Homework 1

Leah Schultz

9/28/2017

```
## Chapter 2: LDA Basics
library(dplyr)
library(tidyr)
library(ggplot2)
purpose <- read.csv("~/Dropbox/Lab & Research/OYSUP Project/oysup_self.csv")
```

1. Move your data into a long format and a wide format. Did you have any specific challenges that you encountered? If so, discuss them.

```
purpose_long <- tbl_df(purpose) %>%
  gather(-c(FAMID, SEX2, MEDUC2, MPEDUC2), key = "grade", value = "value") %>%
  separate(grade, into = c("variable", "grade"), sep = "_", convert = T) %>%
  spread(variable, value)
purpose_long
```

```
## # A tibble: 6,444 x 33
##    FAMID  SEX2 MEDUC2 MPEDUC2 grade cbdad cbmom DID15 DID27 DID31 DID33
##  * <int> <int>  <int>   <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
##  1  1001     2      2       3     1    NA    NA    NA    NA    NA    NA
##  2  1001     2      2       3     2    NA    NA    NA    NA    NA    NA
##  3  1001     2      2       3     3    NA    NA    NA    NA    NA    NA
##  4  1001     2      2       3     4    NA    NA    NA    NA    NA    NA
##  5  1001     2      2       3     5     2     4    NA    NA    NA    NA
##  6  1001     2      2       3    21    NA    NA     5     5     5     5
##  7  1002     2      2       3     1    NA    NA    NA    NA    NA    NA
##  8  1002     2      2       3     2     2     3    NA    NA    NA    NA
##  9  1002     2      2       3     3     2     2    NA    NA    NA    NA
## 10  1002     2      2       3     4     2     2    NA    NA    NA    NA
## # ... with 6,434 more rows, and 22 more variables: LDS01 <dbl>,
## #   LDS02 <dbl>, LDS03 <dbl>, LDS04 <dbl>, LDS05 <dbl>, LDS06 <dbl>,
## #   LDS07 <dbl>, LDS08 <dbl>, LDS09 <dbl>, LDS10 <dbl>, LDS11 <dbl>,
## #   LDS12 <dbl>, LDS13 <dbl>, LDS14 <dbl>, LDS15 <dbl>, lifesat <dbl>,
## #   PSS01R <dbl>, PSS02 <dbl>, PSS03 <dbl>, PSS04R <dbl>, purpose <dbl>,
## #   stress <dbl>
```

```
purpose_wide <- tbl_df(purpose_long) %>%
  gather(-c(FAMID, SEX2, MEDUC2, MPEDUC2, grade), key = "variable", value = "value") %>%
  unite(VarG, variable, grade)  %>%
  spread(key = VarG, value = value) %>%
  select_if(~sum(!is.na(.)) > 0)
purpose_wide
```

```
## # A tibble: 1,074 x 40
##    FAMID  SEX2 MEDUC2 MPEDUC2 cbdad_1 cbdad_2 cbdad_3 cbdad_4 cbdad_5
##  * <int> <int>  <int>   <int>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
##  1  1001     2      2       3      NA      NA      NA      NA       2
##  2  1002     2      2       3      NA       2       2        2       1
##  3  1003     1      3       3      NA      NA      NA        2       6
##  4  1004     2      2       4      NA      NA       0        0       0
##  5  1005     1      3       2      NA      NA      NA       NA       1
##  6  1006     1      3       3      NA      NA      NA        1       2
##  7  1007     2      3       3      NA       0       4        3       0
##  8  1008     2      2       1      NA      NA      NA        2       5
##  9  1009     2      3       4      NA      NA       1        4       3
## 10  1010     1      4       1      NA      NA      NA        5       4
## # ... with 1,064 more rows, and 31 more variables: cbmom_1 <dbl>,
## #   cbmom_2 <dbl>, cbmom_3 <dbl>, cbmom_4 <dbl>, cbmom_5 <dbl>,
## #   DID15_21 <dbl>, DID27_21 <dbl>, DID31_21 <dbl>, DID33_21 <dbl>,
## #   LDS01_21 <dbl>, LDS02_21 <dbl>, LDS03_21 <dbl>, LDS04_21 <dbl>,
## #   LDS05_21 <dbl>, LDS06_21 <dbl>, LDS07_21 <dbl>, LDS08_21 <dbl>,
## #   LDS09_21 <dbl>, LDS10_21 <dbl>, LDS11_21 <dbl>, LDS12_21 <dbl>,
## #   LDS13_21 <dbl>, LDS14_21 <dbl>, LDS15_21 <dbl>, lifesat_21 <dbl>,
## #   PSS01R_21 <dbl>, PSS02_21 <dbl>, PSS03_21 <dbl>, PSS04R_21 <dbl>,
## #   purpose_21 <dbl>, stress_21 <dbl>
```

Challenges: First I forgot to exclude the ID variable and stable demographics, so R tried to make it into a value. I had a lot of variables that had repeated measures, so I had to think about how to split them after I gathered everything. Also, my variables were not consistently named because I was mixing naming conventions (my preferred conventions, and then the ones that OPP used). I went in and cleaned up my file a lot more so that I could use the separate function easily in the next step.

