Various ways to compare histograms

Ben Bolker

March 19, 2012

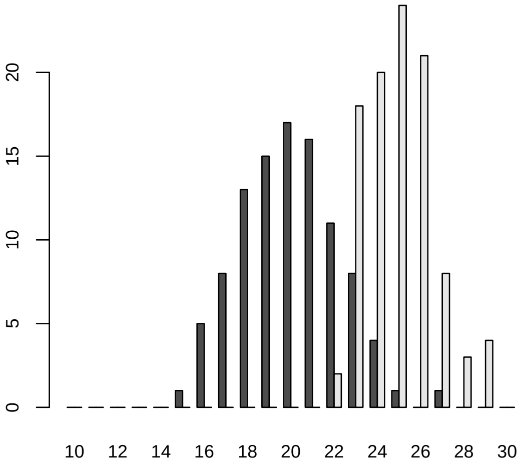
Graphically comparing distributions, especially with small samples, is a challenge. Here are some approaches.

Example data:

```
set.seed(1001)
z1 <- rnorm(100, mean = 20, sd = 2)
z2 <- rnorm(100, mean = 25, sd = 2)
```

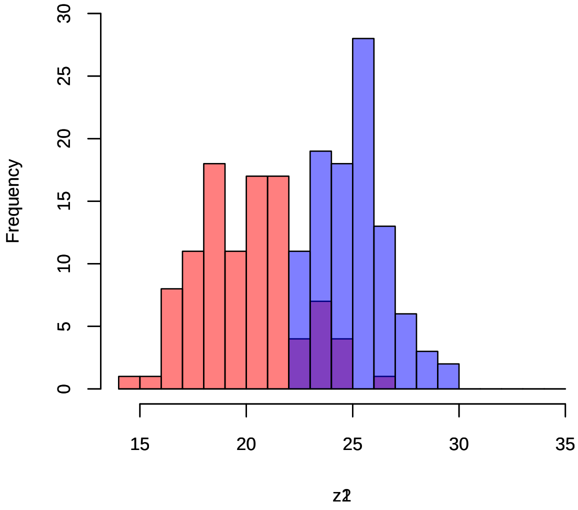
The `multhist` function from the `plotrix` package:

```
library(plotrix)
m <- multhist(list(z1, z2), breaks = seq(9.5,
    30.5), names.arg = 10:30)
```



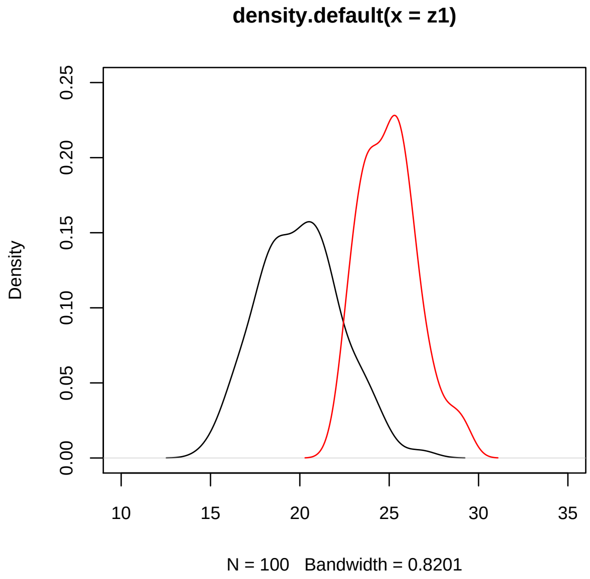
Or overlay histograms with transparent colors:

```
trred <- rgb(1, 0, 0, 0.5)  ## transparent red
trblue <- rgb(0, 0, 1, 0.5)  ## transparent blue
hist(z1, col = trred, breaks = 14:35, main = "",
    ylim = c(0, 30))
par(new = TRUE)
hist(z2, col = trblue, breaks = 14:35, main = "",
    ylim = c(0, 30))
```



