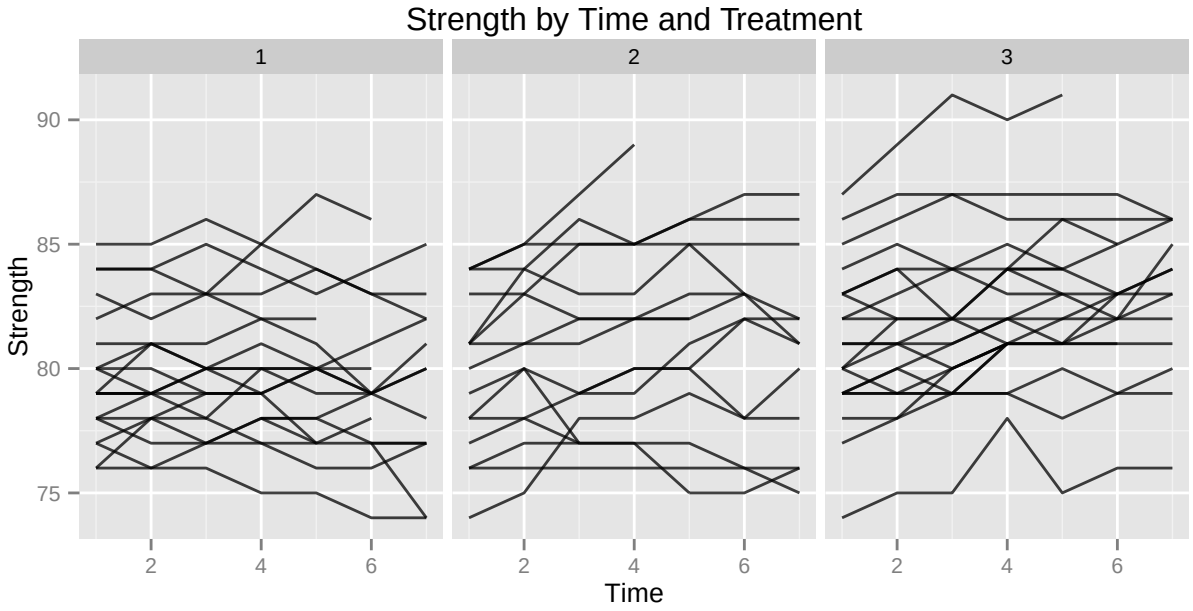
Assignment 3

John Semerdjian

October 19, 2015

1. On one plot, plot separate line plots of all subjects' trajectories of strength versus week.

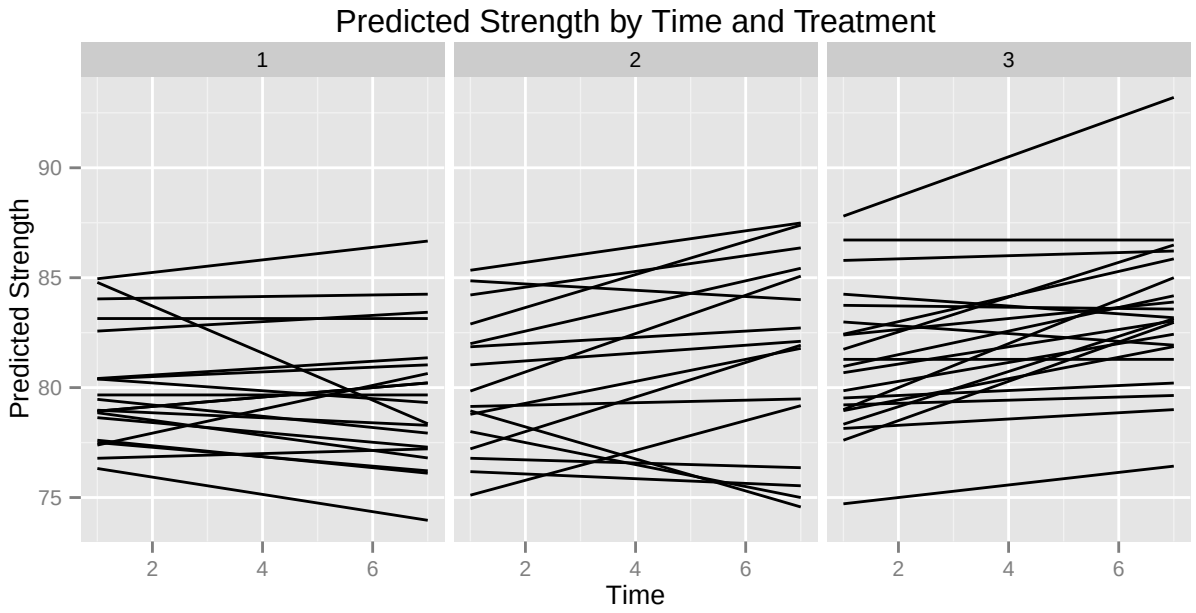
```
library(ggplot2)
options(warn=-1)
df = read.csv("../data/strength.csv", stringsAsFactors=F, header=T)
df$tx = as.factor(df$tx)

ggplot(df, aes(x=time, y=y, group=as.factor(id))) +
  geom_line(alpha=0.75) + facet_wrap(~tx) +
  theme(text = element_text(size=10)) +
  xlab("Time") + ylab("Strength") +
  ggtitle("Strength by Time and Treatment")
```



