Recent developments in direct labeled graphics

http://directlabels.r-forge.r-project.org

Toby Dylan Hocking
toby.hocking@inria.fr

29 March 2012

Motivation: confusing legends

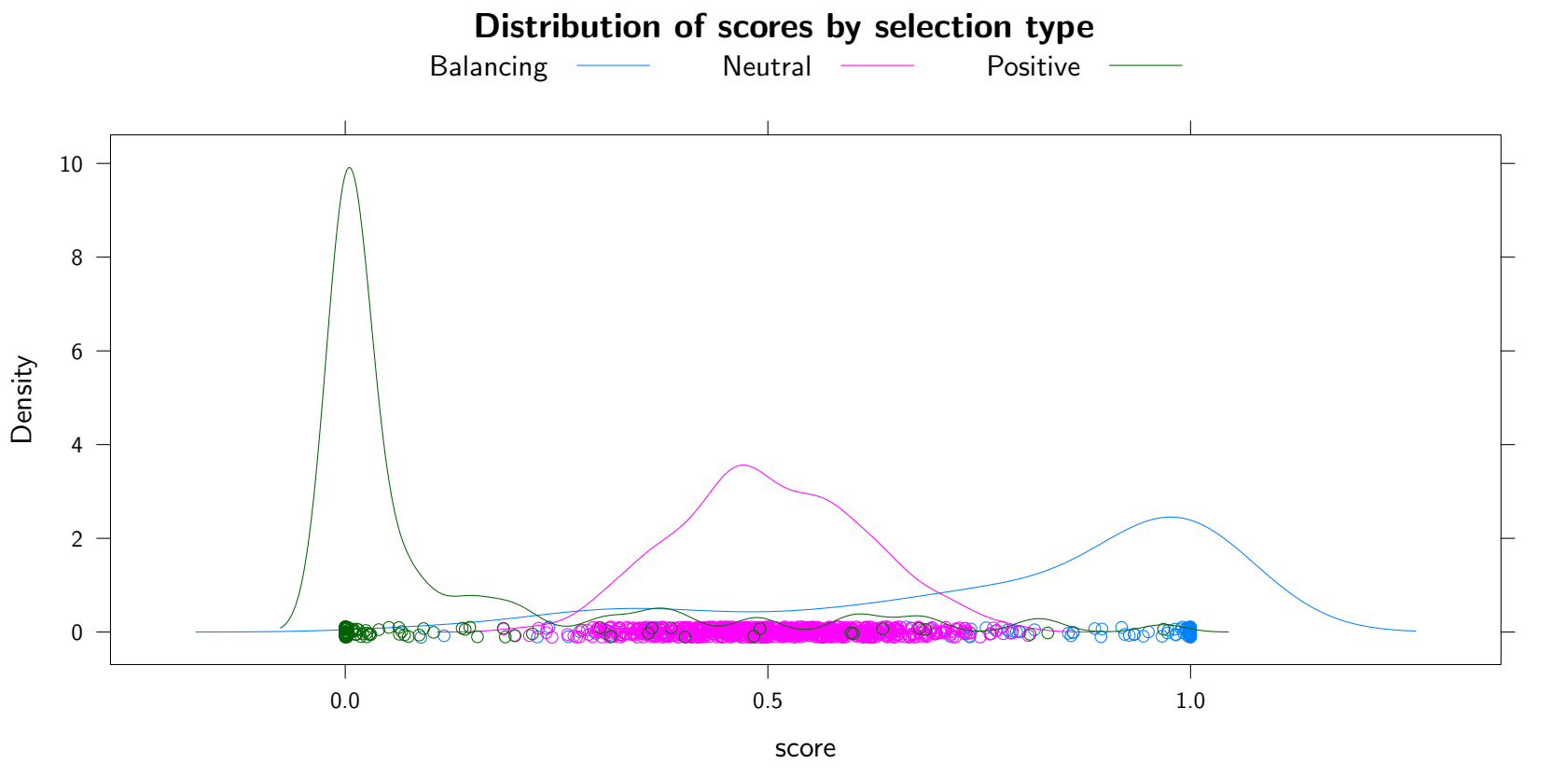
How to add direct labels to some common plots

Recent developments in direct labeling

Conclusions

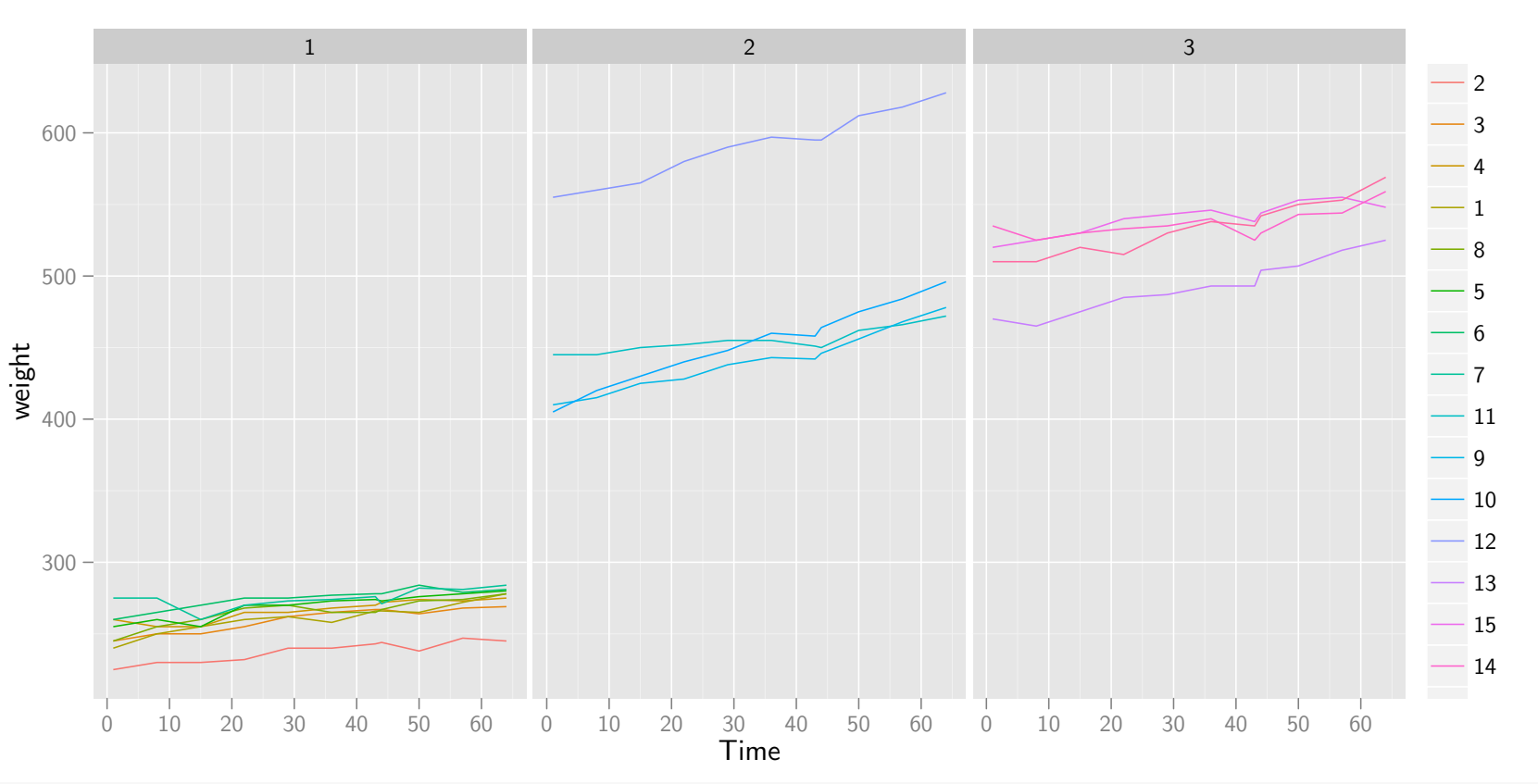
Problem 1: legend inconsistent with data

```
library(lattice)
dens <- densityplot(~score,loci,groups=type,
  auto.key=list(space="top",columns=3),n=500,
  main="Distribution of scores by selection type")
print(dens)
```



Problem 2: too many classes render legend unreadable

```
data(BodyWeight,package="nlme")
library(ggplot2)
ratplot <- ggplot(BodyWeight,aes(Time,weight,colour=Rat))+
  facet_grid(.~Diet)+
  geom_line()
print(ratplot)
```