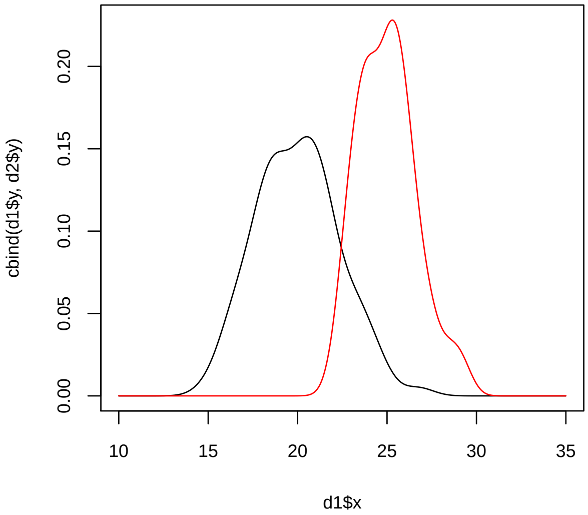
Comparing density estimates instead:

```
plot(density(z1), ylim = c(0, 0.25), xlim = c(10,
    35))
lines(density(z2), col = 2)
```



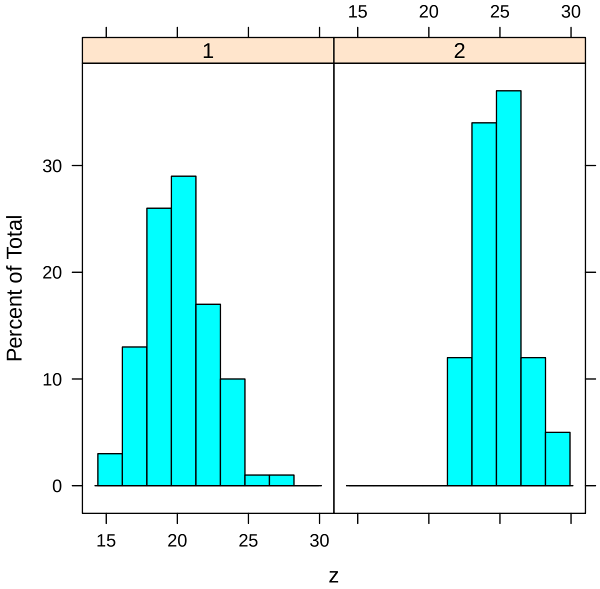
Alternative code:

```
d1 = density(z1, from = 10, to = 35)
d2 = density(z2, from = 10, to = 35)
matplot(d1$x, cbind(d1$y, d2$y), lty = 1, col = 1:2,
    type = "l")
```



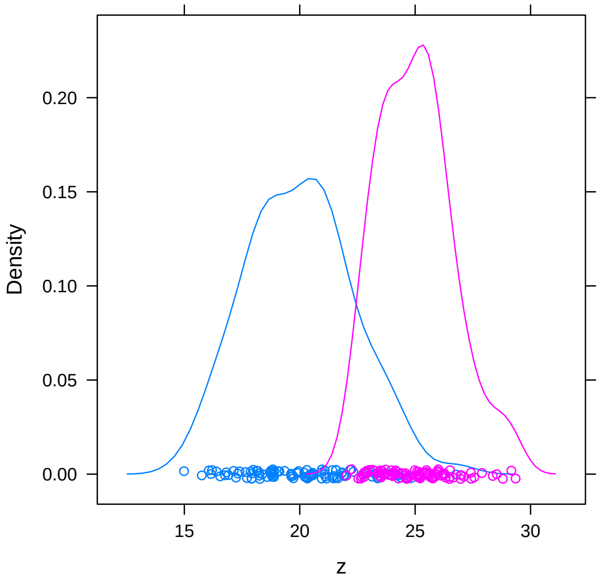
Using lattice or ggplot2 packages requires that we combine the data into a single data frame:

```
dat <- data.frame(z = c(z1, z2), grp = factor(rep(1:2,
    each = 100)))
library(lattice)
print(histogram(~z | grp, data = dat))
```



