

multiplot

Chris Parrish

June 19, 2016

multiplot

reference: - multiplot

- multiplot

- multiplot

- cowplot

- themes in ggplot2

```
library(ggplot2)
# This example uses the ChickWeight dataset, which comes with ggplot2
# First plot
p1 <- ggplot(ChickWeight, aes(x=Time, y=weight, colour=Diet, group=Chick)) +
  geom_line() +
  ggtitle("Growth curve for individual chicks")

# Second plot
p2 <- ggplot(ChickWeight, aes(x=Time, y=weight, colour=Diet)) +
  geom_point(alpha=.3) +
  geom_smooth(alpha=.2, size=1) +
  ggtitle("Fitted growth curve per diet")

# Third plot
p3 <- ggplot(subset(ChickWeight, Time==21), aes(x=weight, colour=Diet)) +
  geom_density() +
  ggtitle("Final weight, by diet")

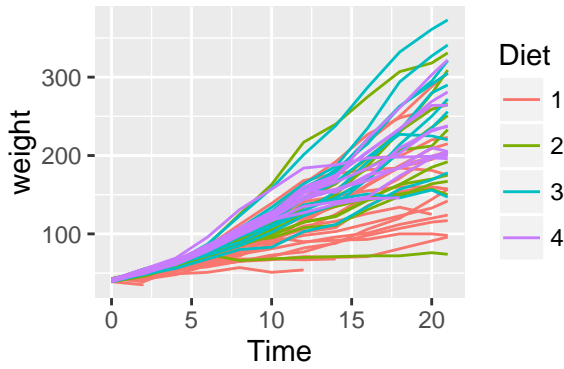
# Fourth plot
p4 <- ggplot(subset(ChickWeight, Time==21), aes(x=weight, fill=Diet)) +
  geom_histogram(colour="black", binwidth=50) +
  facet_grid(Diet ~ .) +
  ggtitle("Final weight, by diet") +
  theme(legend.position="none") # No legend (redundant in this graph)
```

Definition of multiplot

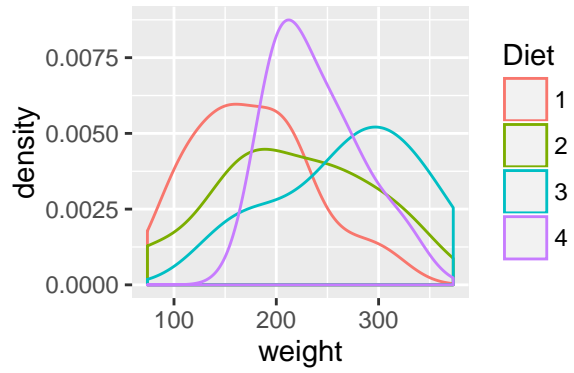
Note: grid is now a base package, so you do not have to download it from cran.

```
multiplot(p1, p2, p3, p4, cols=2)
```

