

# Day Three: Data Analysis

*Dillon Niederhut*

*24 May, 2016*

## Pre-introduction

While everyone is getting situated and/or cloning the course materials, pull up `feedback_cleaner.R`. As a review of day 3, walk through the different parts of the script, and ask the students to describe to you what each piece does. For example,

```
dat$timestamp <- sub(' [0-9]+:[0-9]+:[0-9]+', '', dat$timestamp)
dat$timestamp <- as.Date(dat$timestamp, "%m/%d/%Y")
```

is code that reformats ISO timestamps so that R can read them as date-type values.

## Introduction

analysis generally proceeds in two steps:

1. exploratory data analysis
2. statistical inference

our treatment of graphing owes a lot to the Grammar of Graphics

## Summarizing

let's load in some data about D-Lab feedback

```
load('data/feedback.Rda')
str(dat)
```

```
## 'data.frame': 1062 obs. of 14 variables:
## $ timestamp      : Date, format: "2015-04-23" "2015-04-23" ...
## $ course.delivered : int 7 7 7 6 7 6 3 6 5 7 ...
## $ instructor.communicated: int 6 7 5 6 7 6 2 4 4 7 ...
## $ hear           : Factor w/ 51 levels "-", "a colleague",...: 19 19 19 34 13 NA 24 19 24 31
## $ interest       : int 7 7 7 6 6 7 6 7 7 7 ...
## $ department     : Factor w/ 27 levels "African American Studies",...: NA NA NA NA NA NA NA NA
## $ verbs          : chr "This was a helpful workshop. \n\nKelly was a clear instructor and I
## $ useful         : int 7 7 7 6 6 6 3 7 4 7 ...
## $ gender         : Factor w/ 3 levels "Female/Woman",...: 2 2 NA 1 1 2 2 NA 1 1 ...
## $ ethnicity      : chr "Asian American" "White" "White" "White" ...
## $ outside.barriers : int 2 1 1 3 1 1 1 NA 1 1 ...
## $ inside.barriers  : int 1 1 1 1 1 1 1 NA 1 1 ...
## $ what.barriers    : chr NA NA NA NA ...
## $ position        : Factor w/ 23 levels "Academic staff title",...: 20 4 4 4 9 2 14 NA 15 20
```

## R provides two easy/simple summary functions in the base package

```
summary(dat)
```

```
##      timestamp      course.delivered instructor.communicated
## Min.   :2014-08-19   Min.   :1.000      Min.   :1.000
## 1st Qu.:2014-11-05   1st Qu.:6.000      1st Qu.:6.000
## Median :2015-01-30   Median :7.000      Median :7.000
## Mean   :2015-01-22   Mean   :6.251      Mean   :6.257
## 3rd Qu.:2015-04-03   3rd Qu.:7.000      3rd Qu.:7.000
## Max.   :2015-06-22   Max.   :7.000      Max.   :7.000
##
##                                     hear      interest
## Email from the D-Lab mailing list      :340   Min.   :1.0
## Found it on the D-Lab website          :278   1st Qu.:6.0
## Heard about it from a friend/colleague:247   Median :7.0
## Email from another mailing list        : 99   Mean   :6.6
## Don't remember                        : 12   3rd Qu.:7.0
## (Other)                              : 55   Max.   :7.0
## NA's                                  : 31   NA's   :15
##
##      department      verbs      useful
## Public Health      : 81   Length:1062   Min.   :1.00
## Public Policy      : 44   Class :character 1st Qu.:5.00
## Sociology          : 38   Mode  :character Median :6.00
## Political Science  : 36                      Mean   :6.02
## Integrative Biology: 28                      3rd Qu.:7.00
## (Other)            :288                      Max.   :7.00
## NA's               :547
##
##                                     gender      ethnicity
## Female/Woman          :579   Length:1062
## Male/Man              :332   Class :character
## Genderqueer/Gender non-conforming: 1   Mode  :character
## NA's                  :150
##
##
##
## outside.barriers inside.barriers what.barriers
## Min.   :1.000      Min.   :1.000      Length:1062
## 1st Qu.:1.000      1st Qu.:1.000      Class :character
## Median :1.000      Median :1.000      Mode  :character
## Mean   :2.073      Mean   :1.259
## 3rd Qu.:3.000      3rd Qu.:1.000
## Max.   :5.000      Max.   :5.000
## NA's   :167        NA's   :175
##
##                                     position
## PhD student, dissertation stage: 41
## PhD student, pre-dissertation  : 33
## Visiting fellow or researcher  : 24
## Masters student                : 22
## Undergraduate student          : 21
## (Other)                        : 64
## NA's                           :857
```

```
table(dat$department)
```

```
##
## African American Studies Ag & Resource Econ & Pol
##                24                23
##                Anthropology App Sci & Tech Grad Grp
##                12                10
## Biostatistics Grad Grp City & Regional Planning
##                8                20
##                Economics                Education
##                23                26
## Energy & Resources Group Env Sci, Policy, & Mgmt
##                14                17
## Ethnic Studies Grad Grp                History
##                1                17
## Industrial Eng & Ops Rsch                Information
##                4                9
## Integrative Biology                JSP Grad Pgm
##                28                6
##                Law                Linguistics
##                9                11
##                Music                Neuroscience
##                3                4
## Political Science                Psychology
##                36                28
## Public Health                Public Policy
##                81                44
## Rhetoric Slavic Languages & Lit
##                11                8
## Sociology
##                38
```