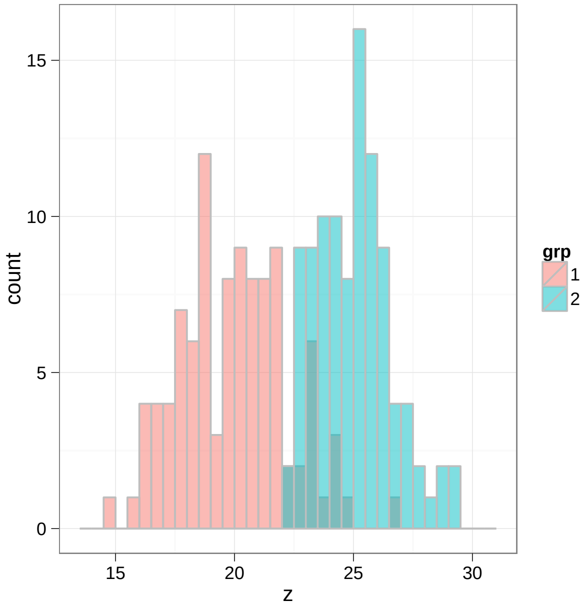
The `groups` argument works for density plots, not histograms (from `?histogram`: "Note that the default panel function for 'histogram' does not support grouped displays, whereas the one for 'densityplot' does.").

```
densityplot(~z, groups = grp, data = dat)
```



ggplot is the new hotness.

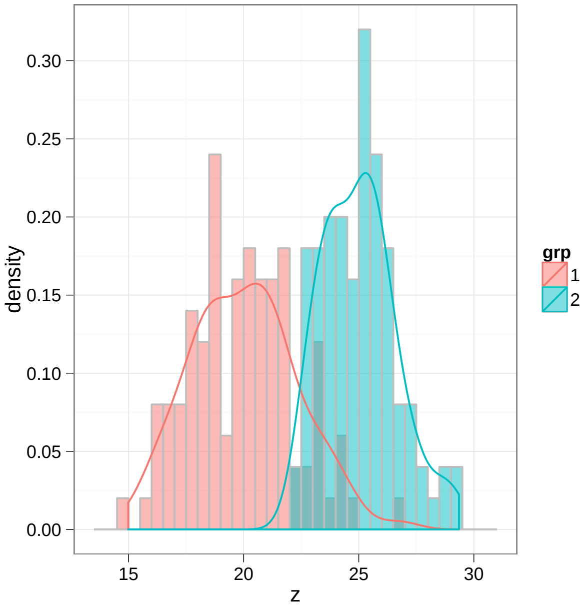
```
library(ggplot2)
## overlapping/transparent
ggplot(dat, aes(x = z, group = grp, fill = grp)) +
    geom_histogram(colour = "gray", binwidth = 0.5, alpha = 0.5,
        position = "identity")
```



(Note that lattice plots have the groups argument, but ggplot uses the group aesthetic. I used theme_set(theme_bw()) to change to my preferred plot style.)