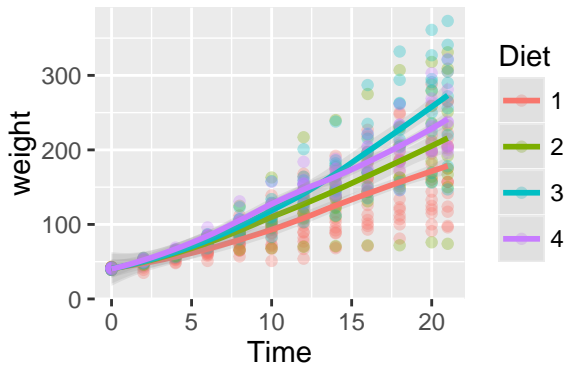
Growth curve for individual chicks



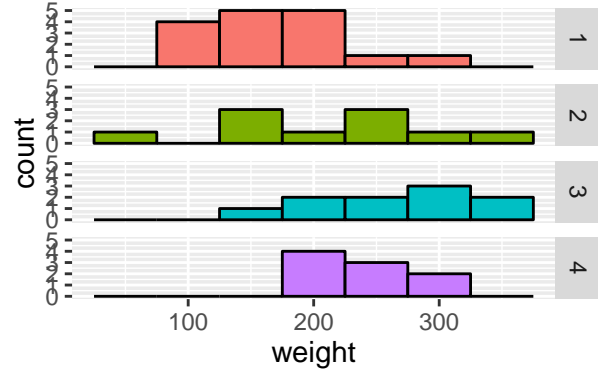
Final weight, by diet



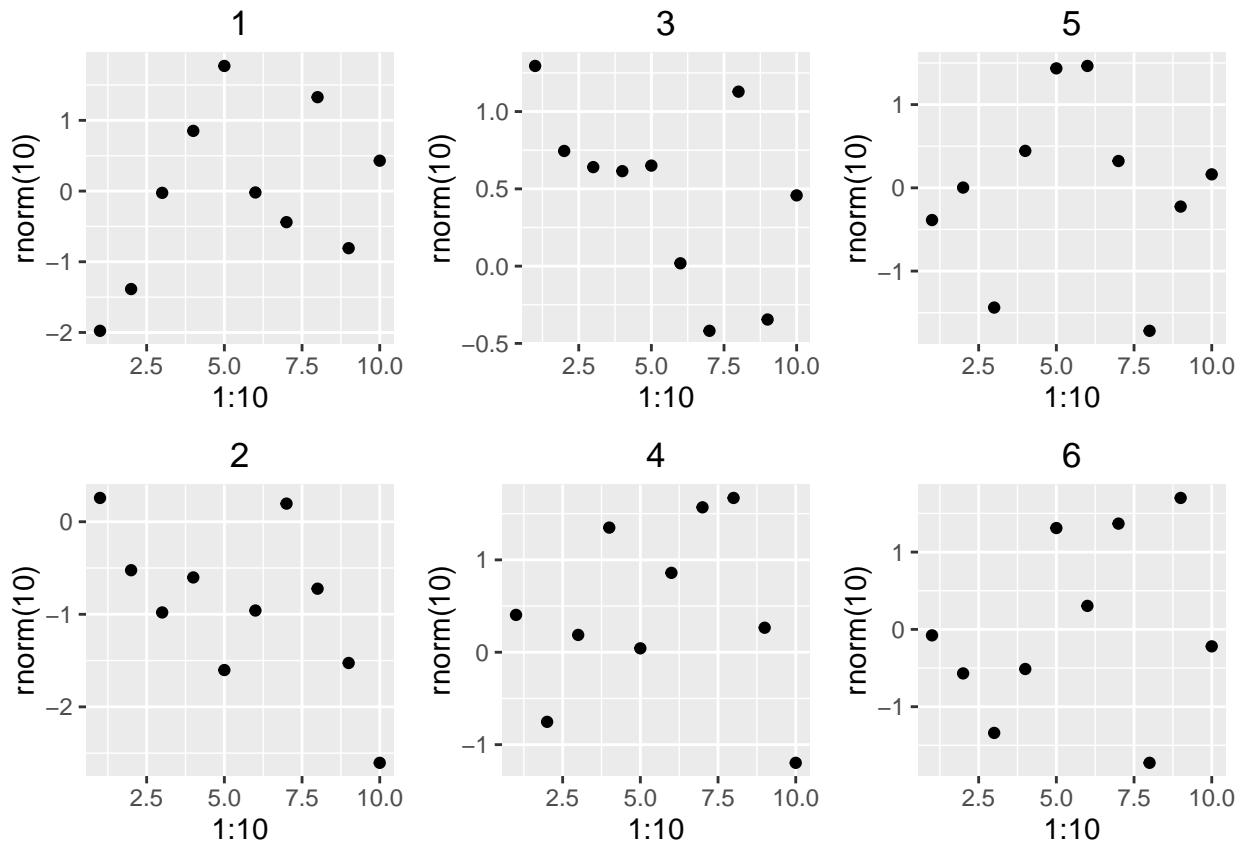
Fitted growth curve per diet



Final weight, by diet



```
plots <- list() # new empty list
for (i in 1:6) {
  p1 = qqplot(1:10, rnorm(10), main = i)
  plots[[i]] <- p1 # add each plot into plot list
}
multiplot(plotlist = plots, cols = 3)
```