the psych package provides trimmed means, skew, kurtosis, and missingness

```
library(psych)
describe(dat)
```

```
## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning
## Inf
```

```
## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning
## Inf
```

```
## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning
## Inf
```

```
## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning
## Inf
```

```
## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf
```

```
## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf
```

```
## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf
```

```
## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf
```

```
##               vars      n mean   sd median trimmed  mad min  max
## timestamp*      1 1062   NaN   NA     NA      NaN   NA Inf -Inf
## course.delivered  2 1062  6.25 1.11     7    6.47 0.00  1    7
## instructor.communicated  3 1062  6.26 1.08     7    6.47 0.00  1    7
## hear*           4 1031 23.08 6.77    24   23.10 7.41  1   51
## interest        5 1047  6.60 0.80     7    6.79 0.00  1    7
## department*     6  515 15.86 8.45    18   16.29 8.90  1   27
## verbs*          7  383   NaN   NA     NA      NaN   NA Inf -Inf
## useful          8 1062  6.02 1.20     6    6.23 1.48  1    7
## gender*         9  912  1.37 0.48     1    1.33 0.00  1    3
## ethnicity*     10 1062   NaN   NA     NA      NaN   NA Inf -Inf
## outside.barriers 11  895  2.07 1.29     1    1.89 0.00  1    5
## inside.barriers  12  887  1.26 0.68     1    1.07 0.00  1    5
## what.barriers*   13  120   NaN   NA     NA      NaN   NA Inf -Inf
## position*       14  205 13.14 5.84    14   13.70 2.97  1   23
##               range  skew kurtosis    se
## timestamp*      -Inf    NA      NA    NA
## course.delivered    6 -1.92    4.20 0.03
## instructor.communicated    6 -1.92    4.35 0.03
## hear*            50  0.44    0.88 0.21
## interest          6 -2.84   11.11 0.02
## department*       26 -0.37   -1.34 0.37
## verbs*           -Inf    NA      NA    NA
## useful            6 -1.57    2.89 0.04
## gender*           2  0.58   -1.58 0.02
## ethnicity*       -Inf    NA      NA    NA
## outside.barriers    4  0.87   -0.53 0.04
## inside.barriers    4  2.93    8.62 0.02
## what.barriers*     -Inf    NA      NA    NA
## position*         22 -0.85   -0.27 0.41
```

you can use `dplyr::groupby` to generate summaries

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
##
## The following objects are masked from 'package:stats':
##
##   filter, lag
##
```

```
## The following objects are masked from 'package:base':
##
## intersect, setdiff, setequal, union
```

```
dat %>% group_by(gender) %>% summarize(n())
```

```
## Source: local data frame [4 x 2]
##
##           gender    n()
##           (fctr) (int)
## 1 Female/Woman    579
## 2 Male/Man       332
## 3 Genderqueer/Gender non-conforming    1
## 4 NA           150
```

and you can combine `dplyr` with `tidyr::spread` to generate crosstabs

side note - we are filtering out missing values of gender, because `tidyr` doesn't allow NA as a column name

```
library(tidyr)
dat %>% filter(!is.na(gender)) %>% group_by(gender, department) %>%
  summarize(n=n()) %>% spread(gender, n)
```