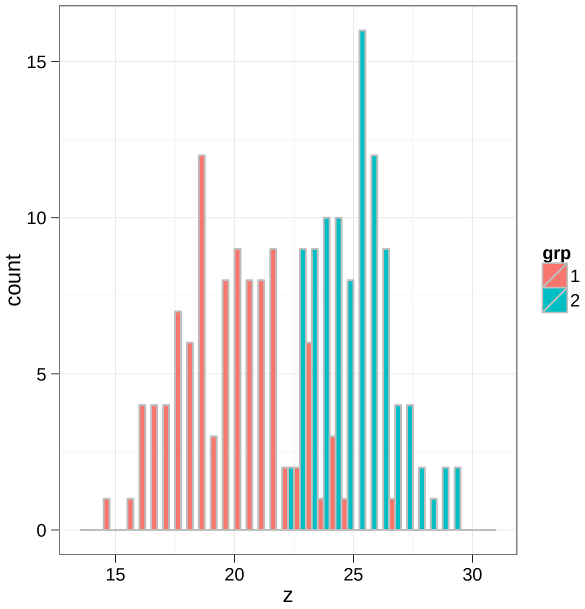
Superimpose histograms (scaled to prob. density) and density lines:

```
ggplot(dat, aes(x = z, group = grp, fill = grp,
    colour = grp)) + stat_bin(colour = "gray", binwidth = 0.5,
    alpha = 0.5, position = "identity", aes(y = ..density..)) +
    geom_density(fill = NA)
```



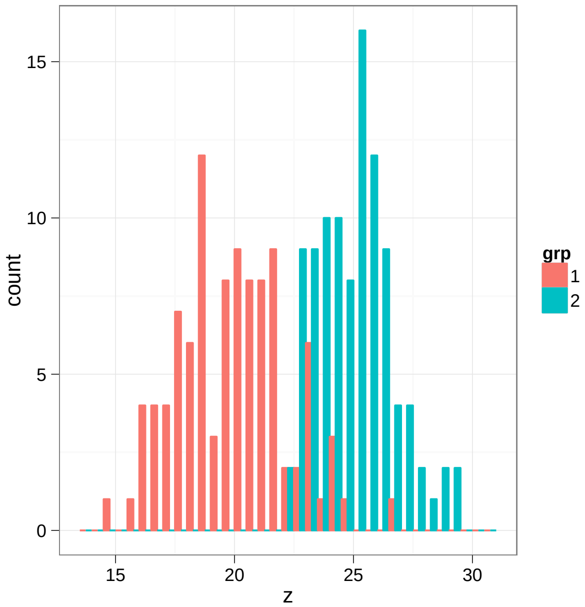
Put the bars next to each other:

```
ggplot(dat, aes(x = z, group = grp, fill = grp)) +
    geom_histogram(colour = "gray", binwidth = 0.5, position =
"dodge") +
    theme_bw()
```



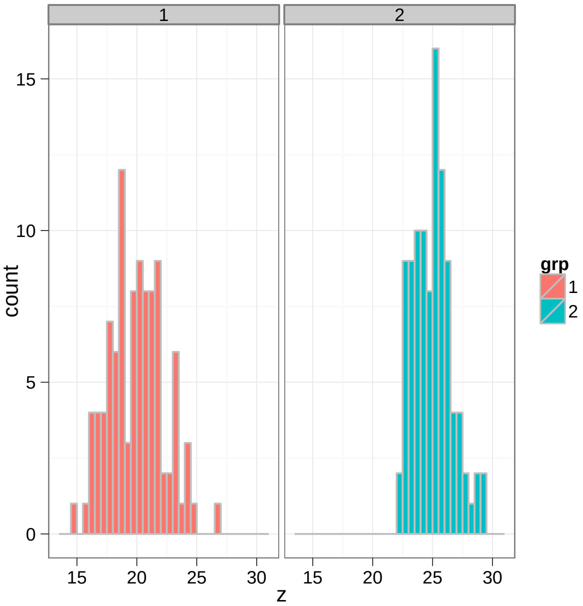
The same thing but without gray borders on the bars (seem to overlap slightly??)

```
ggplot(dat, aes(x = z, group = grp, fill = grp,
    colour = grp)) + geom_histogram(binwidth = 0.5, position =
"dodge") +
    theme_bw()
```



Separate graphs:

```
ggplot(dat, aes(x = z, group = grp, fill = grp)) +
    geom_histogram(colour = "gray", binwidth = 0.5) +
facet_wrap(~grp) +
    theme_bw()
```



For more than two groups one should probably look into vertical presentations of the data, i.e. boxplots/violin plots/bean plots . . .