Motivation: confusing legends

How to add direct labels to some common plots

Recent developments in direct labeling

Conclusions

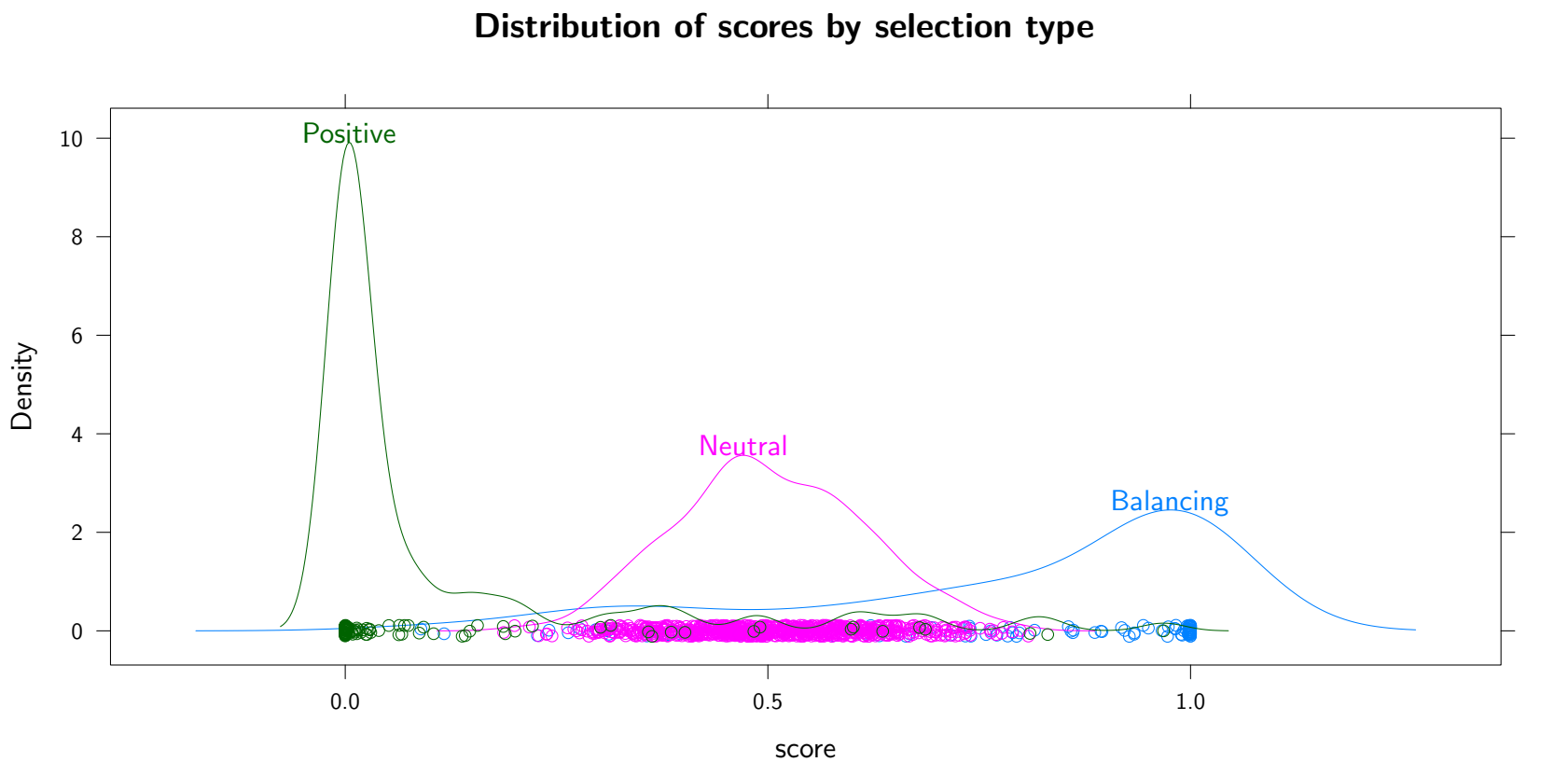
The protocol I use for everyday plots in practice:

Do as many steps as needed until the plot is readable:

1. Make a lattice or ggplot2 plot `p` using colors and default legends.
2. Try the default direct labels: `direct.label(p)`.
3. Check to see if another Positioning Method exists on `http://directlabels.r-forge.r-project.org/docs/index.html` then use `direct.label(p,"method")`.
4. If no Positioning Methods exist you can always write your own.

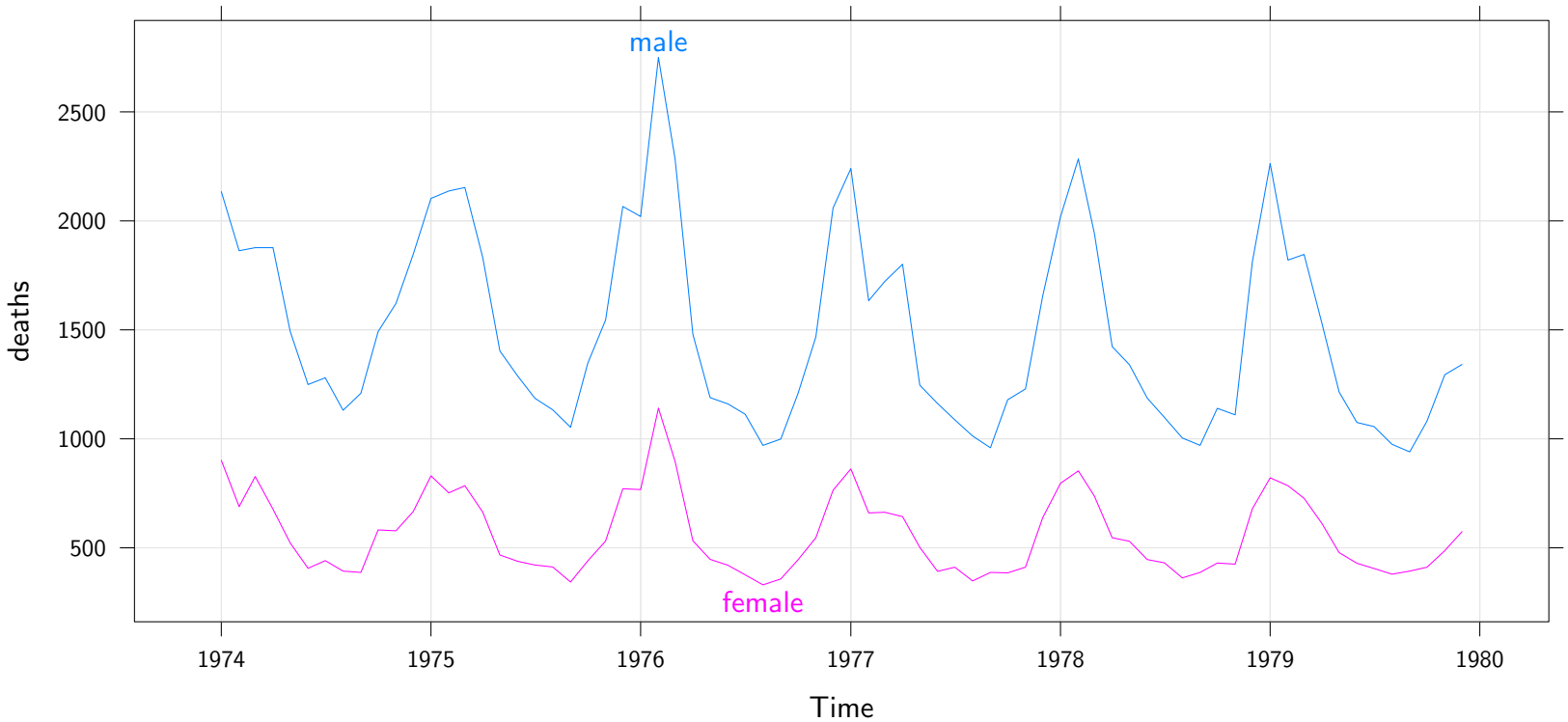
Add default direct labels at the mode of each density

```
library(directlabels)
direct.label(dens)
```



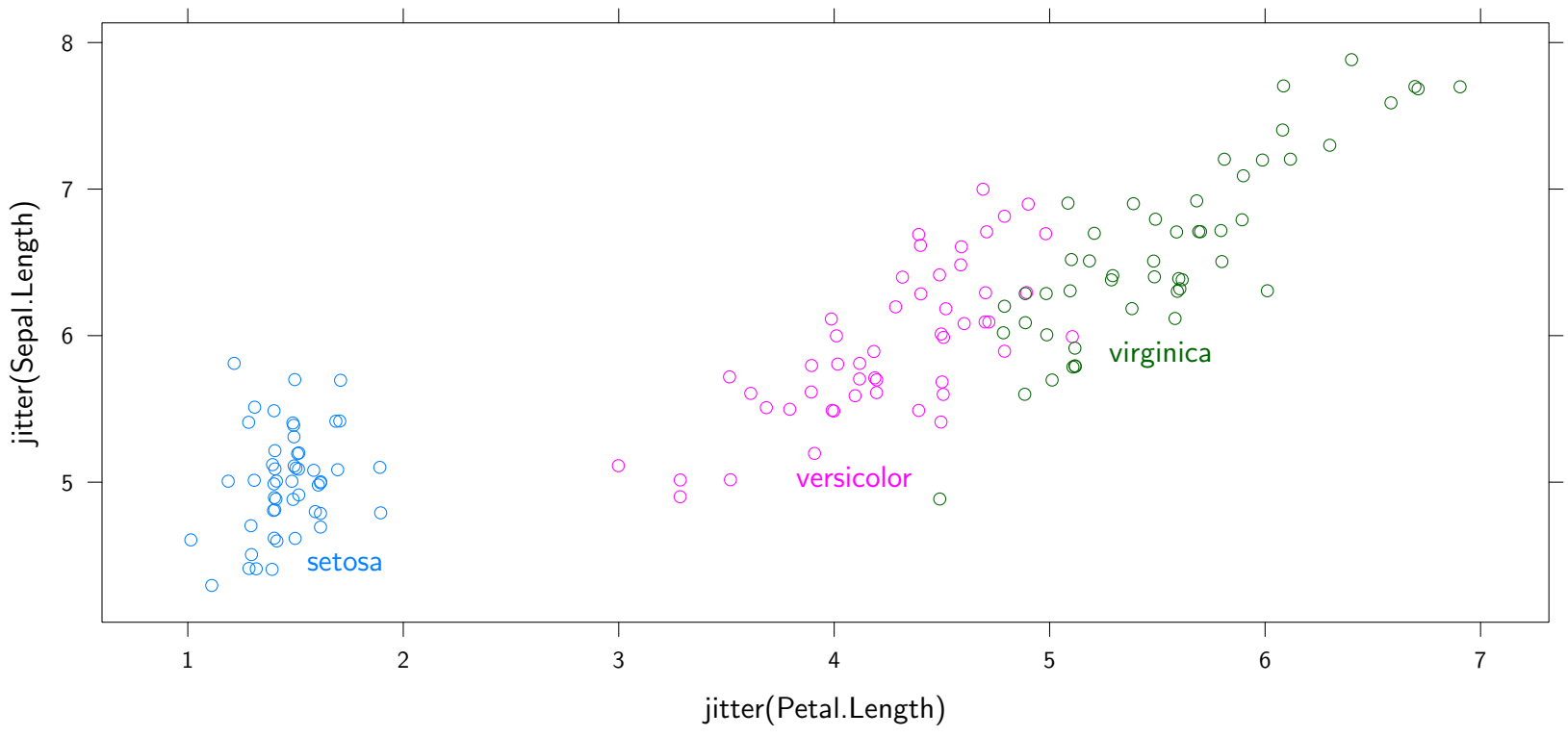
With 2 groups, we label the min and max points

```
p <- xyplot(deaths~Time,uk.lung,
            groups=sex,type=c("l","g"))
direct.label(p)
```