2. Fit a linear regression by subject, and repeat 1, replacing the strength with predicted strength based on these regressions

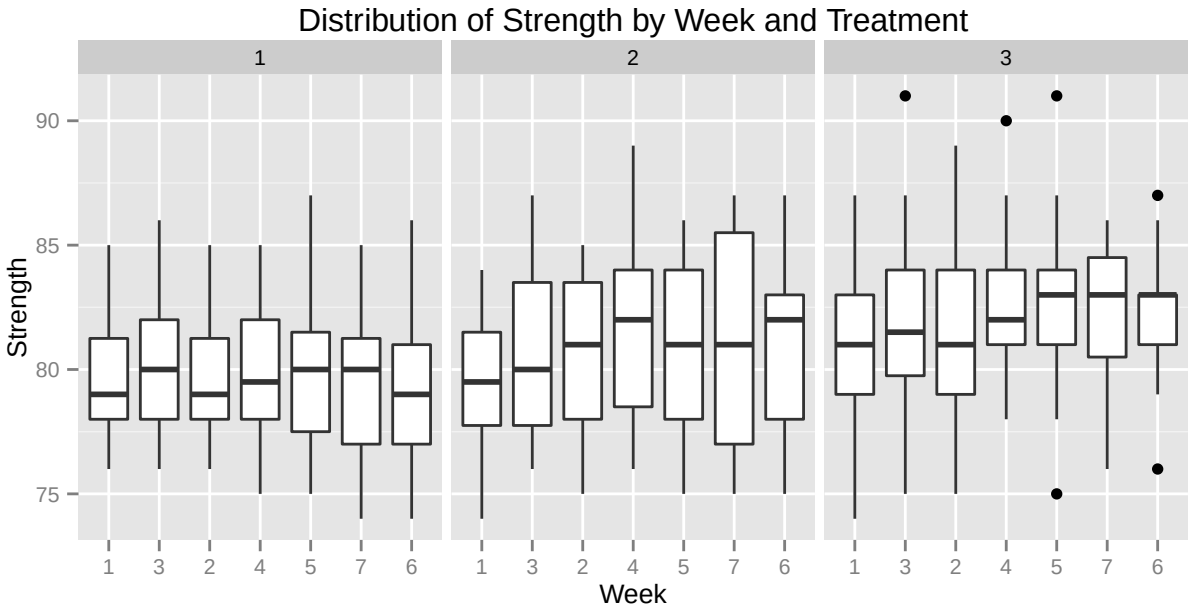
```
reg_pred = function(data, x) {
  m = lm(y ~ time, data = df[df$id == x,])
  p = predict(m, df[df$id == x,])
  return(p)
}

df$pred = unlist(lapply(unique(df$id), function(x) reg_pred(data, x)))

ggplot(df, aes(x=time, y=pred, group=as.factor(id))) +
  geom_line() + facet_wrap(~tx) +
```

```
  theme(text = element_text(size=10)) +
  xlab("Time") + ylab("Predicted Strength") +
  ggtitle("Predicted Strength by Time and Treatment")
```



3. Optional: Consider whether there are other plots that might be useful, and include one more if you want to. (For example, box plots of the distribution of strength by both week and tx

```
ggplot(df, aes(x = reorder(as.factor(time), y, FUN=median, na.rm=T), y = y)) +
  geom_boxplot() + facet_grid(.~tx, space="free", scales="free") +
  theme(text = element_text(size=10)) +
  xlab("Week") + ylab("Strength") +
  ggtitle("Distribution of Strength by Week and Treatment")
```



4. Write a short paragraph describing what the plots suggest about the bulleted questions above

    - Does weight training have any impact on strength? Yes, it appears that there is an effect due to strength training.
    - Is there a difference between tx 2 and 3? The predicted strength line chart seems to show a difference between 2 and 3, but it also looks like treatment 3 may have had greater strength to begin with.
    - Which training program works quickest to increase strength? The boxplots and line charts (both true and predicted) seem to suggest a greater improvement in strength over time due to treatment 3. It seems like treatment 2 has some effect when compared to treatment 1 from the box plots, but not as large as treatment 3.