```
## Source: local data frame [28 x 4]
##
##           department Female/Woman Male/Man
##           (fctr)      (int)      (int)
## 1 African American Studies         8        16
## 2 Ag & Resource Econ & Pol        20         3
## 3 Anthropology                     9         3
## 4 App Sci & Tech Grad Grp          6         4
## 5 Biostatistics Grad Grp          5         3
## 6 City & Regional Planning        12         7
## 7 Economics                       16         5
## 8 Education                       20         3
## 9 Energy & Resources Group         10         3
## 10 Env Sci, Policy, & Mgmt         11         5
## .. ... ... ...
## Variables not shown: Genderqueer/Gender non-conforming (int)
```

## Plotting

every time you use `base::plot`, [Edward Tufte does something unkind to a cute animal](#)

- we'll be using `ggplot`, R's implementation of the **grammar of graphics**
- in this grammar, you use 'aesthetics' to define how data is mapped to objects the graph space
- each graph space has at least three layers:

- theme/background/annotations
  - axes
  - objects
- most objects are geometric shapes
  - some objects are statistics built on those shapes
  - you can stack as many layers as you like

```
install.packages('ggplot2')
```

```
##
## The downloaded binary packages are in
## /var/folders/rj/8gpcssqd52z9yrqw7f8xxfym0000gn/T//RtmpmP1txl/downloaded_packages
```

```
library(ggplot2)
```

```
##
## Attaching package: 'ggplot2'
##
## The following objects are masked from 'package:psych':
##
##      %+%, alpha
```

## getting weekdays

let's imagine that we are interested in looking at differences in feedback based on the day of the week – how would we do this in R?

side note - `weekdays` is locale aware, so students who have their laptop language set to something other than english will get their weekday names in the other language

