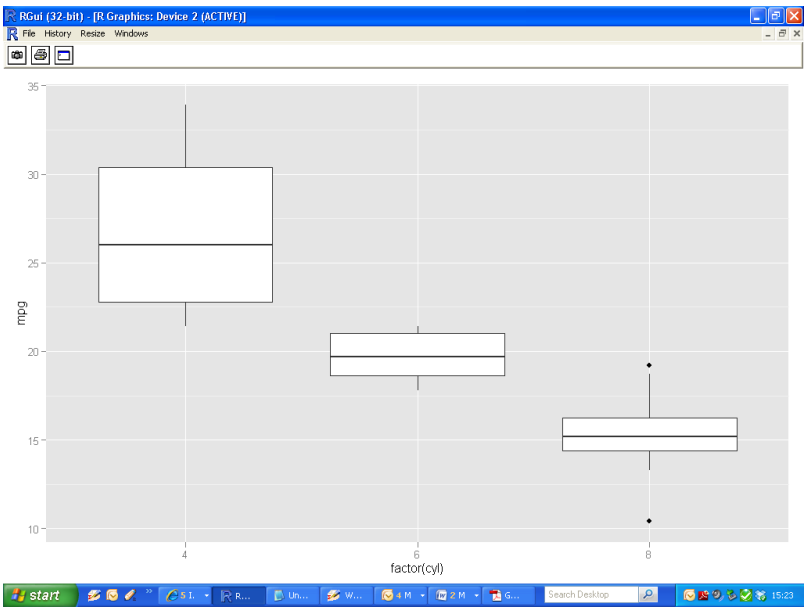
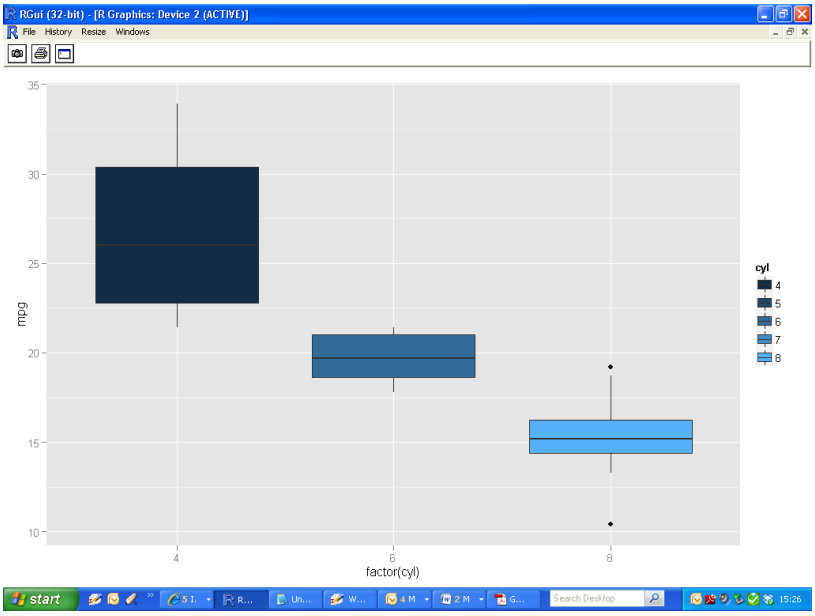
# Contents