Another thing that was difficult was I ended up with some NA columns when I spread my data back to wide format – the drop and fill arguments didn't seem to help, so I had to find a solution for how to drop the NA columns from the key-pair combinations that didn't exist (for example, purpose wasn't assessed at grade 1).

2. Create a wave variable and date variable (if applicable).

I already have a grade variable, which is equivalent to wave for my purposes, and do not have dates available beyond year, which is not very useful.

3. What is your sample size for each wave of assessment?

```r
purpose_long %>%
  group_by(grade) %>%
  filter(!is.na(cbmom)) %>%
  count()
```

```
## # A tibble: 5 x 2
## # Groups:   grade [5]
##   grade     n
##   <int> <int>
## 1     1   220
## 2     2   408
## 3     3   606
## 4     4   806
## 5     5   994
```

4. Take the date variable and convert it to a different date format such as time in study or age (if appropriate). What scale is most suitable for your analyses? (weeks/months/years?)

Not applicable for my analyses.

5. Graph your data using the different time metrics, fitting individual curves for each person.

Needed to drop variables at age 21:

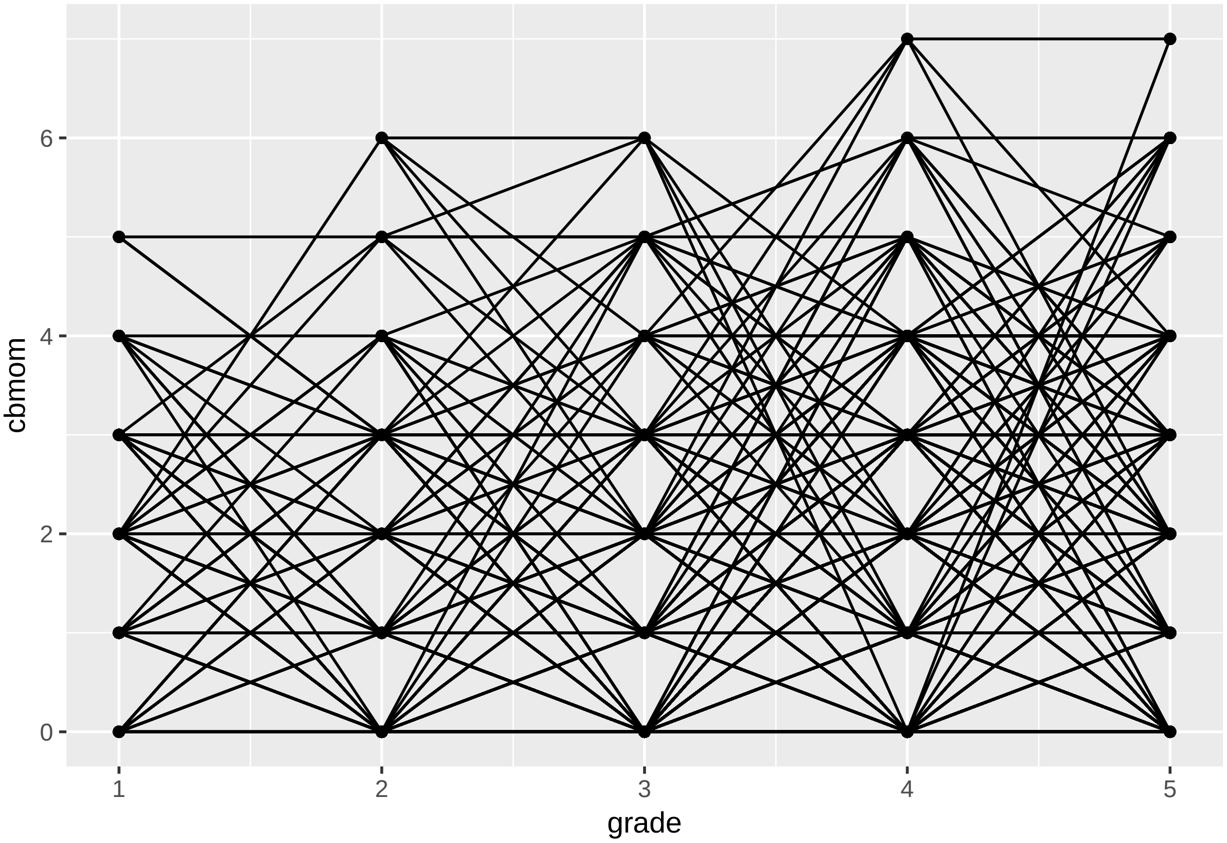
```
purpose_long_elem <- purpose_long %>%
  filter(grade != 21)
```

