Once in a while, you might find yourself wanting to embed one plot within another plot. ggplot2 makes this really easy with the annotation\_custom function. The following example illustrates how you can achieve this.

Let’s generate some random data and make a scatterplot along with a smoothed estimate of the relationship:

library(ggplot2)

set.seed(42)

n <- 1000

x <- runif(n) \* 3

y <- x \* sin(1/x) + rnorm(n) / 25

df <- data.frame(x = x, y = y)

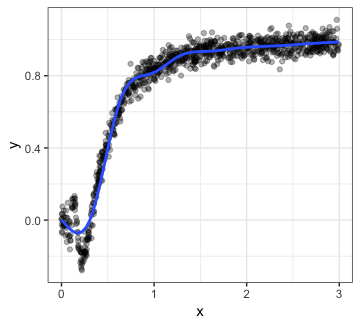
p1 <- ggplot(df, aes(x, y)) +

geom\_point(alpha = 0.3) +

geom\_smooth(se = FALSE) +

theme\_bw()

p1



The smoother seems to be doing a good job of capturing the relationship for most of the plot, but it looks like there’s something more going on in the x \in [0, 0.5]region. Let’s zoom in:

p2 <- ggplot(df, aes(x, y)) +

geom\_point(alpha = 0.3) +

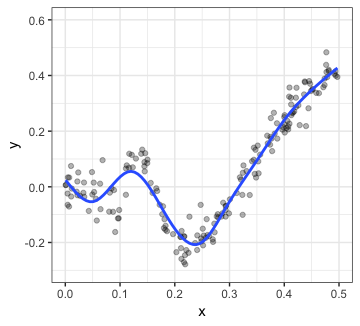
geom\_smooth(se = FALSE) +

scale\_x\_continuous(limits = c(0, 0.5)) +

scale\_y\_continuous(limits = c(-0.3, 0.6)) +

theme\_bw()

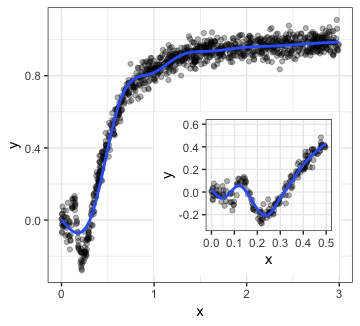
p2



That certainly seems like a meaningful relationship! While we might want to plot p1 to depict the overall relationship, it is probably a good idea to show p2 as well. This can be achieved very easily:

p1 + annotation\_custom(ggplotGrob(p2), xmin = 1, xmax = 3,

ymin = -0.3, ymax = 0.6)



The first argument is for annotation\_custom must be a “grob” (what is a grob? see details [here](https://stat.ethz.ch/R-manual/R-devel/library/grid/html/grid.grob.html)) which we can create using the ggplotGrob function. The 4 other arguments (xmin etc.) indicate the coordinate limits for the inset: these coordinates are with reference to the axes of the outer plot. As explained in the documentation, the inset will try to fill up the space indicated by these 4 arguments while being center-justified.

For ggmap objects, we need to use inset instead of annotation\_custom. We illustrate this by making a map of continental USA with insets for Alaska and Hawaii.

Let’s get a map of continental US:

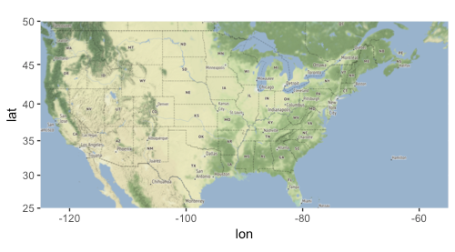
library(ggmap)

us\_bbox <- c(left = -125, bottom = 25, right = -55, top = 50)

us\_main\_map <- get\_stamenmap(us\_bbox, zoom = 5, maptype = "terrain")

p\_main <- ggmap(us\_main\_map)

p\_main



Next, let’s get maps for Alaska and Hawaii and save them into R variables. Each plot will have a title for the state, and information on the axes will be removed.

alaska\_bbox <- c(left = -180, bottom = 50, right = -128, top = 72)

alaska\_map <- get\_stamenmap(alaska\_bbox, zoom = 5, maptype = "terrain")

p\_alaska <- ggmap(alaska\_map) +

labs(title = "Alaska") +

theme(axis.title = element\_blank(),

axis.text = element\_blank(),

axis.ticks = element\_blank())

p\_alaska

hawaii\_bbox <- c(left = -160, bottom = 18.5, right = -154.5, top = 22.5)

hawaii\_map <- get\_stamenmap(hawaii\_bbox, zoom = 6, maptype = "terrain")

p\_hawaii <- ggmap(hawaii\_map) +

labs(title = "Hawaii") +

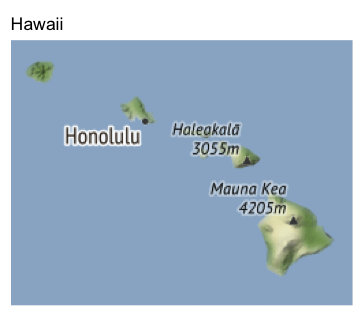
theme(axis.title = element\_blank(),

axis.text = element\_blank(),

axis.ticks = element\_blank())

p\_hawaii





We can then use inset twice to embed these two plots (I had to fiddle around with the xmin etc. options to get it to come out right):

library(grid)

p\_main +

inset(ggplotGrob(p\_alaska), xmin = -76.7, xmax = -66.7, ymin = 26, ymax = 35) +

inset(ggplotGrob(p\_hawaii), xmin = -66.5, xmax = -55.5, ymin = 26, ymax = 35)