Draw PI with ggplot2

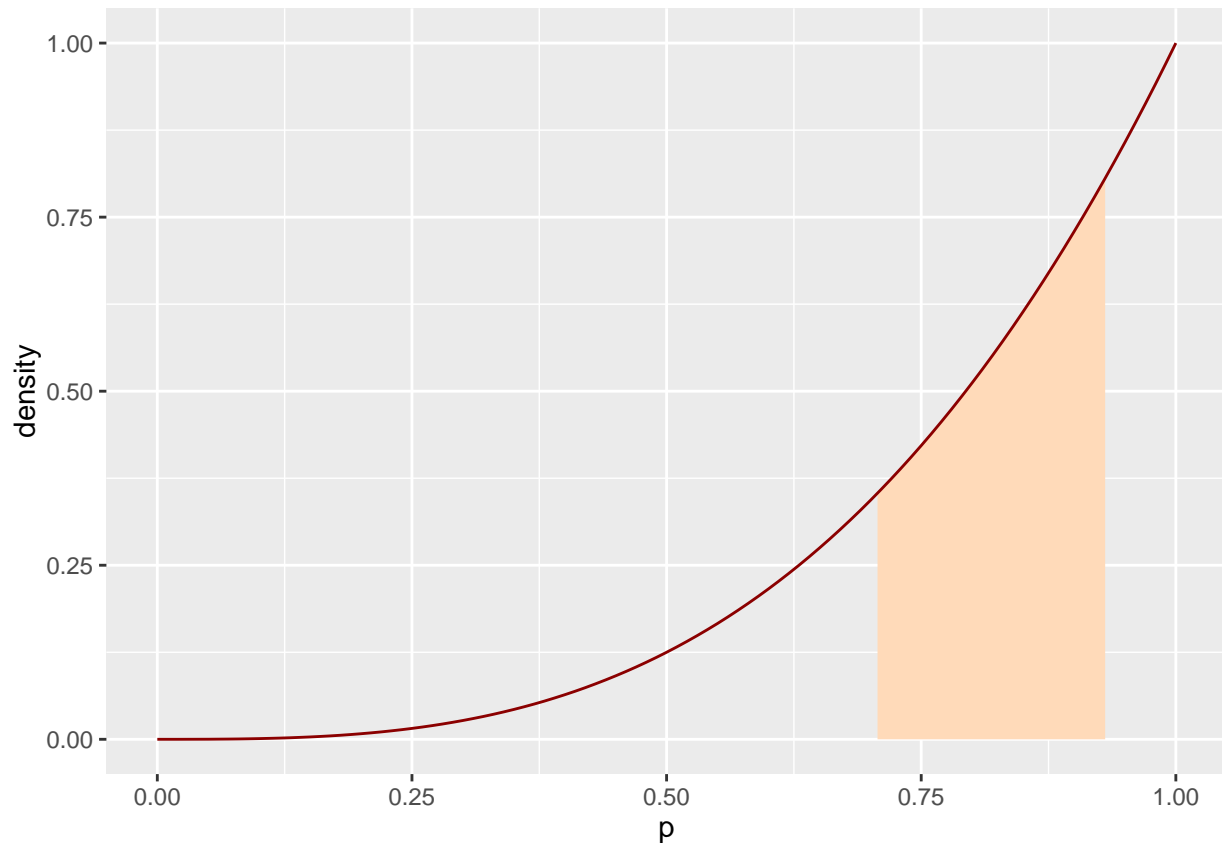
Want $q^4 == 0.75$.