# Boxplots

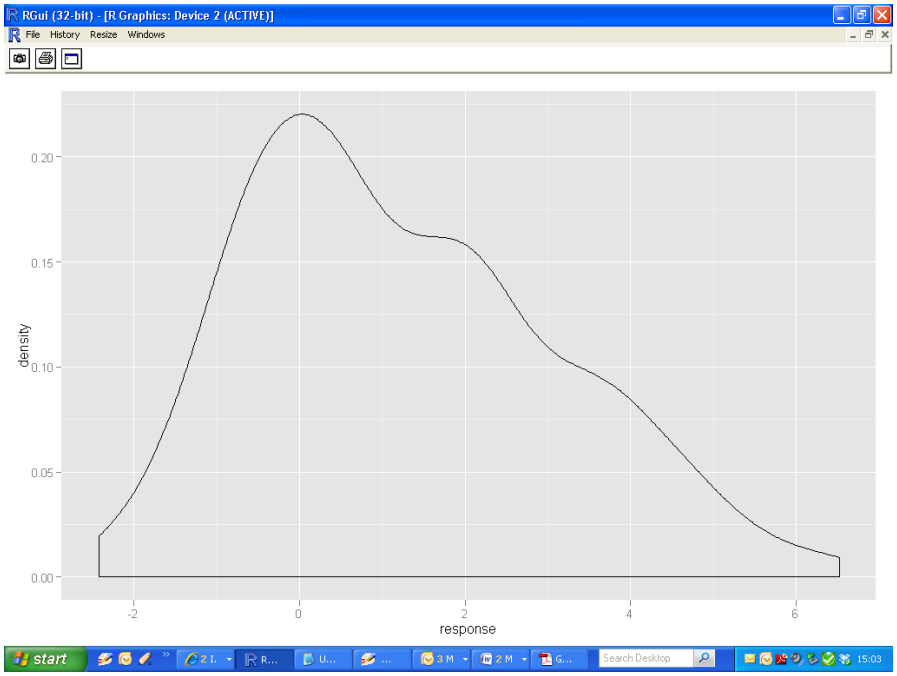
```
p <- ggplot(mtcars, aes(factor(cyl), mpg))

p + geom_boxplot()

qplot(factor(cyl), mpg, data = mtcars, geom = "boxplot")

p + geom_boxplot(aes(fill = cyl))
```

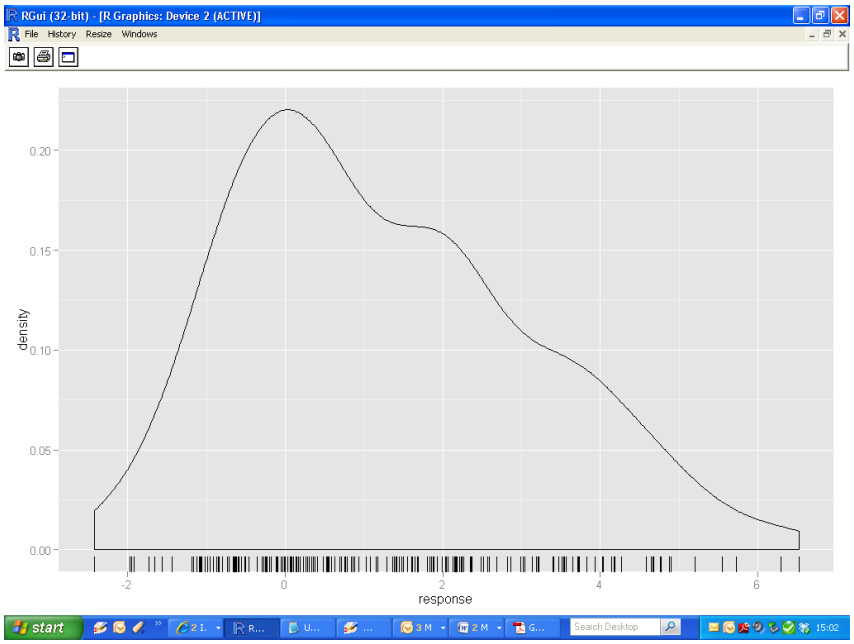



RGui (32-bit) - [R Graphics: Device 2 (ACTIVE)]
File History Resize Windows
35
30
25
20
15
10
mpg
4
6
8
factor(cyl)
cyl
4
5
6
7
8
start
Search Desktop
15:26