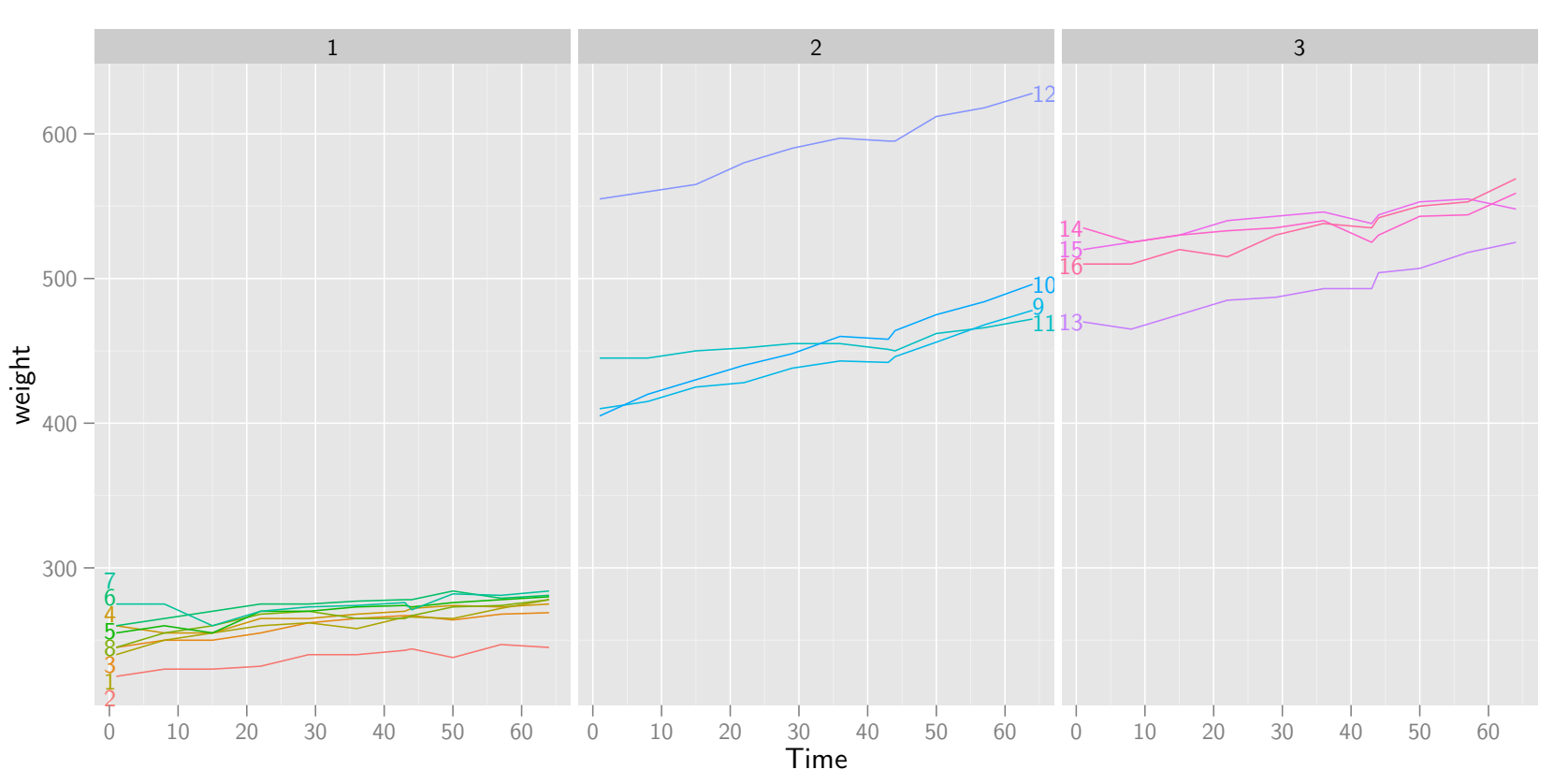
Label a scatterplot of the iris data by species

```
set.seed(1)
irisp <- xyplot(jitter(Sepal.Length)~jitter(Petal.Length),
                iris,groups=Species)
direct.label(irisp)
```



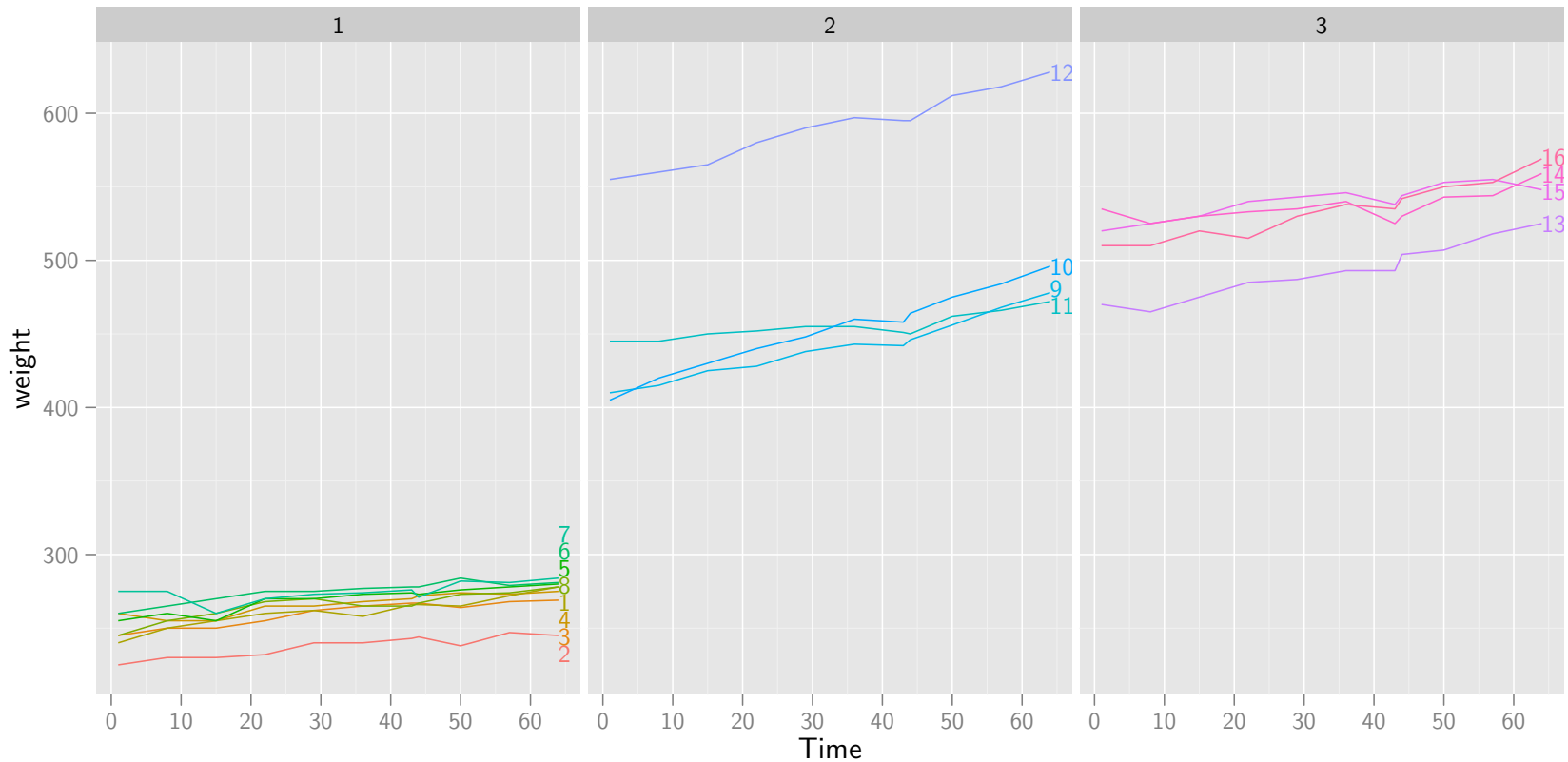
Default direct labels for lineplots

```
direct.label(ratplot)
```