Take $q = (0.75)^{1/4}$ $\ln(q) = (1/4) * \log(0.75)$ $q = \exp((1/4) * \log(0.75))$

```
# p cubed
data <- data.frame(x = c(0, 1), y = c(0, 1))
p.cubed <- function(p){
  return(p^3)
}
# polygon
p.min <- exp((1/4) * log(0.25))
p.max <- exp((1/4) * log(0.75))
round(data.frame(p.min = p.min, p.max = p.max), 3)

##  p.min p.max
## 1 0.707 0.931

xs <- seq(from=p.min, to=p.max, length.out=200)
ys <- p.cubed(xs)
poly.data <- data.frame(x = c(p.min, xs, p.max), y = c(0, ys, 0))
# plot
library(ggplot2)
PI.plot <- ggplot(data, aes(x, y)) +
  geom_polygon(poly.data, mapping = aes(x, y), fill = "peachpuff") +
  stat_function(fun = p.cubed, color = "darkred") +
  labs(x = "p", y = "density")
PI.plot
```



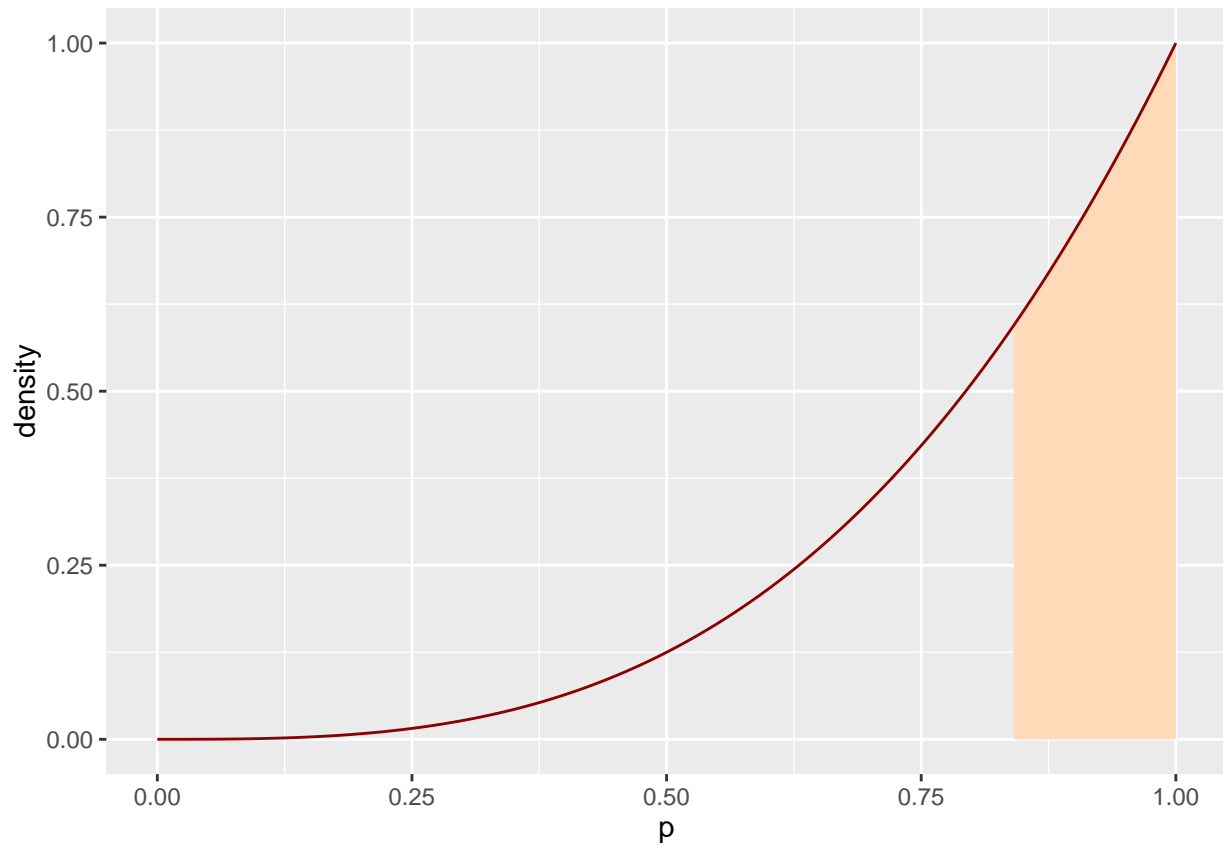
Draw HPDI with ggplot2

Want $q^4 == 0.50$.

```
# p cubed
data <- data.frame(x = c(0, 1), y = c(0, 1))
p.cubed <- function(p){
  return(p^3)
}
# polygon
p.min <- exp((1/4) * log(0.50))
p.max <- 1
round(data.frame(p.min = p.min, p.max = p.max), 3)
```