## Density Functions

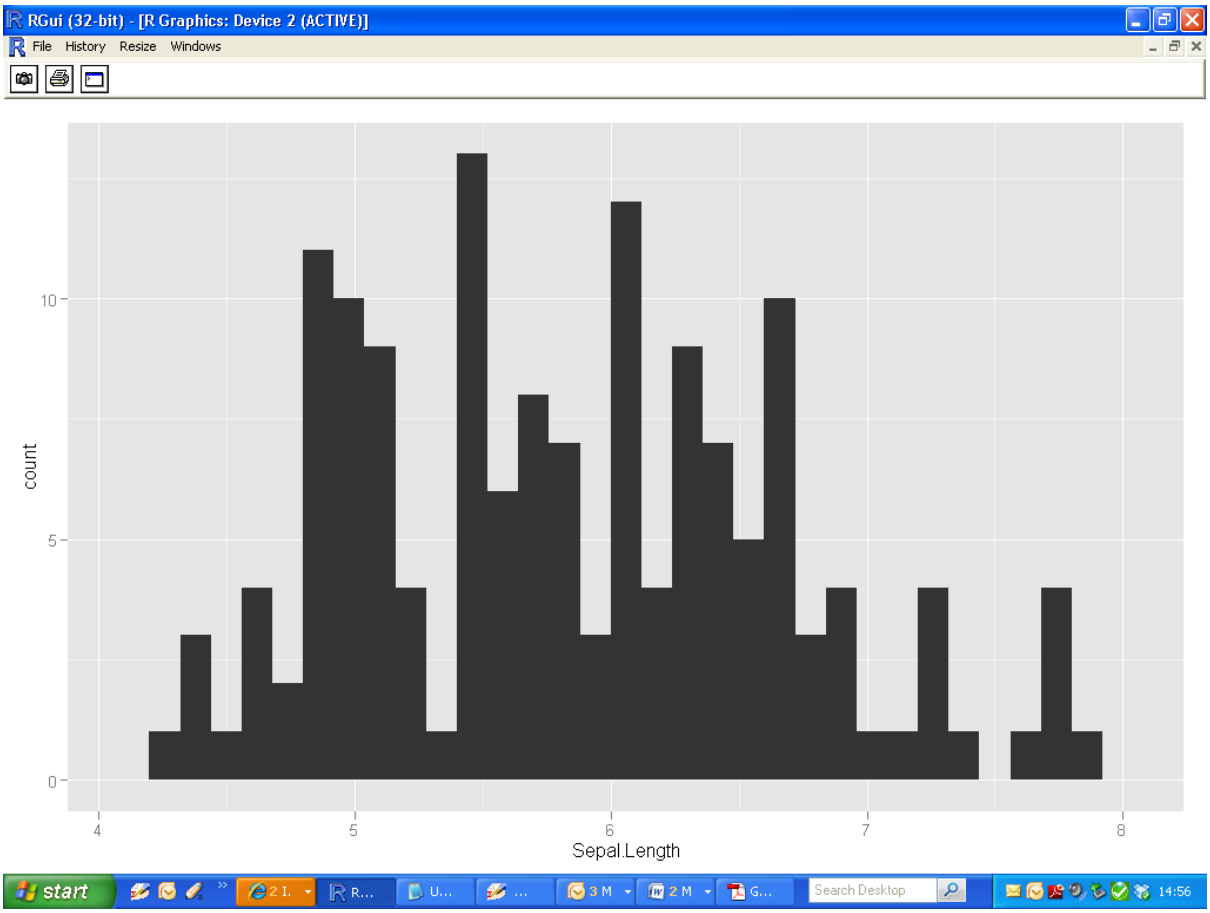
```
response <- c(rnorm(100), rnorm(100, mean=2, sd=2))
mygroup <- factor(c(rep('group a', 100), rep('group b', 100)))
d <- data.frame(response, mygroup)
rm(response, mygroup)

g <- ggplot(d, aes(x=response)) + geom_density()
g
g <- ggplot(d, aes(x=response)) + geom_density() + geom_rug()
g
```