Plotting individual curves for conflict with mother over time:

```
gg2 <- ggplot(purpose_long_elem, aes(x = grade, y = cbmom, group = FAMID)) +
  geom_line() + geom_point()
gg2
```

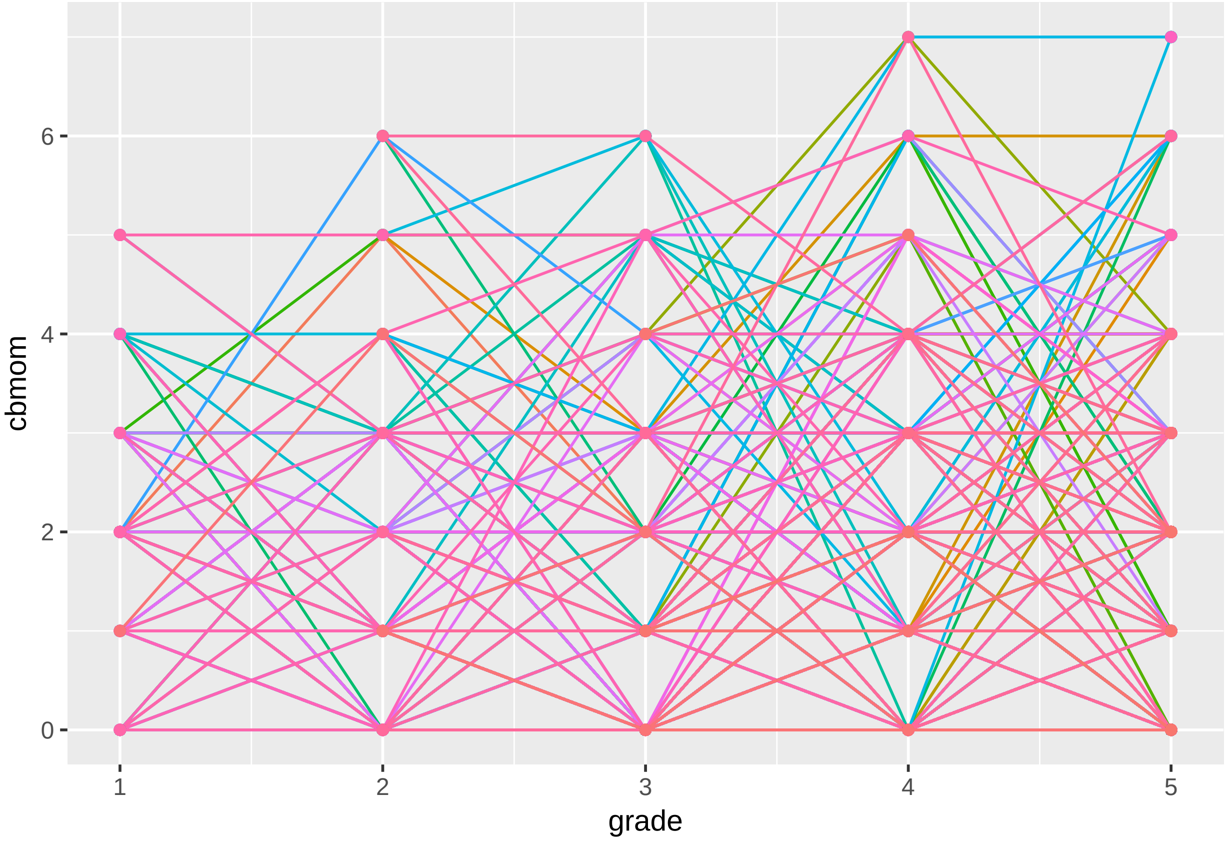
```
## Warning: Removed 2217 rows containing missing values (geom_path).
```

```
## Warning: Removed 2336 rows containing missing values (geom_point).
```



```
gg3 <- gg2 + aes(colour = factor(FAMID)) + guides(colour=FALSE)
gg3
```

```
## Warning: Removed 2217 rows containing missing values (geom_path).
## Warning: Removed 2336 rows containing missing values (geom_point).
```