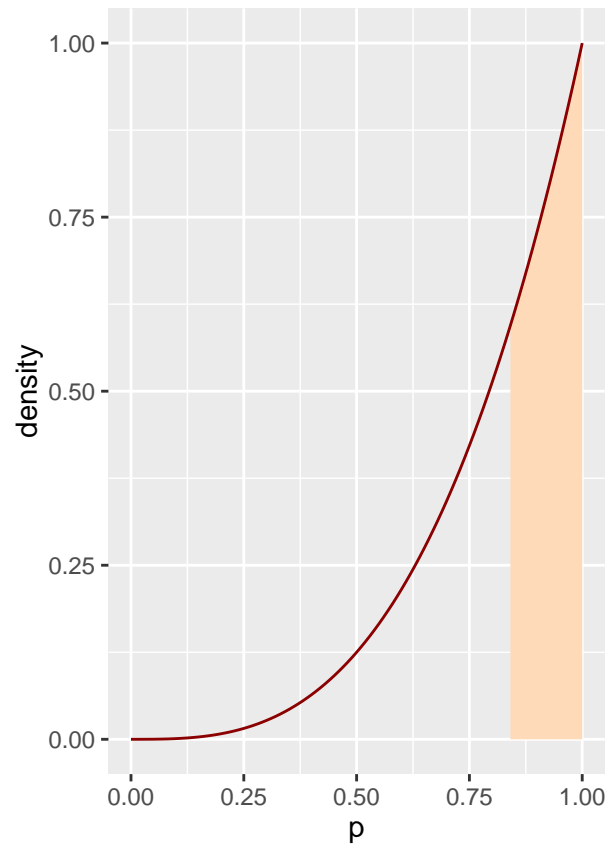
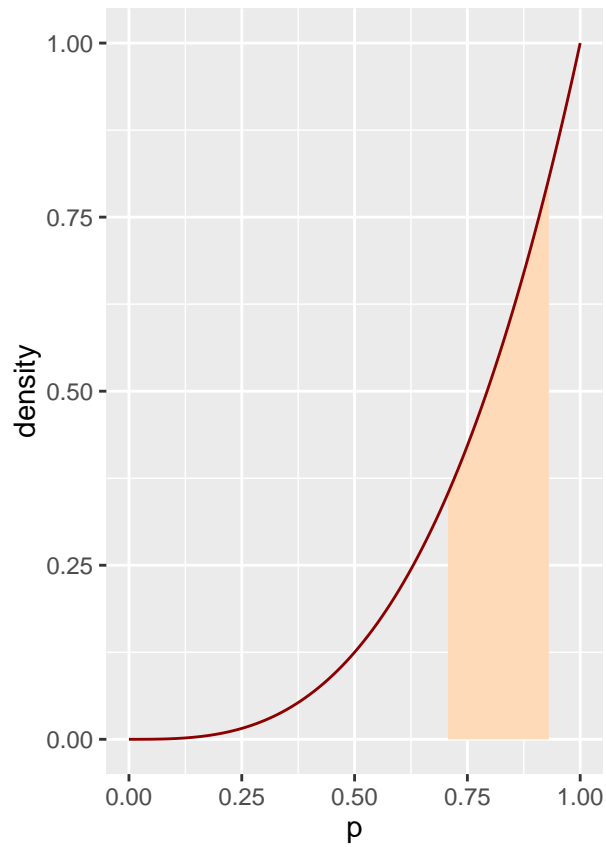
```
##  p.min p.max
## 1 0.841    1
```

```
xs <- seq(from=p.min, to=p.max, length.out=200)
ys <- p.cubed(xs)
poly.data <- data.frame(x = c(p.min, xs, p.max), y = c(0, ys, 0))
# plot
library(ggplot2)
HDPI.plot <- ggplot(data, aes(x, y)) +
  geom_polygon(poly.data, mapping = aes(x, y), fill = "peachpuff") +
  stat_function(fun = p.cubed, color = "darkred") +
  labs(x = "p", y = "density")
HDPI.plot
```