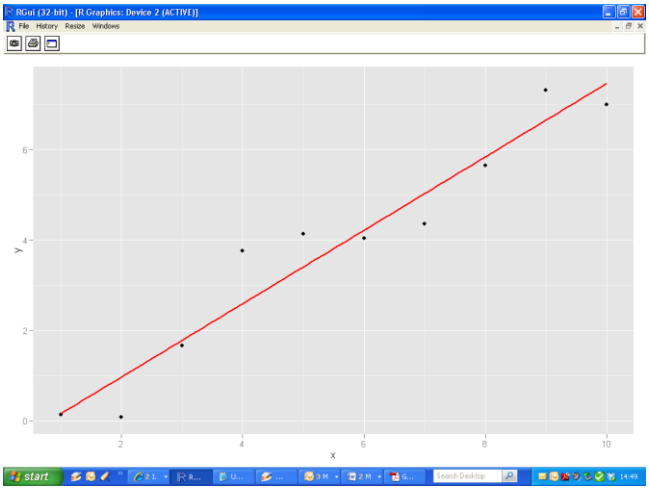
## Histograms

```
qplot(Sepal.Length, data=iris, geom="histogram")

# Warning Message:
# stat_bin: binwidth defaulted to range/30.
# Use 'binwidth = x' to adjust this.
```



## Linear regression Example

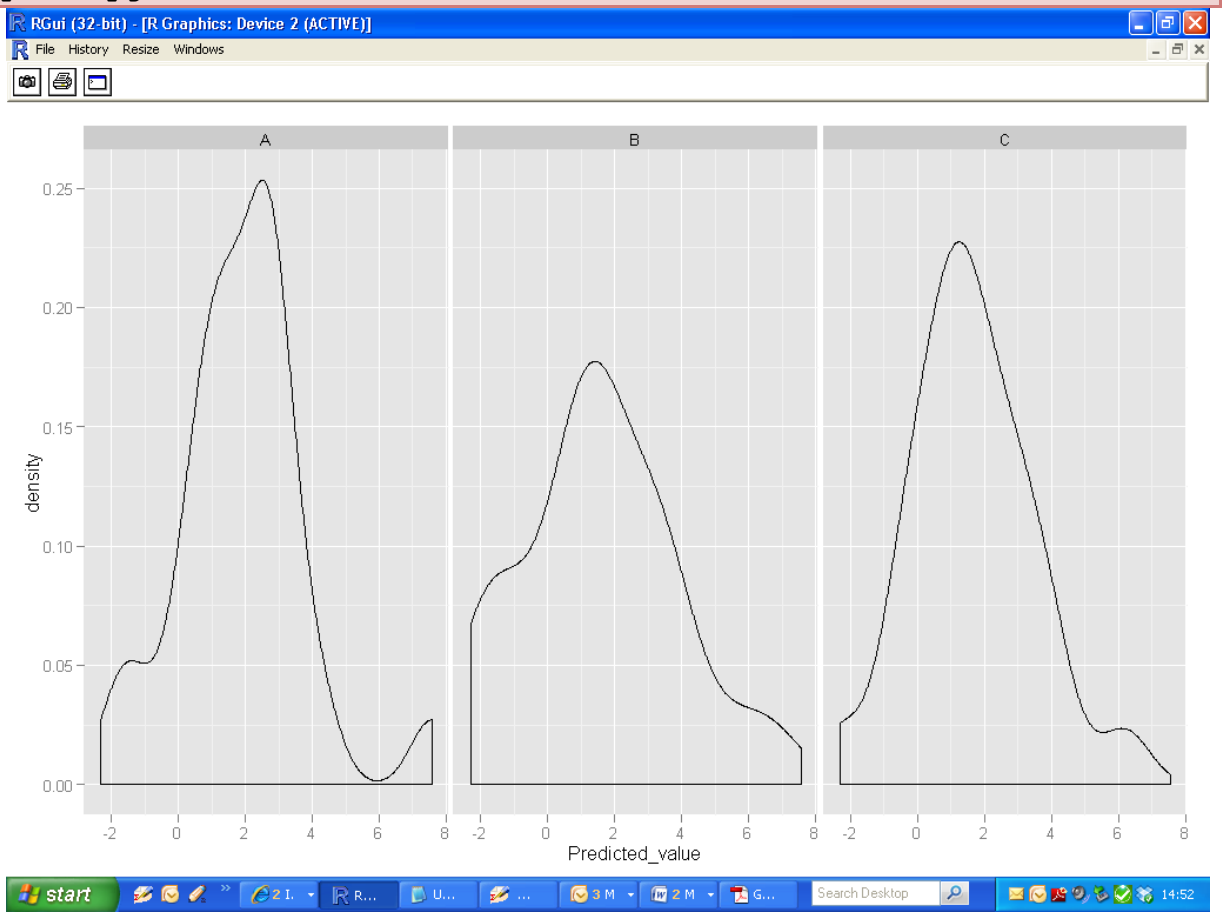
```
ddf <- data.frame(x = 1:10, y = 0.4 + 0.6 * (1:10) + rnorm(10))
# Find the linear model coefficients
lmc <- coef(lm(y ~ x, data = ddf))
# Create a function to produce the fitted line
lmeq <- function(x) lmc[1] + lmc[2] * x

# Construct the ggplot() and use stat_function():
ggplot(ddf, aes(x = x, y = y)) +
    geom_point() +
    stat_function(fun = lmeq, colour = 'red', size = 1)
```



