

Section 5.5:

Exercises: 1 & 3 (plus a 90% CI for the slope)

- First, assign your data to vectors `x` & `y` (not necessary, but saves typing later)

```
x<-cig$Nic.Wt
y<-cig$CO.Wt
or
x<-c(...)
y<-c(...)
```

- Create your vector for storing all `n` choose 2 slopes:

```
slopes<-c()
```

- Make a scatterplot of the data:

```
plot(x,y, pch=18,xlab="Nic/Wt", ylab="CO/wt")
```

And add least squares lines to each pair of points. Note that when you only have a pair of points, the least squares line will be the same as the slope between those two points. The least squares is given by the function `lm(y~x)` which says to regress `Y` on `X` in a linear model.

```
for(i in 1:24){
  for(j in (i+1):25){
    temp<-lm(c(y[i],y[j])~c(x[i],x[j]))
    abline(temp,col="lightgray")
    slopes<-c(slopes,temp$coef[2])
  }}
```

- To enhance the plot, add the points back and draw the least squares and median regression lines. (Note: `abline` adds a line to your plot in any of a variety of ways. Two of those ways include: you can give it linear regression output or you can tell it the slope (b) and the intercept (a).)

```
points(x,y, pch=18)
abline(lm(y~x),col="orangered")
abline(a=median(y)-median(slopes)*median(x),b=median(slopes),col="purple")
```

- To find the slopes of interest, use the `quantile` function to find the median or percentile cutoffs; use the `sort` function to find the  $k^{th}$  slope of interest.

```
quantile(slopes)
sort(slopes)[c(108,192)]
```