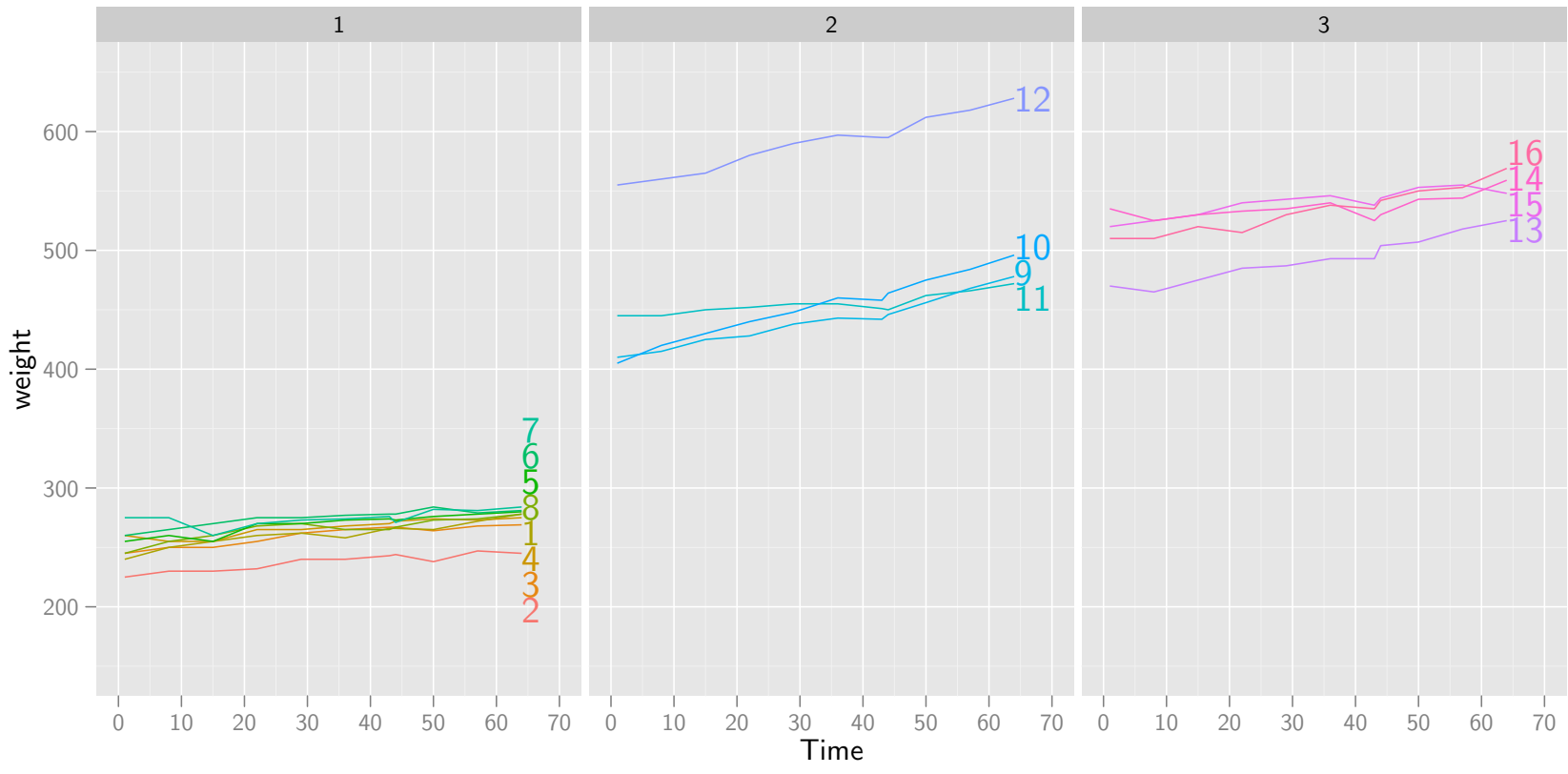
Look up the Positioning Method on the directlabels website

```
direct.label(ratplot,"last.qp")
```



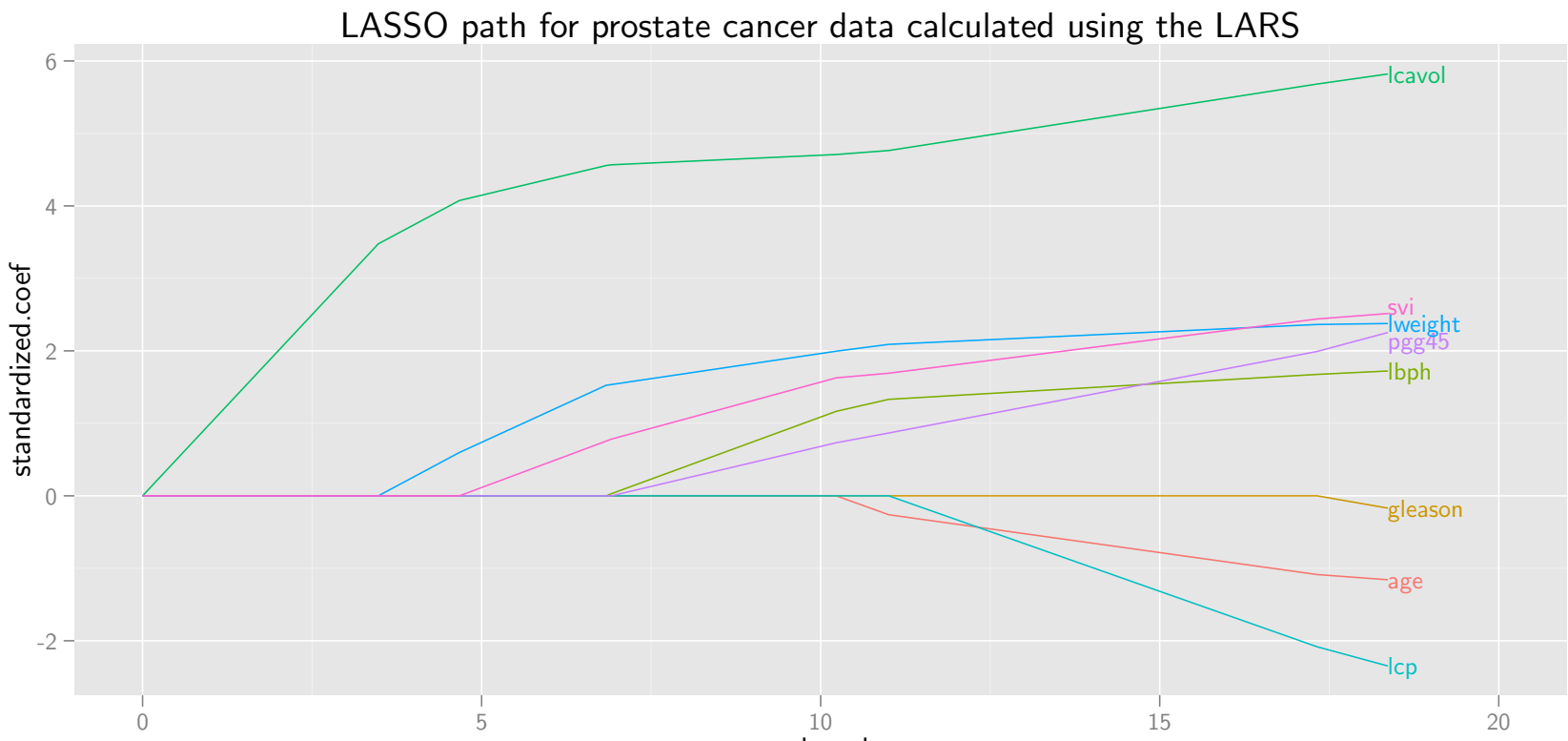
Construct your own custom Positioning Method

```
rp2 <- ratplot+
  xlim(0,70)+ylim(150,650)
big.last <- list(cex=1.5,"last.qp")
direct.label(rp2,"big.last")
```



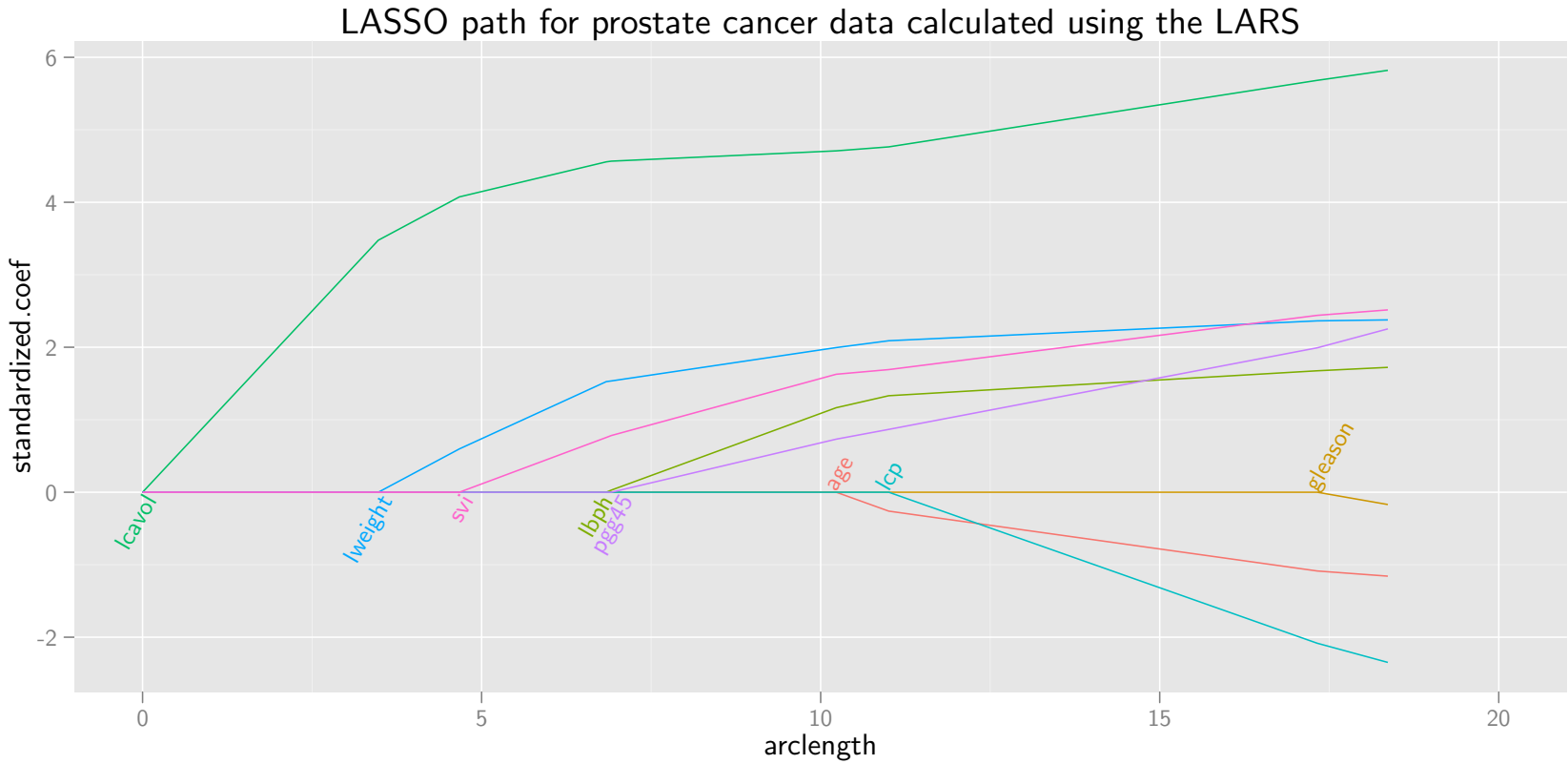
Direct label the LASSO path to visualize variables

```
lasso.plot <-
ggplot(path,aes(arclength,standardized.coef,colour=variable))+
  geom_line(aes(group=variable))+
  opts(title="LASSO path for prostate cancer data
calculated using the LARS")+
  xlim(0,20)
direct.label(lasso.plot)
```



