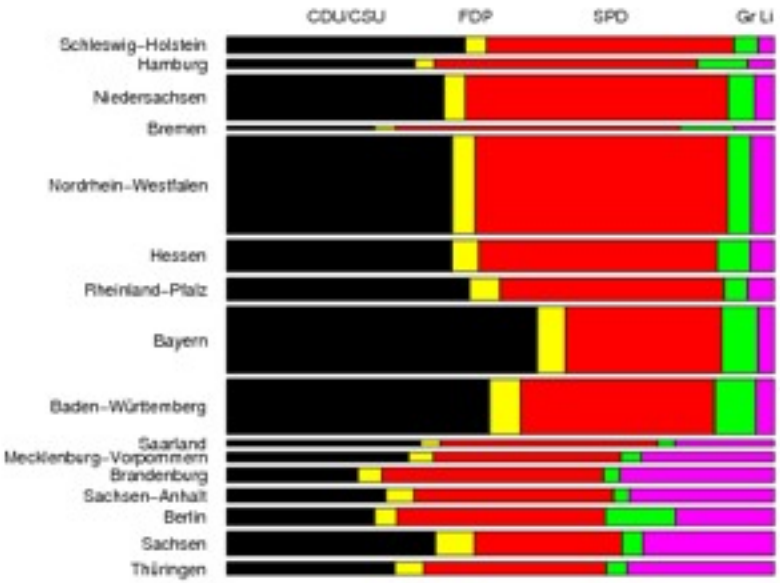


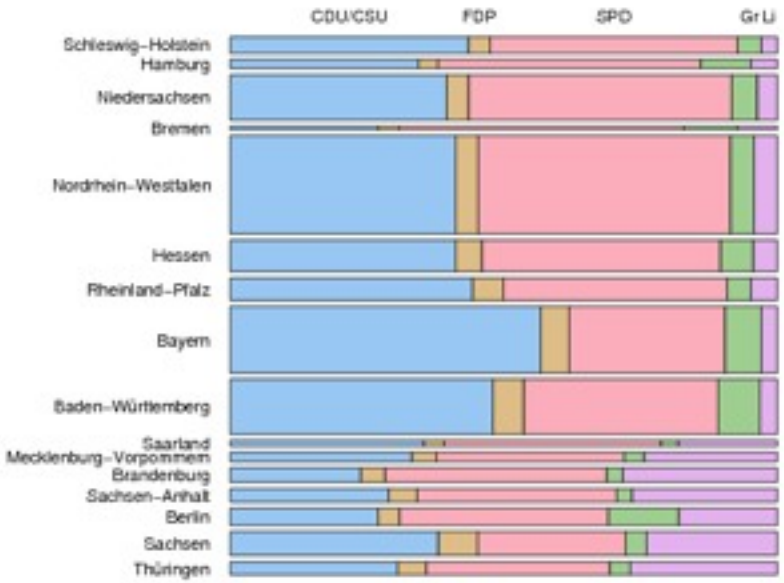
diverging



RColorBrewer and **vcd** packages

Pick your favourite





From A. Zeileis, Reisensburg 2007

Some useful functions for working with colours

- RColorBrewer
- `display.brewer.all` show all palettes
- `brewer.pal` choose one particular palette