5. Reduce the data for each subject to 1 number – the slope of the change in strength estimated in each person separately (can use program below). One way you could test the treatment effect of tx=3 versus tx=2 on this outcome (ignoring tx=1) is to use a standard two-sample t-test.

```
reg_slope = function(data, x) {
  m = lm(y ~ time, data = df[df$id == x,])
  return(m$coeff[2])
}

slopes = unlist(sapply(unique(df$id), function(x) reg_slope(data, x)))
df_slopes = df[!duplicated(df[,c("id","tx")]), c("id","tx")]
df_slopes$slopes = slopes
tx2 = df_slopes[df_slopes$tx == 2,]
tx3 = df_slopes[df_slopes$tx == 3,]
t.test(tx2$slopes, tx3$slopes, paired = FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  tx2$slopes and tx3$slopes
## t = -0.83772, df = 28.048, p-value = 0.4093
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.4087493  0.1714503
## sample estimates:
## mean of x mean of y
## 0.2313342 0.3499837
```

6. To further explore the relationship between treatment and group, try fitting a simple cross sectional model to this data as discussed in Chapter 3. Look the model with both naïve and robust standard errors. What happens if you expand the model to include a longitudinal term?

    I created my naive model using treatment ("tx") and cross-sectional time ("time") variables. In my longitudinal model, I added a longitudinal term ("timediff"). This longitudinal term was automatically removed by the model since it is collinear with the cross-sectional "time" variable. The coefficient estimates between naive and robust regressions are the same, while the standard errors are slightly different, but not by much. The estimate of "time" is small and not significant.

```r
library(sandwich)
library(lmtest)

# naive model
naive_model = lm(y ~ tx + time, data = df)
summary(naive_model)$coef
```

```
##               Estimate Std. Error    t value     Pr(>|t|)
## (Intercept) 79.3613137 0.42641713 186.111925 0.000000e+00
## tx2          0.9534016 0.41628587   2.290257 2.257462e-02
## tx3          2.2953295 0.39048914   5.878088 9.348868e-09
## time         0.1139682 0.08378795   1.360198 1.746044e-01
```

```r
# This is the default method for robust SE in Stata per:
# https://stats.stackexchange.com/questions/117052/replicating-statas-robust-option-in-r
coeftest(naive_model, vcov = vcovHC(naive_model, "HC1"))
```

```
##
## t test of coefficients:
##
##              Estimate Std. Error  t value  Pr(>|t|)
## (Intercept) 79.361314   0.393598 201.6302 < 2.2e-16 ***
## tx2          0.953402   0.430861   2.2128   0.02753 *
## tx3          2.295330   0.366188   6.2682 1.028e-09 ***
## time         0.113968   0.083655   1.3624   0.17392
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# longitudinal model
df$timediff = df$time - 1
longitudinal_model = lm(y ~ tx + time + timediff, data = df)
summary(longitudinal_model)$coef
```

```
##               Estimate Std. Error    t value     Pr(>|t|)
## (Intercept) 79.3613137 0.42641713 186.111925 0.000000e+00
## tx2          0.9534016 0.41628587   2.290257 2.257462e-02
## tx3          2.2953295 0.39048914   5.878088 9.348868e-09
## time         0.1139682 0.08378795   1.360198 1.746044e-01
```

```r
# robust SE
coeftest(longitudinal_model, vcov = vcovHC(longitudinal_model, "HC1"))
```

```
##
## t test of coefficients:
##
##              Estimate Std. Error  t value  Pr(>|t|)
## (Intercept) 79.361314   0.393598 201.6302 < 2.2e-16 ***
## tx2          0.953402   0.430861   2.2128   0.02753 *
## tx3          2.295330   0.366188   6.2682 1.028e-09 ***
## time         0.113968   0.083655   1.3624   0.17392
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```