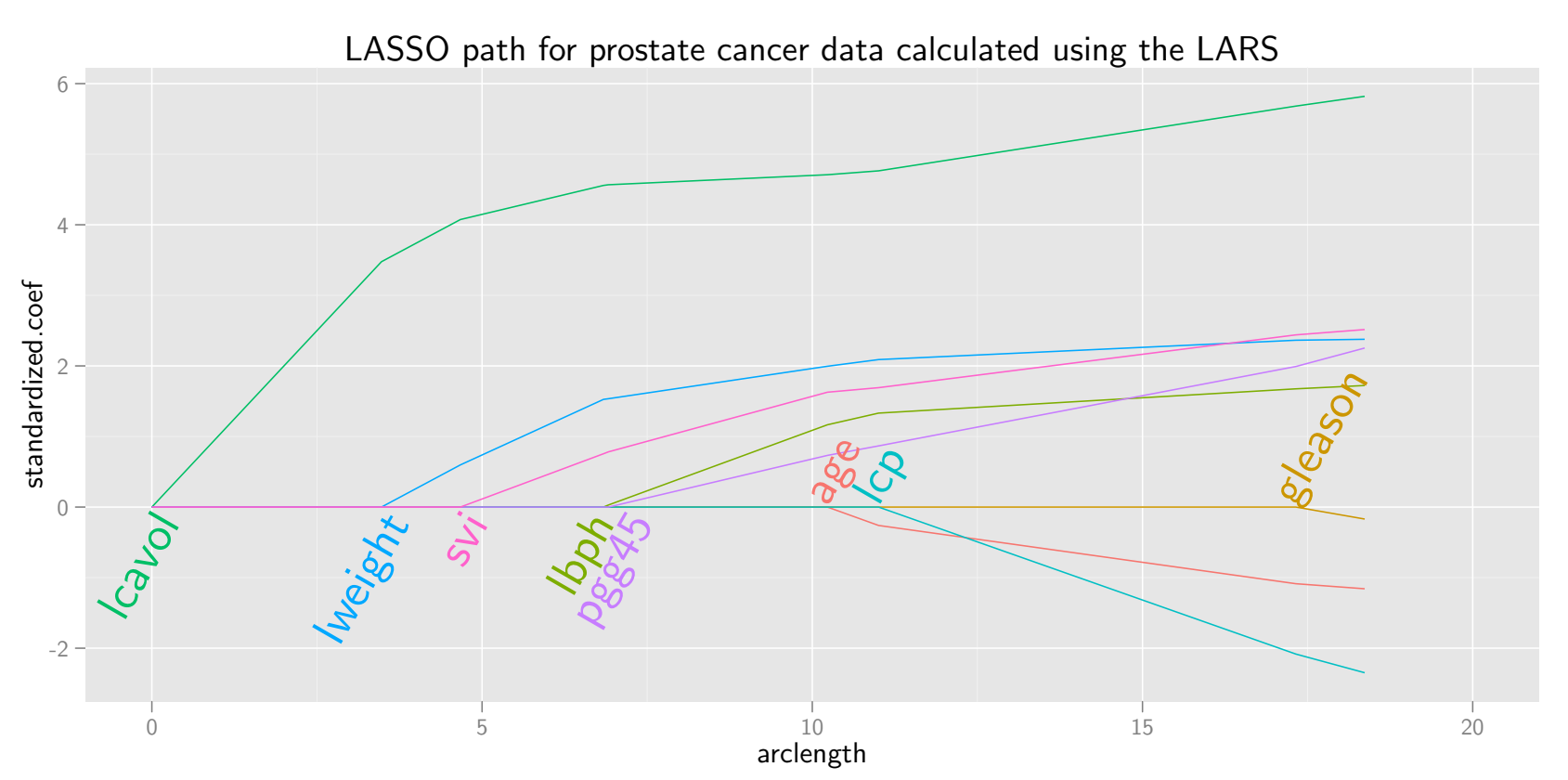
Label the zero point to emphasize variable importance

```
direct.label(lasso.plot,"lasso.labels")
```



Increase text size to make reading easier

```
direct.label(lasso.plot,list(cex=2,"lasso.labels"))
```



Motivation: confusing legends

How to add direct labels to some common plots

Recent developments in direct labeling

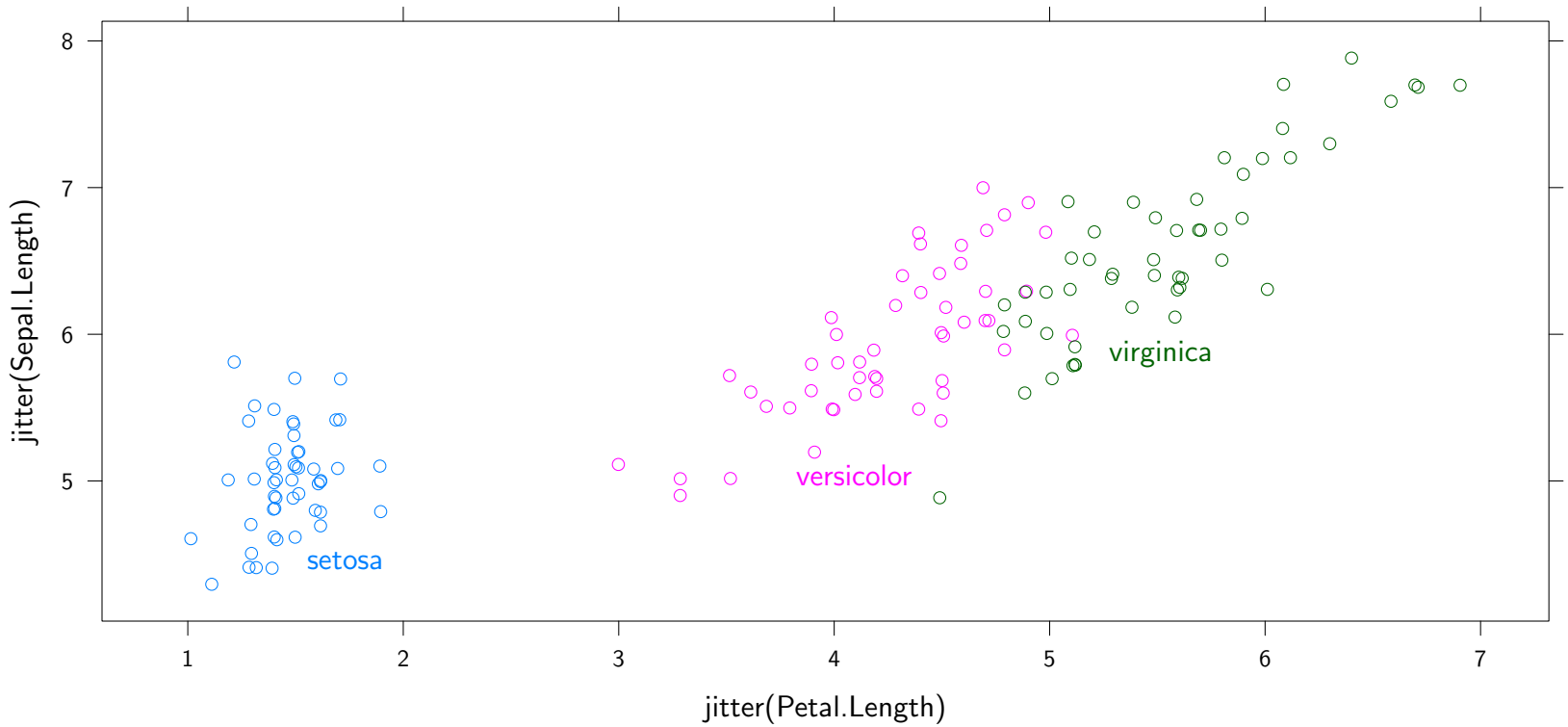
Conclusions

Changes in recent versions of directlabels

directlabels version plotting package	< 2.0 lattice	< 2.0 ggplot2	≥ 2.0 lattice	≥ 2.0 ggplot2
basic Positioning Methods	✓	✓	✓	✓
smart Positioning Methods that avoid label collisions	✓		✓	✓
redraw labels after window resize			✓	✓
`fontface` and `fontfamily` text parameters	✓		✓	✓
label black and white plots	✓		✓	✓
label aesthetics other than color				✓