Since these are sums and not averages, these curves aren't AS interesting at the moment (the predicted values look better).

```
## Subset of 10 curves
set.seed(11)
ex.random <- purpose_long_elem %>%
  select(FAMID) %>%
  distinct %>%
  sample_n(10)

example <-
  left_join(ex.random, purpose_long_elem)
```

```
## Joining, by = "FAMID"
```

```
gg4 <- ggplot(example, aes(x = grade, y = cbmom, group = FAMID)) +
  geom_point() + stat_smooth(method="lm") + facet_wrap(~FAMID)
gg4
```

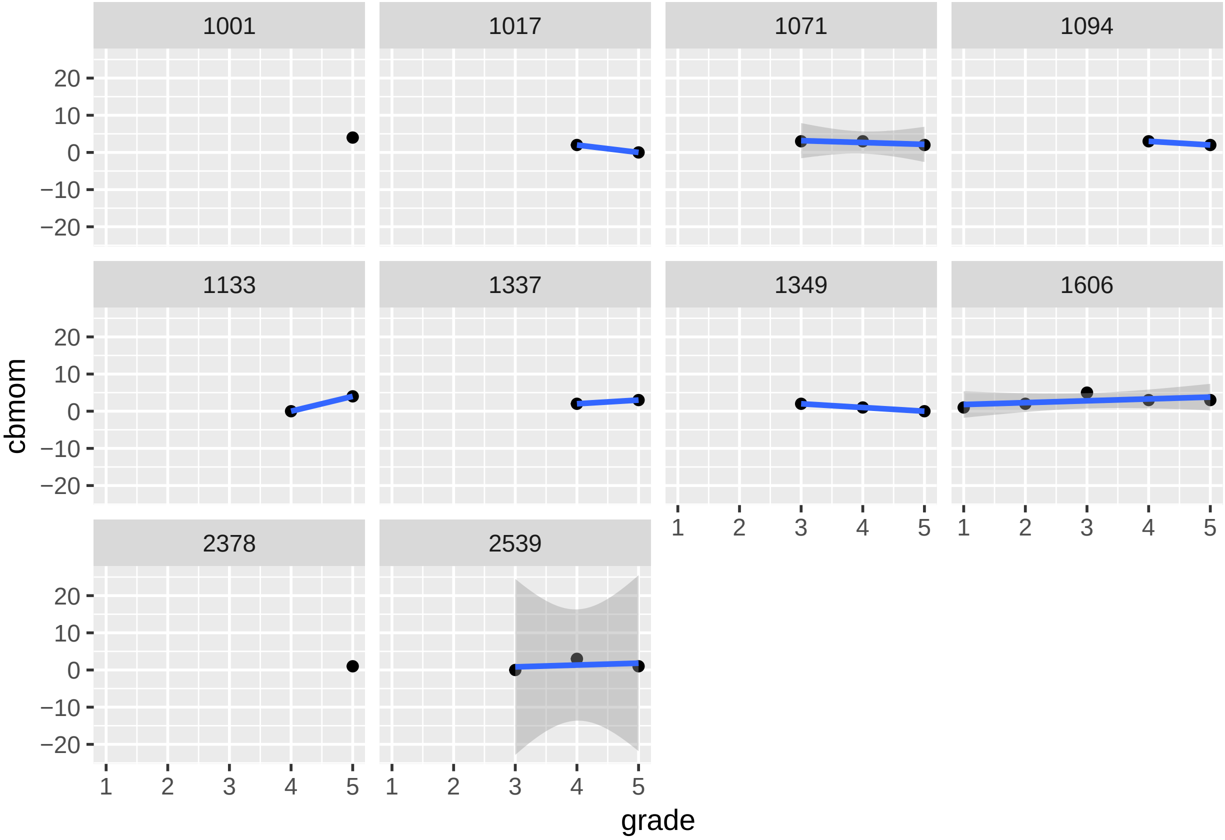
```
## Warning: Removed 26 rows containing non-finite values (stat_smooth).

## Warning in qt((1 - level)/2, df): NaNs produced

## Warning in qt((1 - level)/2, df): NaNs produced

## Warning in qt((1 - level)/2, df): NaNs produced
## Warning in qt((1 - level)/2, df): NaNs produced

## Warning: Removed 26 rows containing missing values (geom_point).
```