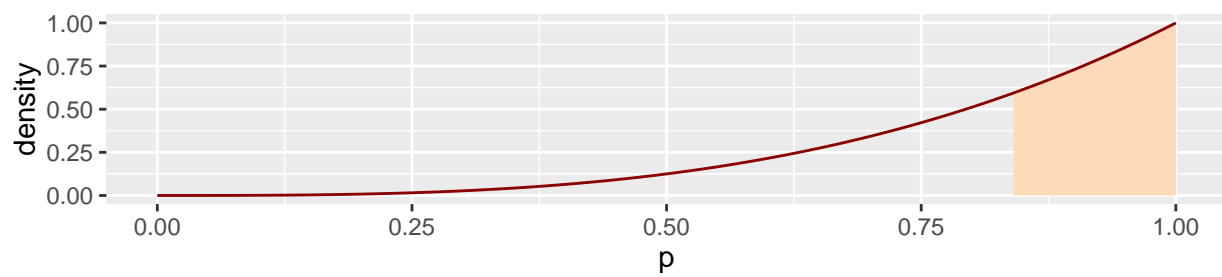
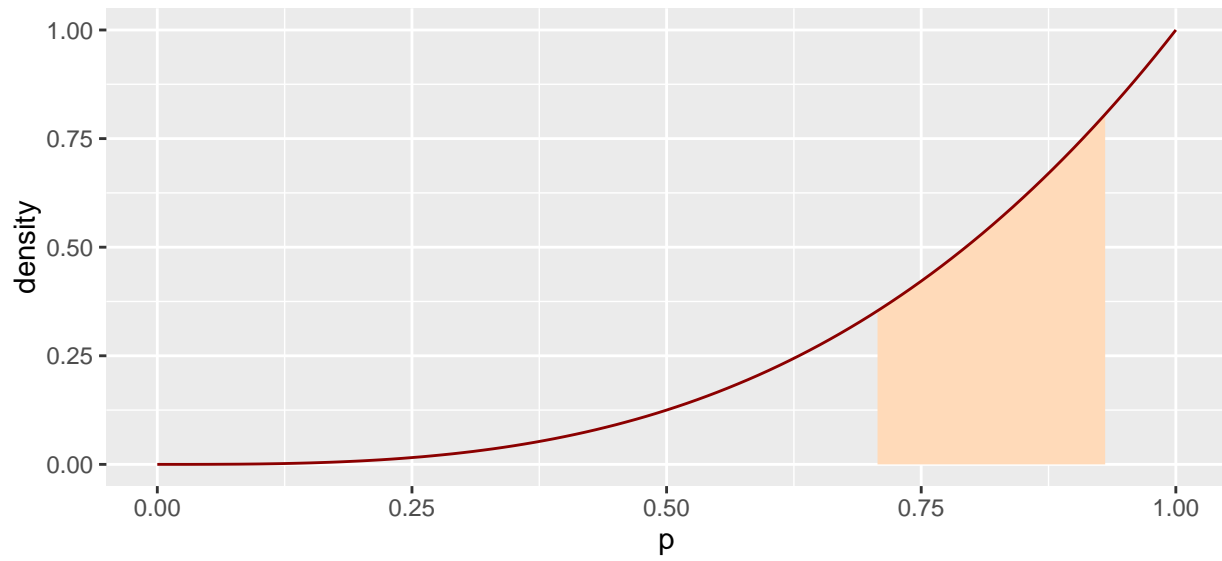
Show both.

```
multiplot(PI.plot, HDPI.plot, cols=2)
```