- RColorBrewer
- `colorRamp, colorRampPalette` interpolate

- vcd
- `sequential_hcl, diverge_hcl, rainbow_hcl` palettes

- ... and avoid R's default colours

Acknowledgement

- Robert Gentleman
- Florian Hahne
- Steffen Durinck
- Greg Pau

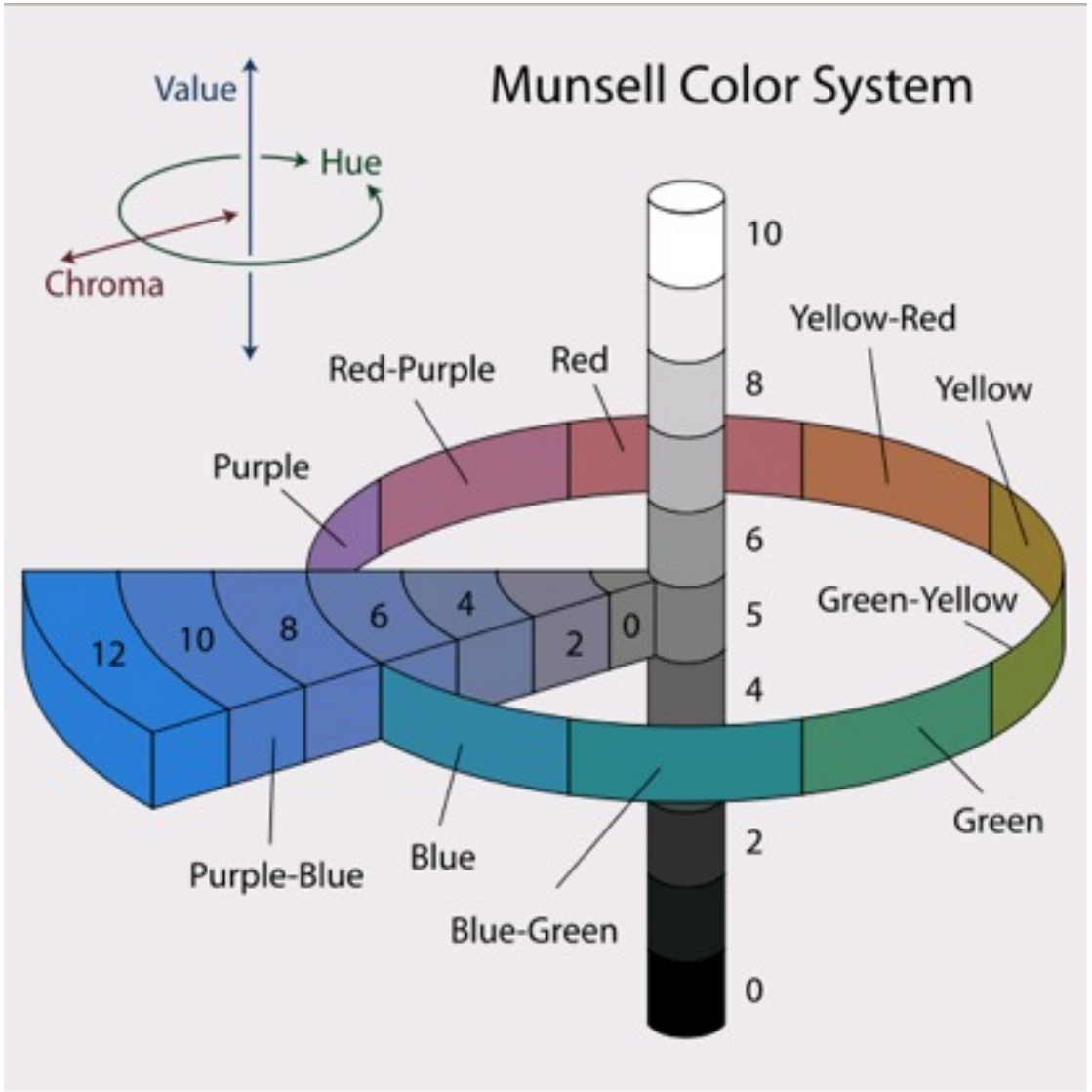
References

- 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

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.

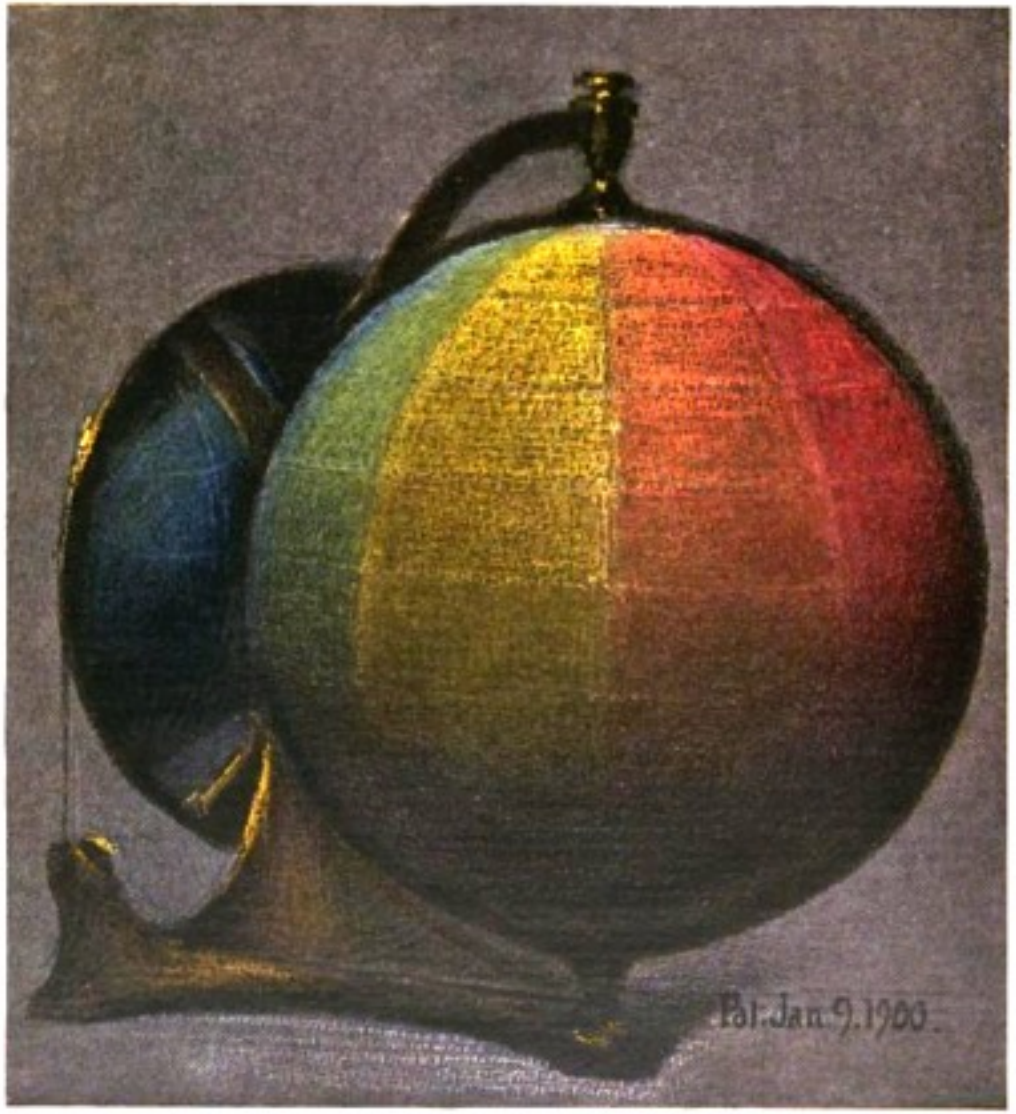


Munsell Colour System

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 much 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.**