Label a scatterplot of the iris data by species

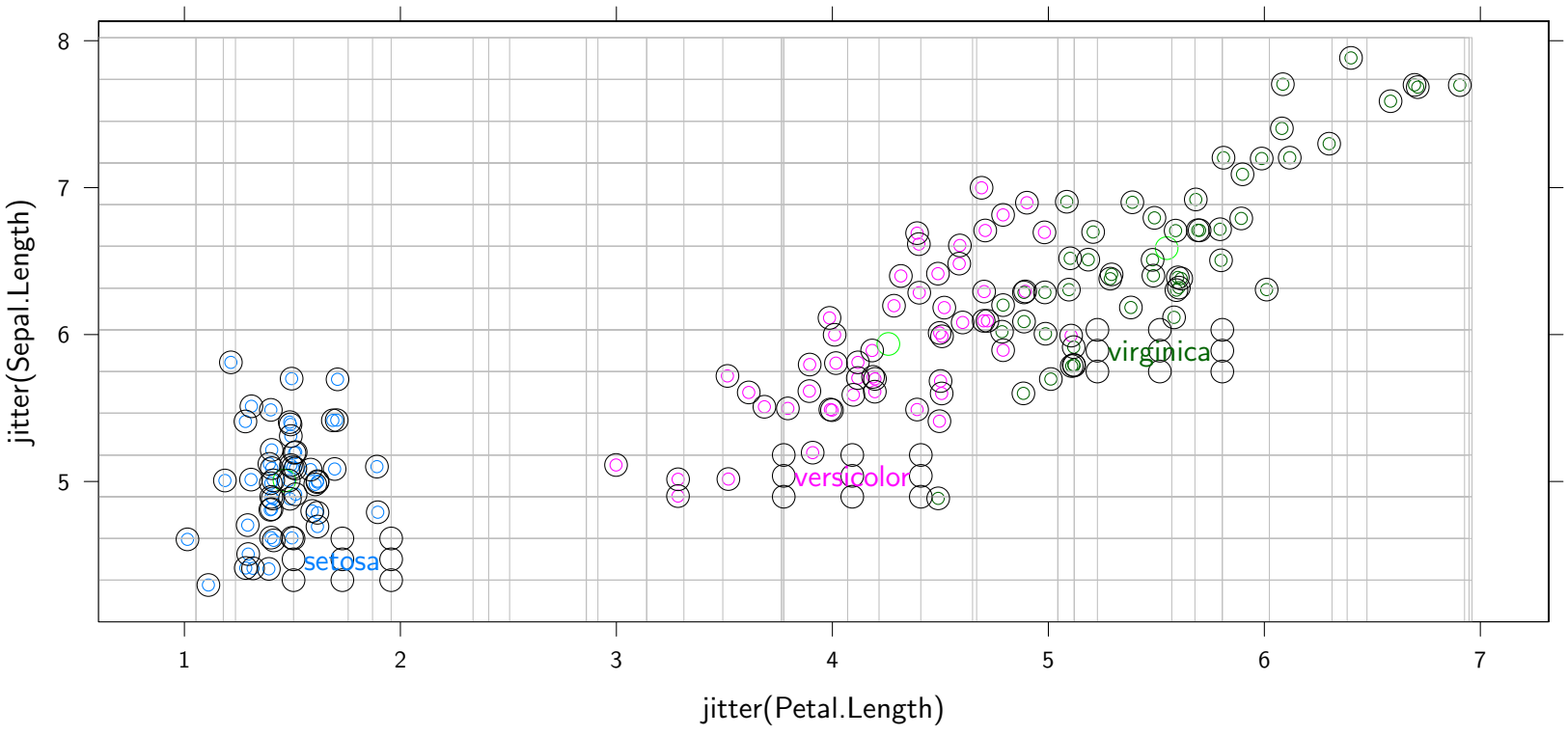
```
direct.label(irisp)
```



Show the grid for the search

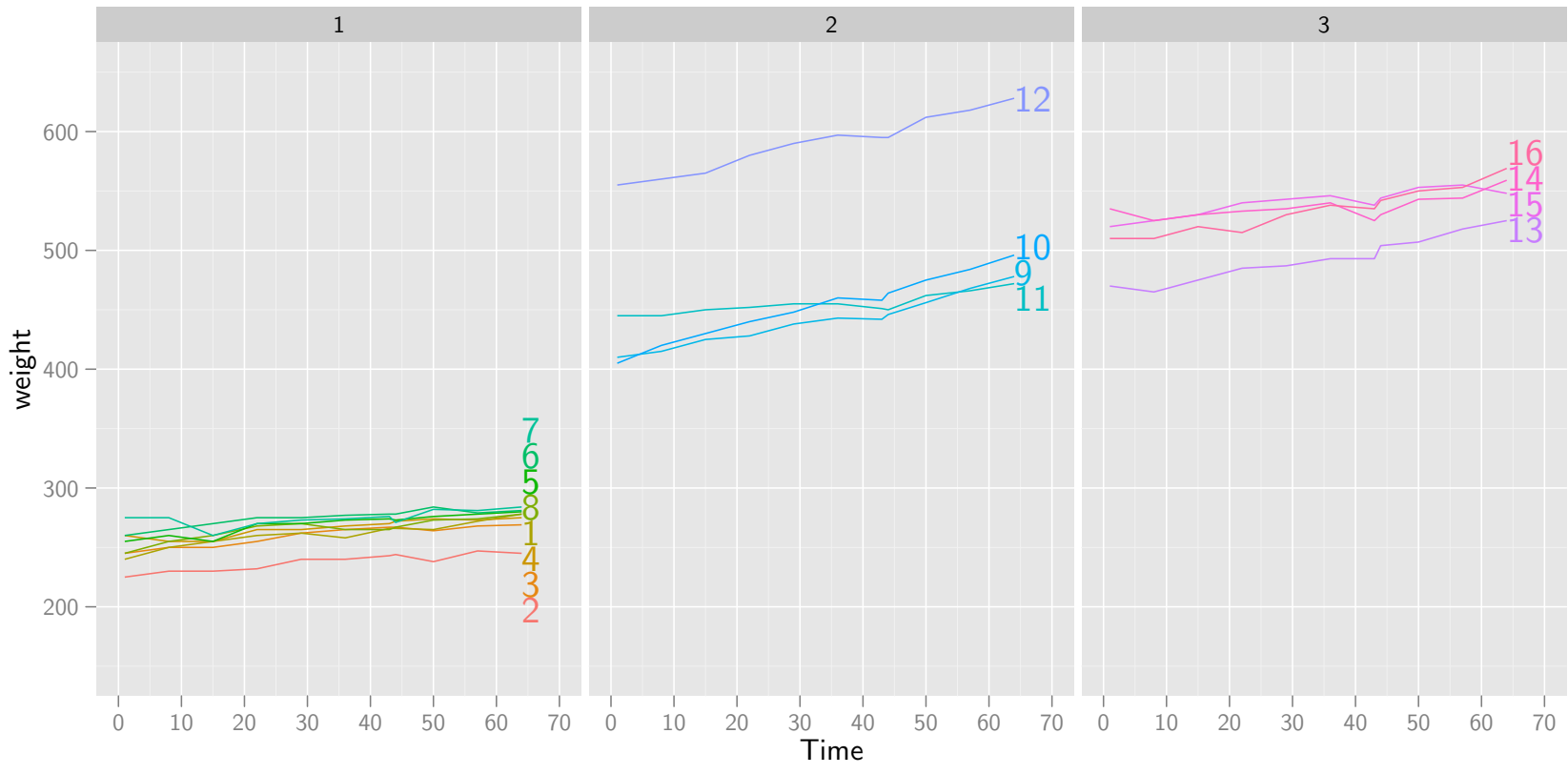
Find a label position on a grid (grey rectangles)
that is near the center of each point cloud (green dots),
but does not overlap any points or other labels (black dots).

```
direct.label(irisp,debug=TRUE)
```



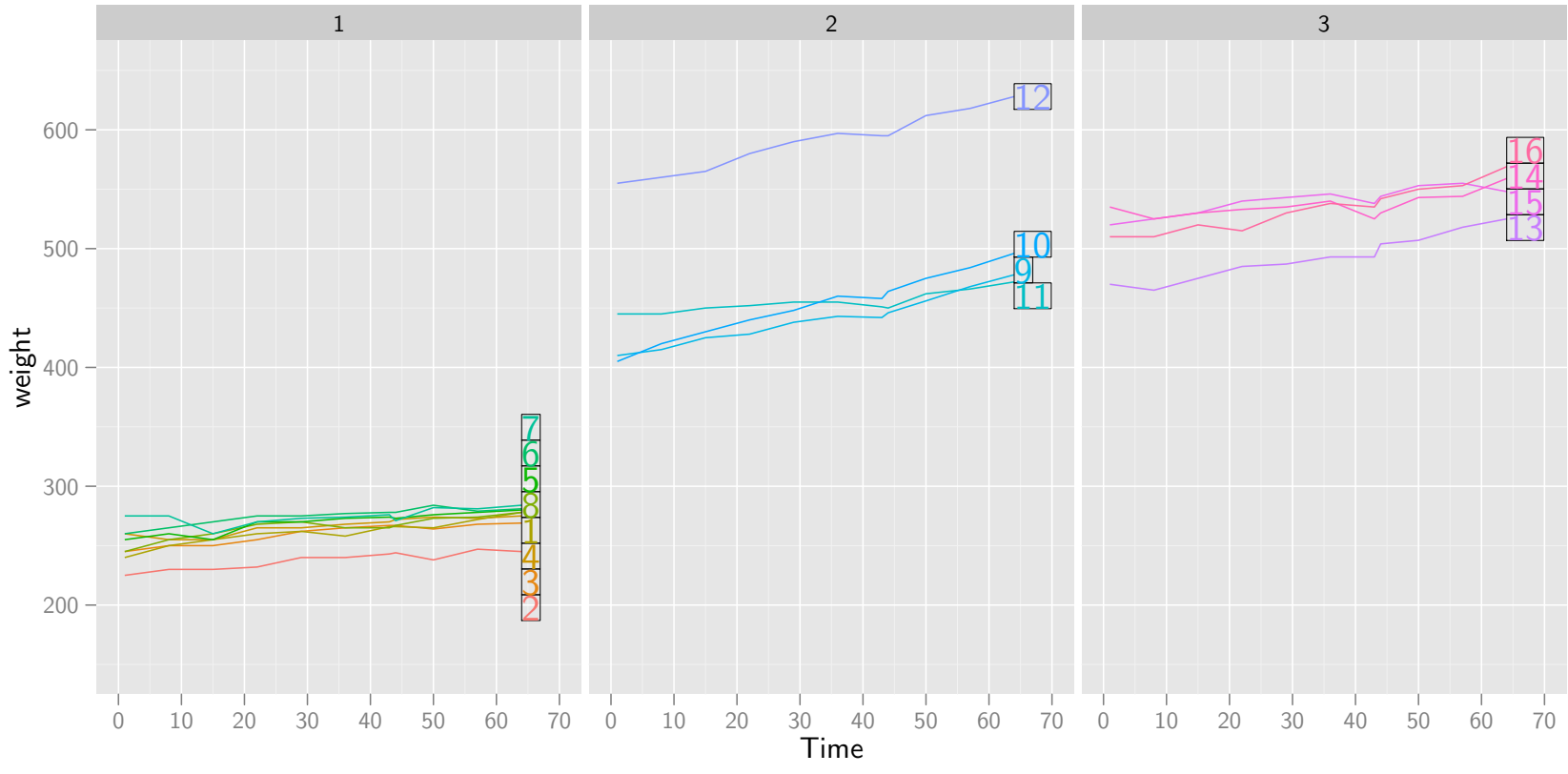
Increase text size before calculating label positions

```
rp2 <- ratplot+
  xlim(0,70)+ylim(150,650)
big.last <- list(cex=1.5,"last.qp")
direct.label(rp2,"big.last")
```