## Stat_function

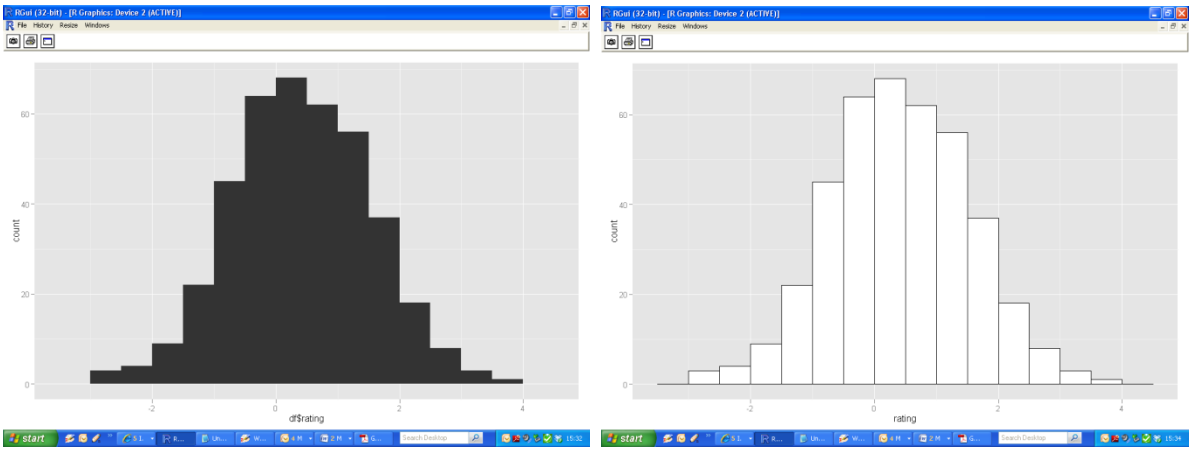
`stat_function` is designed to overlay the same function in every panel. (There's no obvious way tomatch up the parameters of the function with the different panels).

```
library(ggplot2)
#make some example data

dd<-data.frame(matrix(rnorm(144, mean=2,
sd=2),72,2),c(rep("A",24),rep("B",24),rep("C",24)))
colnames(dd) <- c("x_value", "Predicted_value",  "State_CD")

#This works

pg <- ggplot(dd) + geom_density(aes(x=Predicted_value)) +
facet_wrap(~State_CD)
print(pg)
```