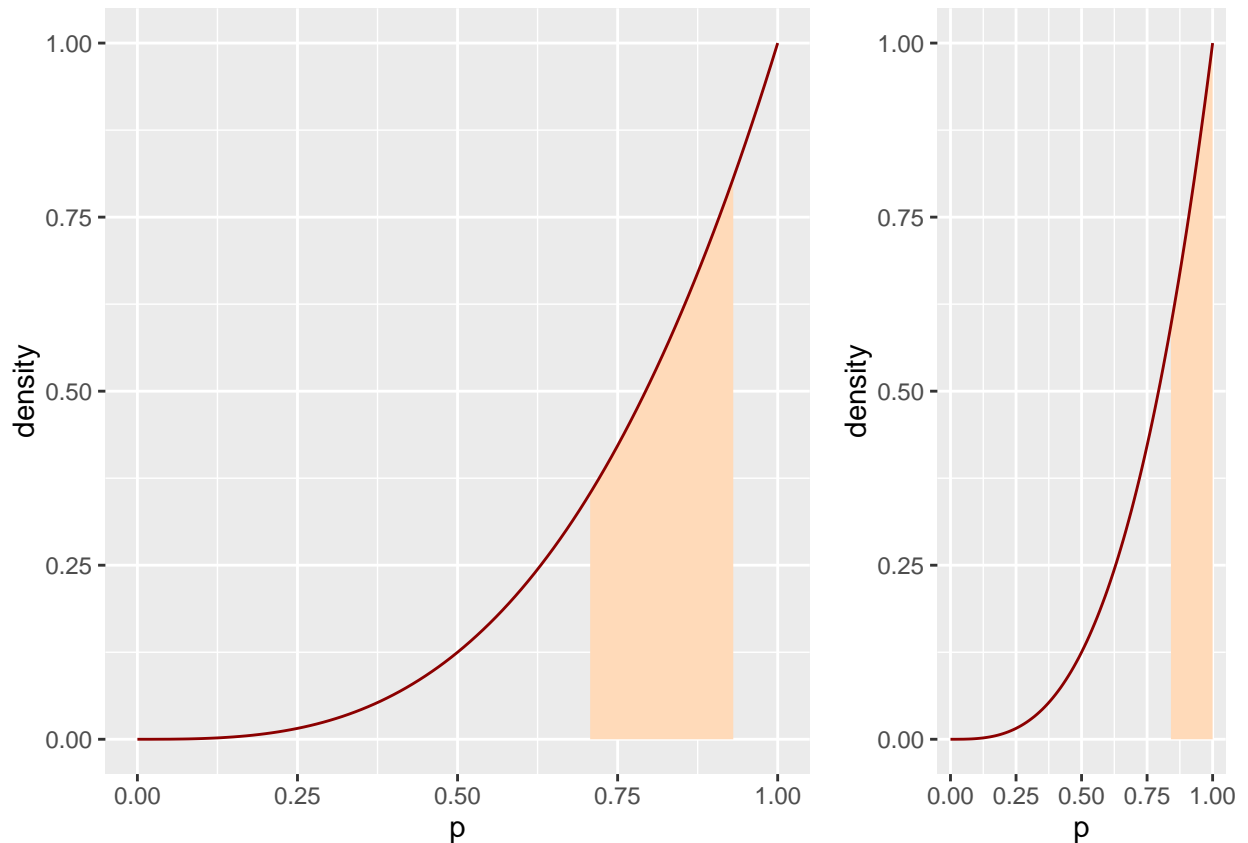


Use layout to control placement.

```
# image1 in first two rows
layout <- matrix(c(rep(1, 6), rep(2, 3)), nrow = 3, byrow = TRUE)
multiplot(PI.plot, HDPI.plot, layout = layout)
```



```
# image1 in first two cols  
layout <- matrix(rep(c(1, 1, 2), 3), nrow = 3, byrow = TRUE)  
multiplot(PI.plot, HDPI.plot, layout = layout)
```