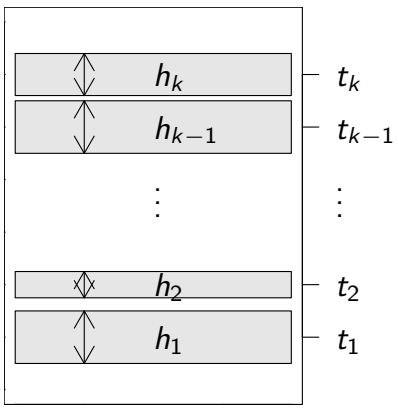


Show the label borders used in the position calculation

```
direct.label(rp2,
  list("big.last",
       "calc.boxes",
       "draw.rects"))
```



Label positions for lineplots are the solutions of a QP



Assume that for each text label $i = 1, \ldots, k$ we have its position t_i and height h_i.
Then optimal direct labels do not overlap, and are as close as possible to the target locations:

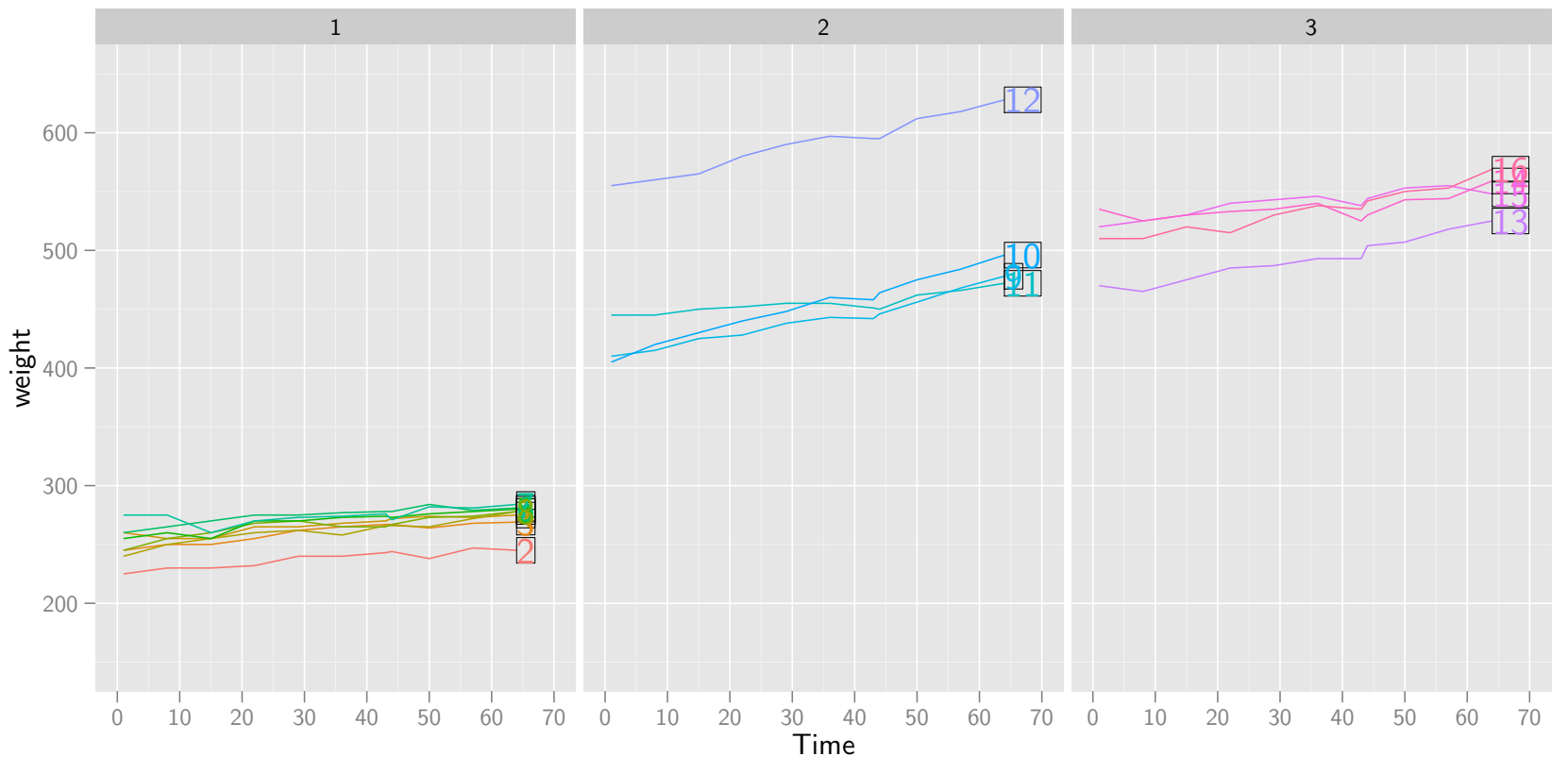
$$\min_{b \in \mathbb{R}^k} \quad \sum_{i=1}^{k} (b_i - t_i)^2 = ||b - t||^2$$

$$\text{subject to} \quad b_{i+1} \geq b_i + h_{i+1}/2 + h_i/2, \ \forall \ i = 1, \ldots, k-1$$

This is a quadratic program (QP). QPs are convex so there is a unique global minimum which corresponds to the best labels.
We can solve this using `quadprog::solve.QP()` and use the optimal b for the direct label positions.

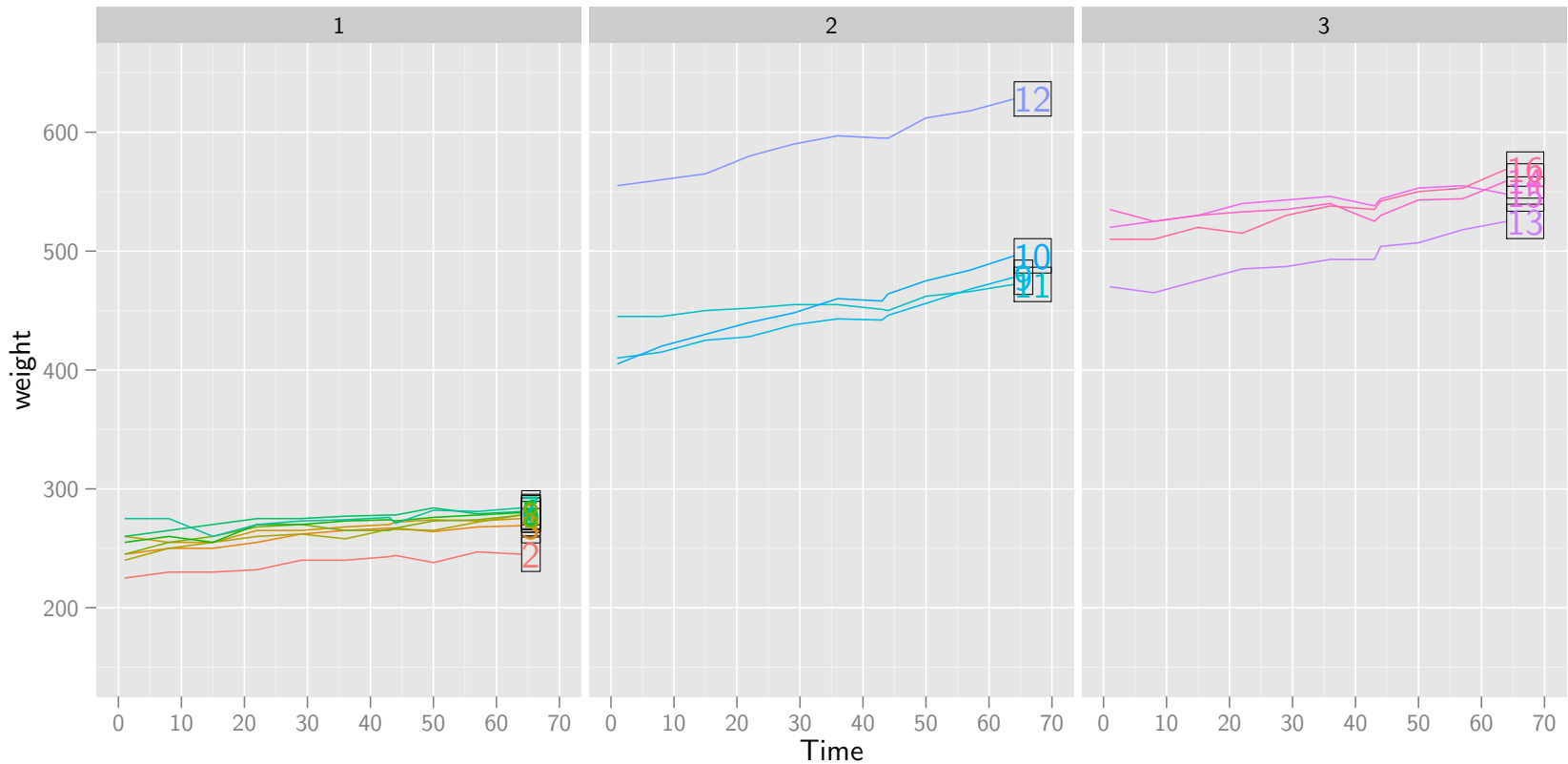
Start with boxes at the end of each line

```
direct.label(rp2,list("last.points",cex=1.5,
                      "calc.boxes",
                      "draw.rects"))
```