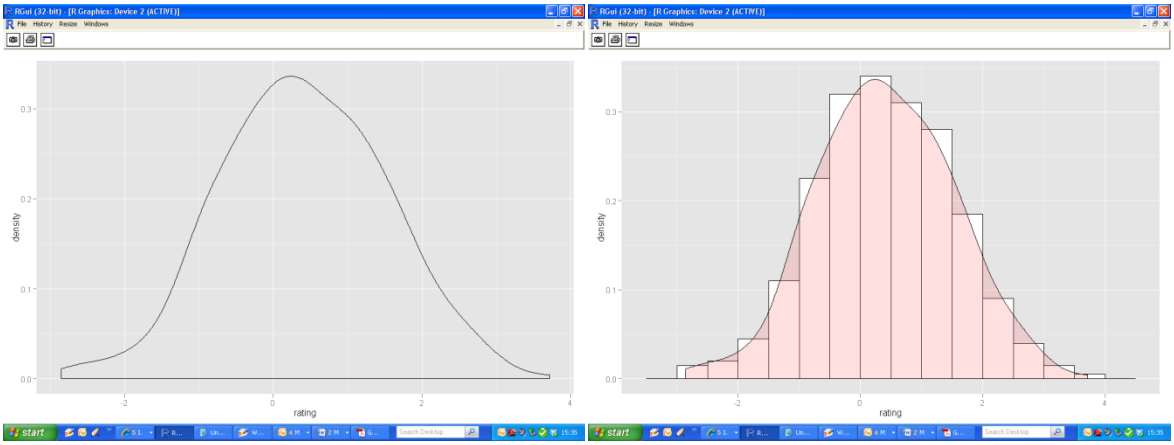


## Plotting distributions

```
set.seed(1234)
df <- data.frame(cond = factor( rep(c("A","B"), each=200) ),
                 rating = c(rnorm(200),rnorm(200, mean=.8)))

# Basic histogram from the vector "rating". Each bin is .5 wide.
# These both do the same thing:

qplot(df$rating, binwidth=.5)
ggplot(df, aes(x=rating)) + geom_histogram(binwidth=.5)
```



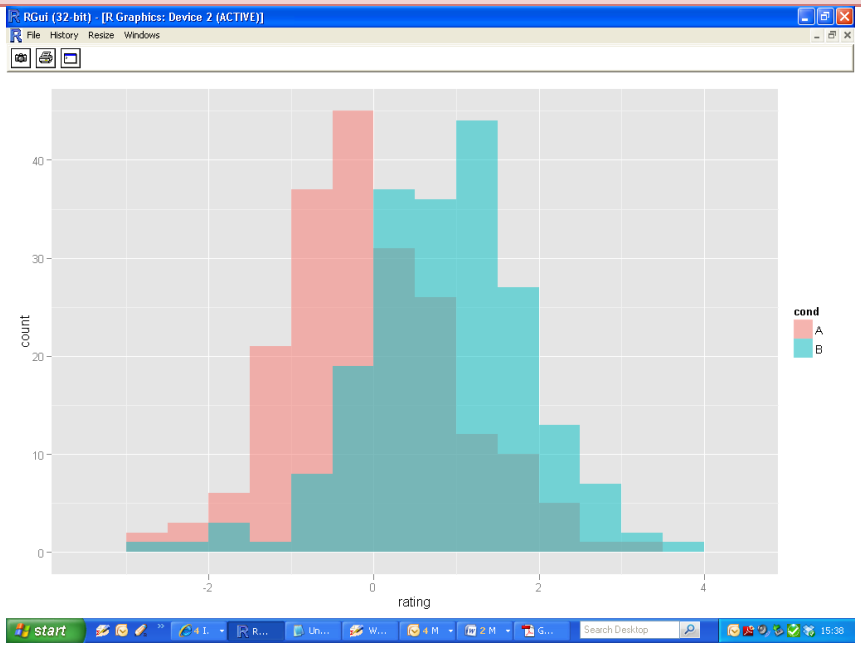

```
# Draw with black outline, white fill

ggplot(df, aes(x=rating)) + geom_histogram(binwidth=.5, colour="black",
fill="white")
```




```
# Density curve
ggplot(df, aes(x=rating)) + geom_density()

# Histogram overlaid with kernel density curve
ggplot(df, aes(x=rating)) +
    geom_histogram(aes(y=..density..),      # Histogram with density
instead of count on y-axis
                   binwidth=.5,
                   colour="black", fill="white") +
    geom_density(alpha=.2, fill="#FF6666")  # Overlay with transparent
density plot
```

## Histogram and density plots with multiple groups

```
# Overlaid histograms
ggplot(df, aes(x=rating, fill=cond)) + geom_histogram(binwidth=.5,
alpha=.5, position="identity")
```



```
# Density plots with semi-transparent fill
ggplot(df, aes(x=rating, fill=cond)) + geom_density(alpha=.3)
```