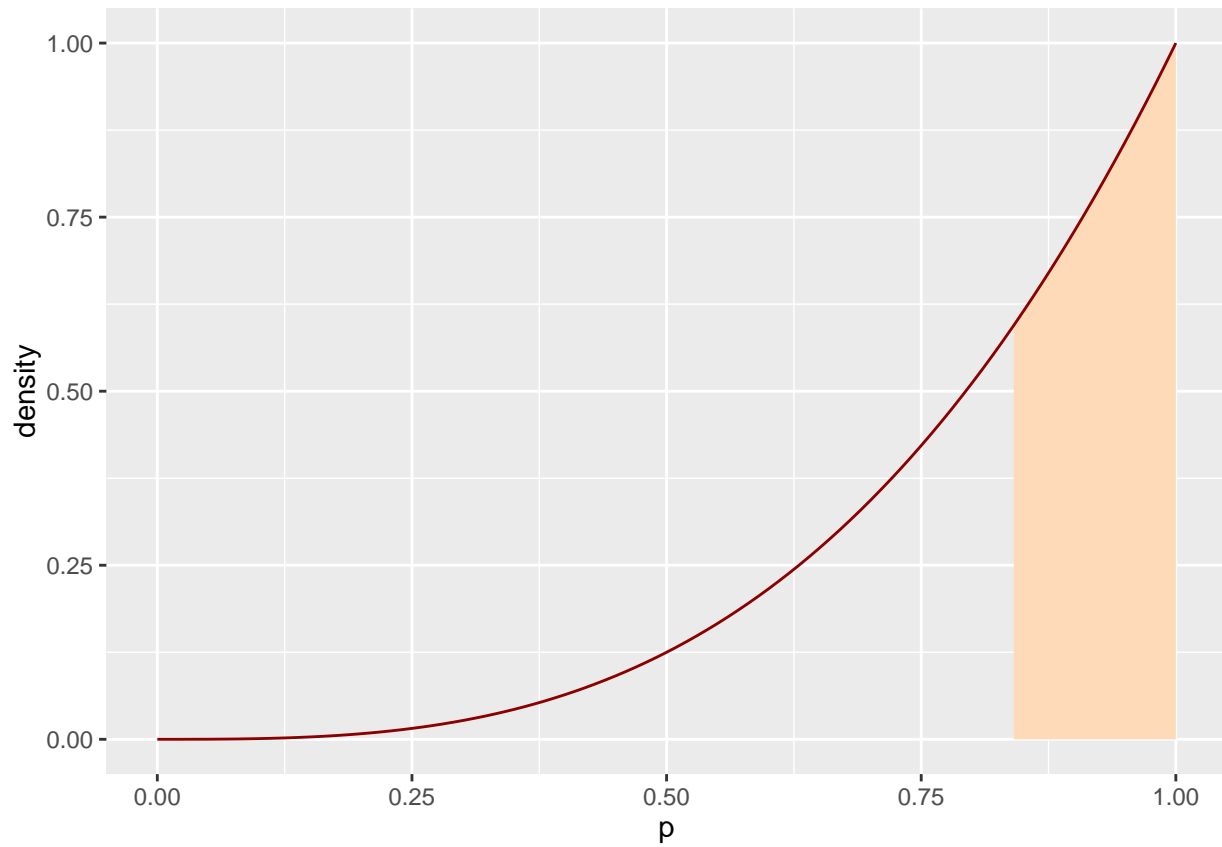
```
# image1 in first row, then image1 and image2 in second row
```

To recover `theme_gray` with `cowplot` use `theme_gray()` in `ggplot` Results in a better font for axis ticks as well.

```
# p cubed
data <- data.frame(x = c(0, 1), y = c(0, 1))
p.cubed <- function(p){
  return(p^3)
}
# polygon
p.min <- exp((1/4) * log(0.50))
p.max <- 1
round(data.frame(p.min = p.min, p.max = p.max), 3)

##   p.min p.max
## 1 0.841    1

xs <- seq(from=p.min, to=p.max, length.out=200)
ys <- p.cubed(xs)
poly.data <- data.frame(x = c(p.min, xs, p.max), y = c(0, ys, 0))
# plot
HDPI.plot <- ggplot(data, aes(x, y)) +
  geom_polygon(poly.data, mapping = aes(x, y), fill = "peachpuff") +
  stat_function(fun = p.cubed, color = "darkred") +
  labs(x = "p", y = "density") +
  theme_gray()
HDPI.plot
```

Show both plots.

```
library(cowplot)
plot_grid(PI.plot, HDPI.plot, labels=c("PI", "HDPI"), ncol = 2, nrow = 1)
```