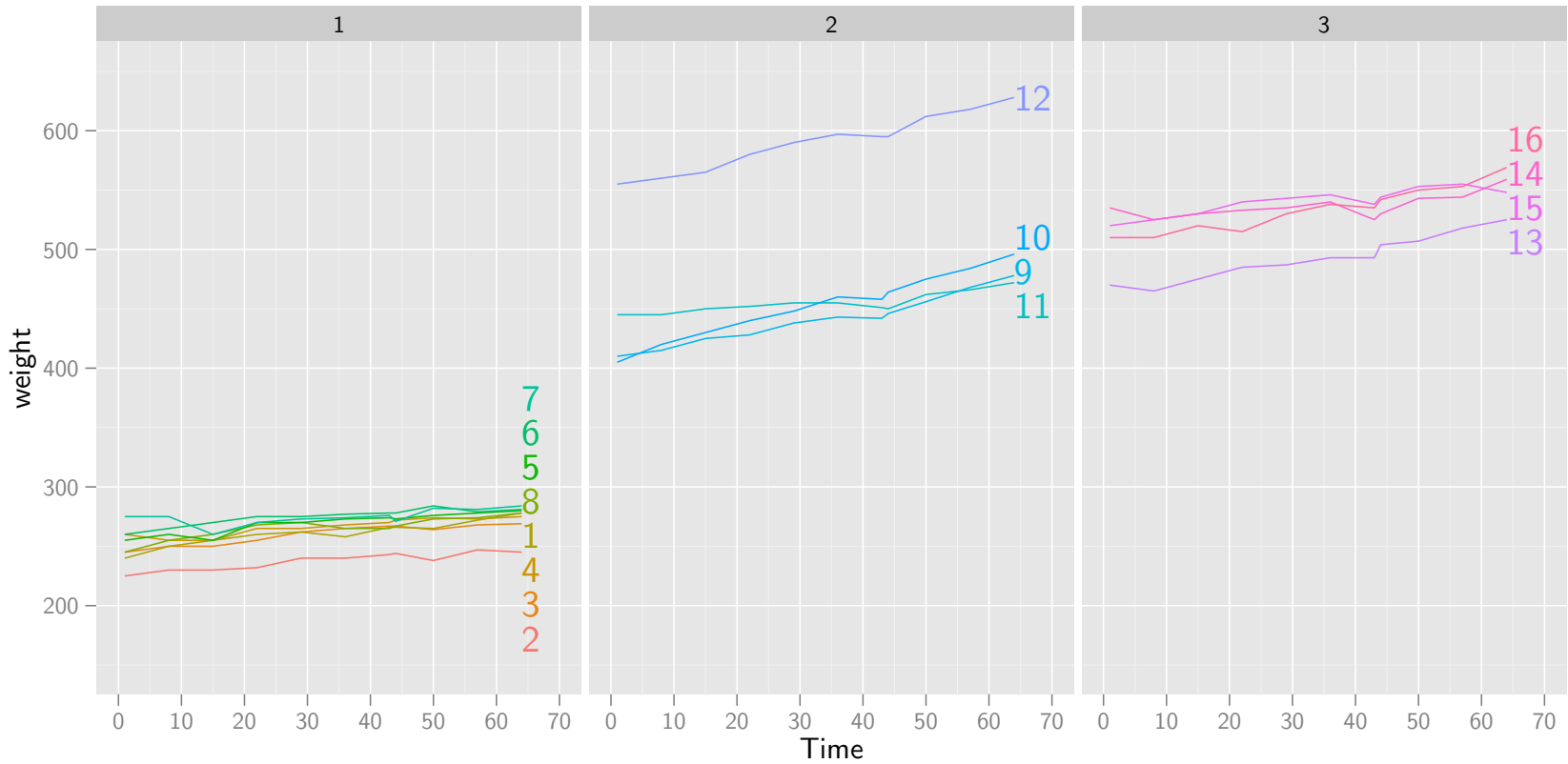
Adjust box height if desired

```
direct.label(rp2,list("last.points",cex=1.5,"calc.boxes",
                      dl.trans(h=h+h/3),"calc.borders",
                      "draw.rects"))
```



Apply QP solver to get optimal labels

```
direct.label(rp2,list("last.points",cex=1.5,"calc.boxes",
                      dl.trans(h=h+h/3),"calc.borders",
                      qp.labels("y","h")))
```



Motivation: confusing legends

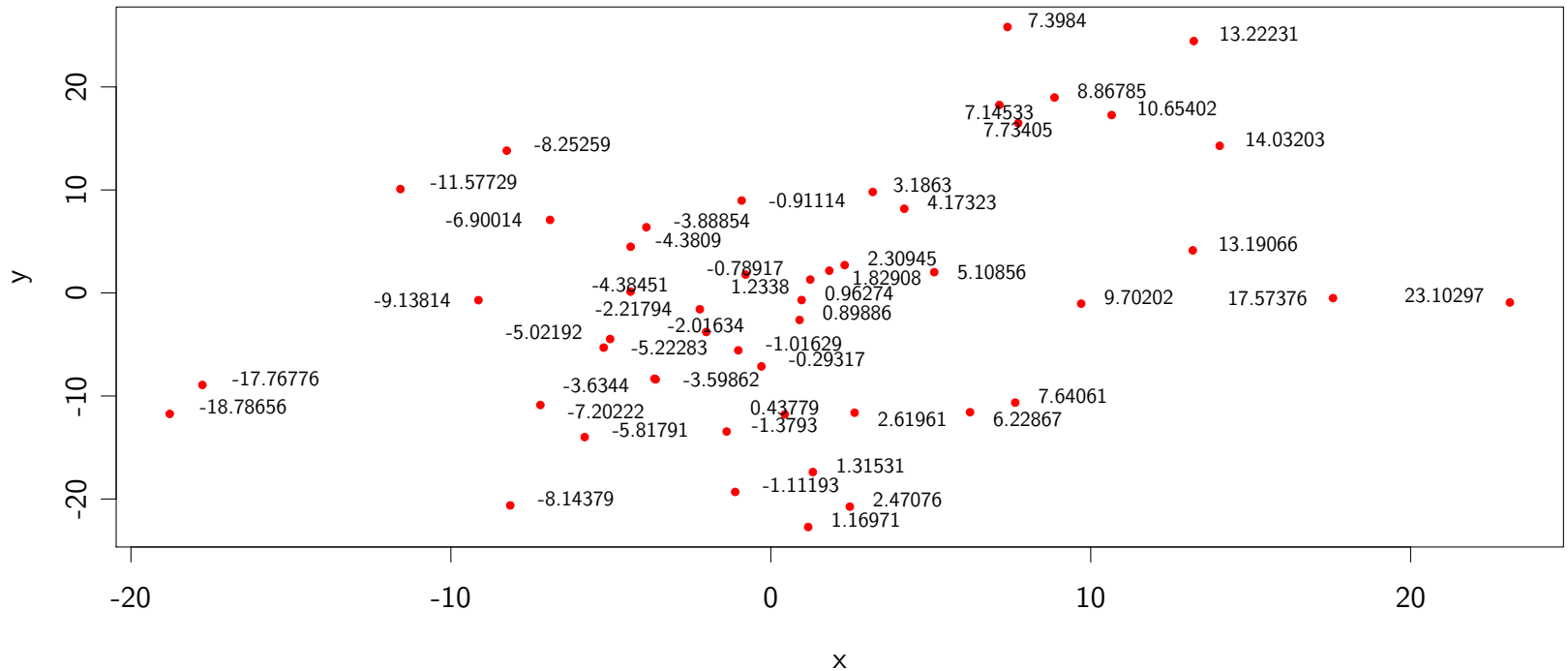
How to add direct labels to some common plots

Recent developments in direct labeling

Conclusions

What directlabels is NOT

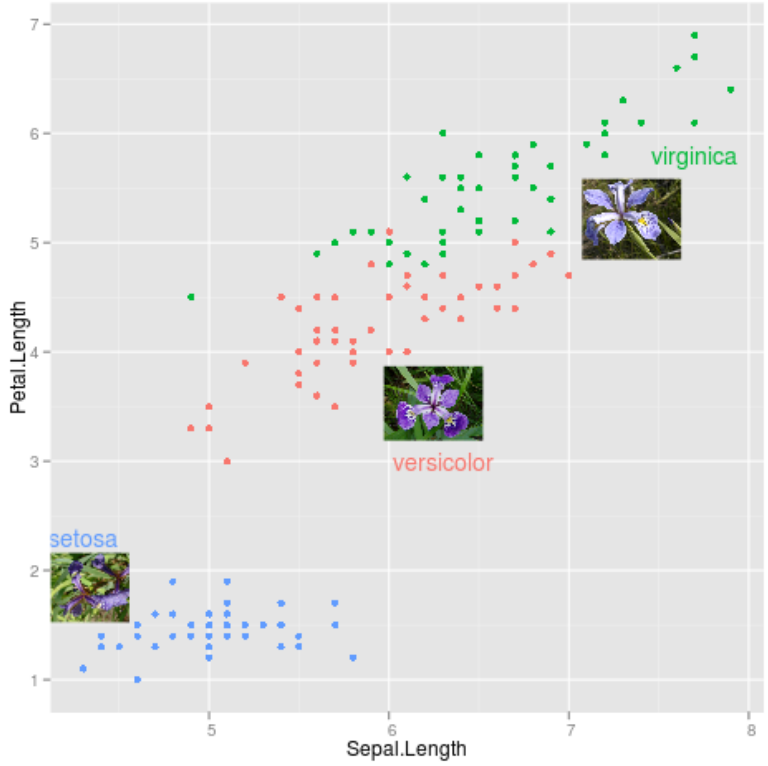
- Automatic publication-quality direct labels.
  (some manual tweaking will always be necessary)
- Optimal labels for individual points on scatterplots.
  (it is a bit more complicated)



Use directlabels instead of confusing legends!

- Works with lattice and ggplot2.
- Sensible defaults.
- Useful in everyday plots in practice.
- Smart Positioning Methods that avoid label collisions.
- Customizable: you can write your own Positioning Methods.

Future work

- Automatically adjust xlim/ylim so labels stay on plot region?
- Contourplot labels as in `contour()`?
- Label using images instead of textual factor names?
  Possible Google Summer of Code 2012 project:

  `http://rwiki.sciviews.org/doku.php?id=developers:projects:gsoc2012`