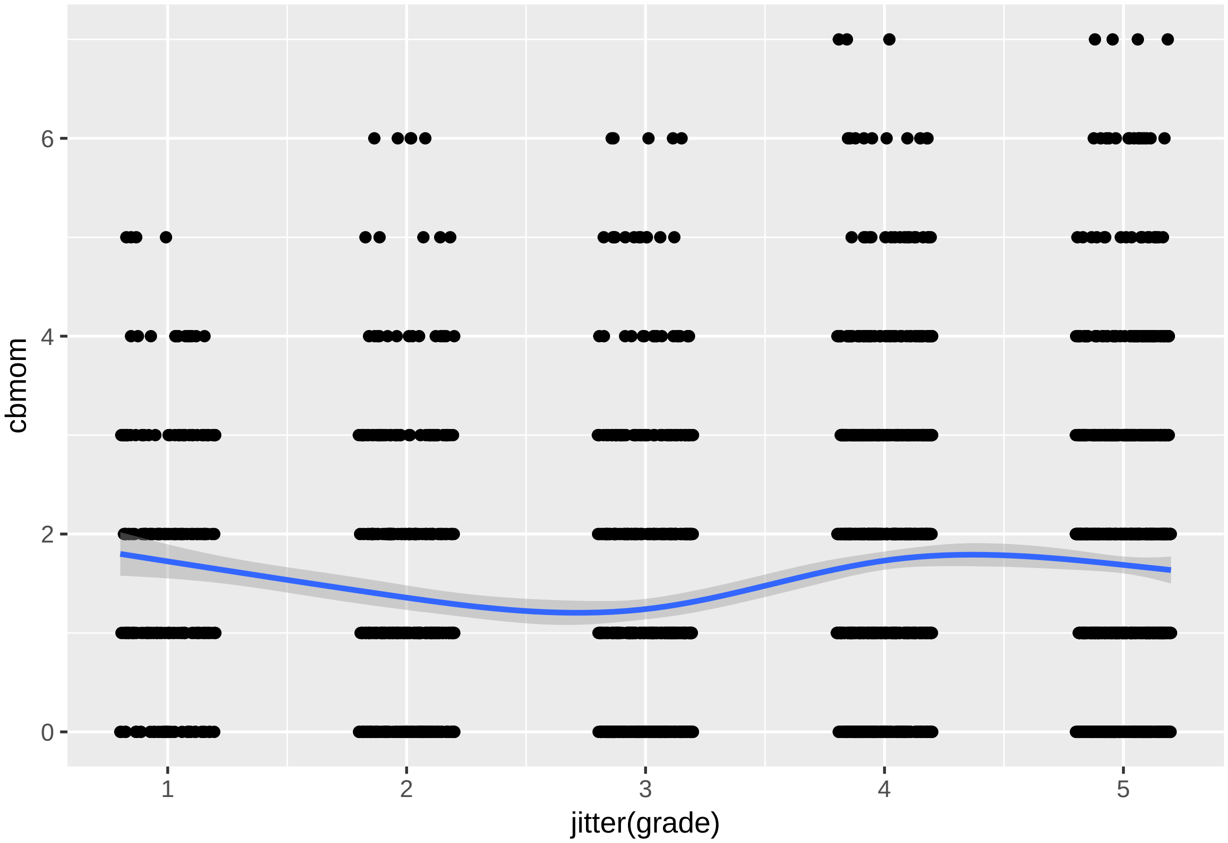
6. Create an overall average trend of your data (split up into groups if appropriate). Attempt to color your individual data points and/or shade different lines (highlight some participants, highlight the average trend line but not the individual level lines).

```
gg5 <- ggplot(purpose_long_elem, aes(x = jitter(grade), y = cbmom)) +
  geom_point() + stat_smooth()
gg5
```

```
## `geom_smooth()` using method = 'gam'

## Warning: Removed 2336 rows containing non-finite values (stat_smooth).

## Warning: Removed 2336 rows containing missing values (geom_point).
```



7. Look at the correlations of your DV across time.

```
conflict_mom <- purpose_wide %>%
  select(cbmom_1:cbmom_5)
cor(conflict_mom, use = "complete.obs")
```

```
##           cbmom_1   cbmom_2   cbmom_3   cbmom_4   cbmom_5
## cbmom_1 1.0000000 0.3466768 0.3413231 0.1729499 0.2528775
## cbmom_2 0.3466768 1.0000000 0.5052771 0.3430116 0.2370214
## cbmom_3 0.3413231 0.5052771 1.0000000 0.3243603 0.3673301
## cbmom_4 0.1729499 0.3430116 0.3243603 1.0000000 0.4884292
## cbmom_5 0.2528775 0.2370214 0.3673301 0.4884292 1.0000000
```