```
dat$wday <- factor(weekdays(dat$timestamp, abbreviate = TRUE),
                  levels = c('Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat', 'Sun')
                  )
summary(dat$wday)
```

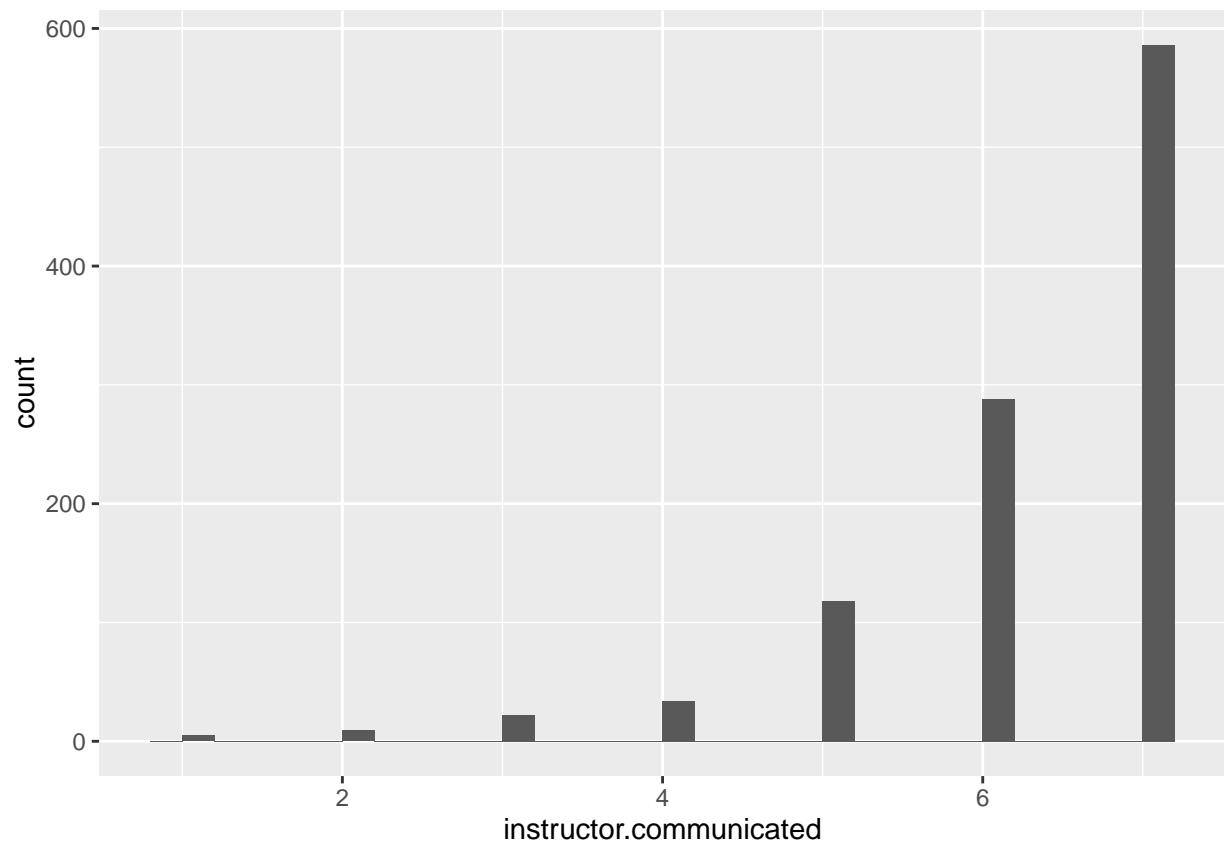
```
## Mon Tue Wed Thu Fri Sat Sun
## 168 124 144 323 277 16 10
```

## use qplot for initial poking around

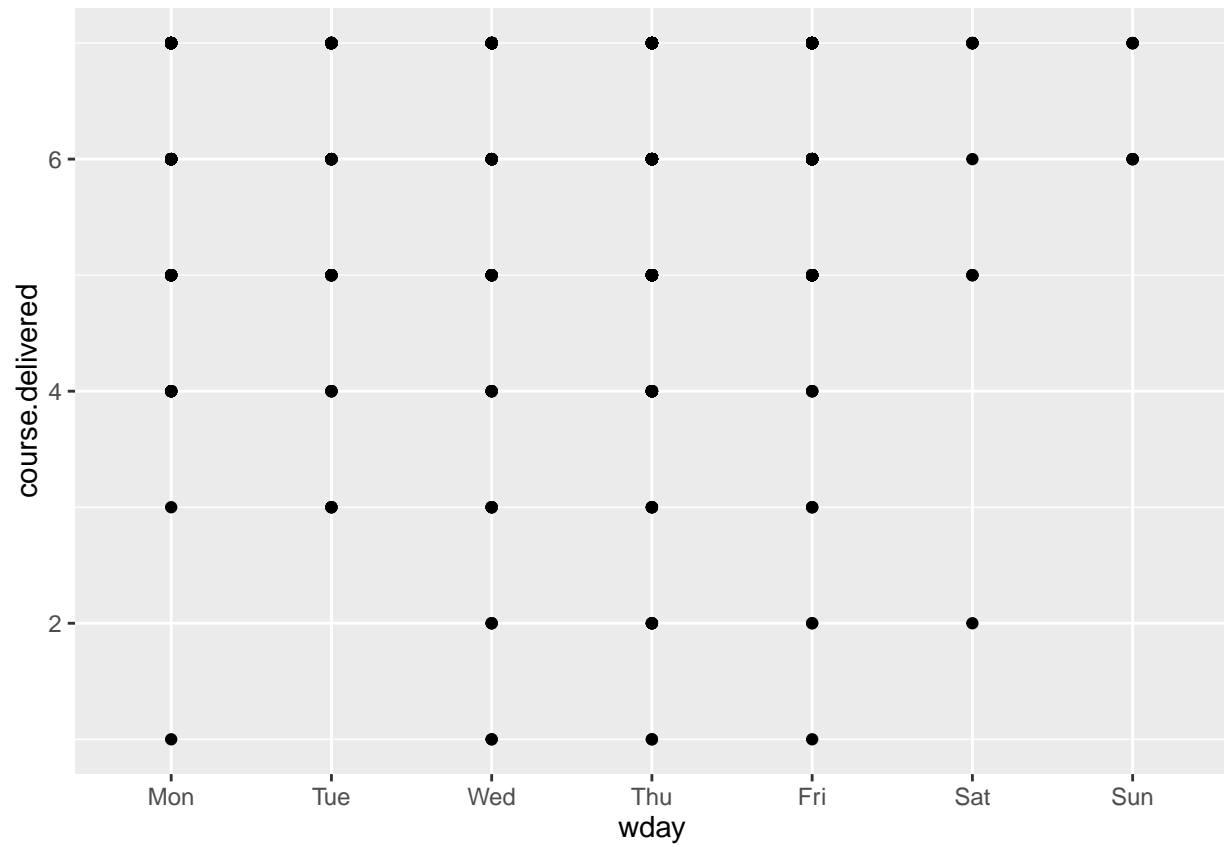
it has very strong intuitions about what you want to see, and is not particularly customizable

```
qplot(instructor.communicated, data = dat)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



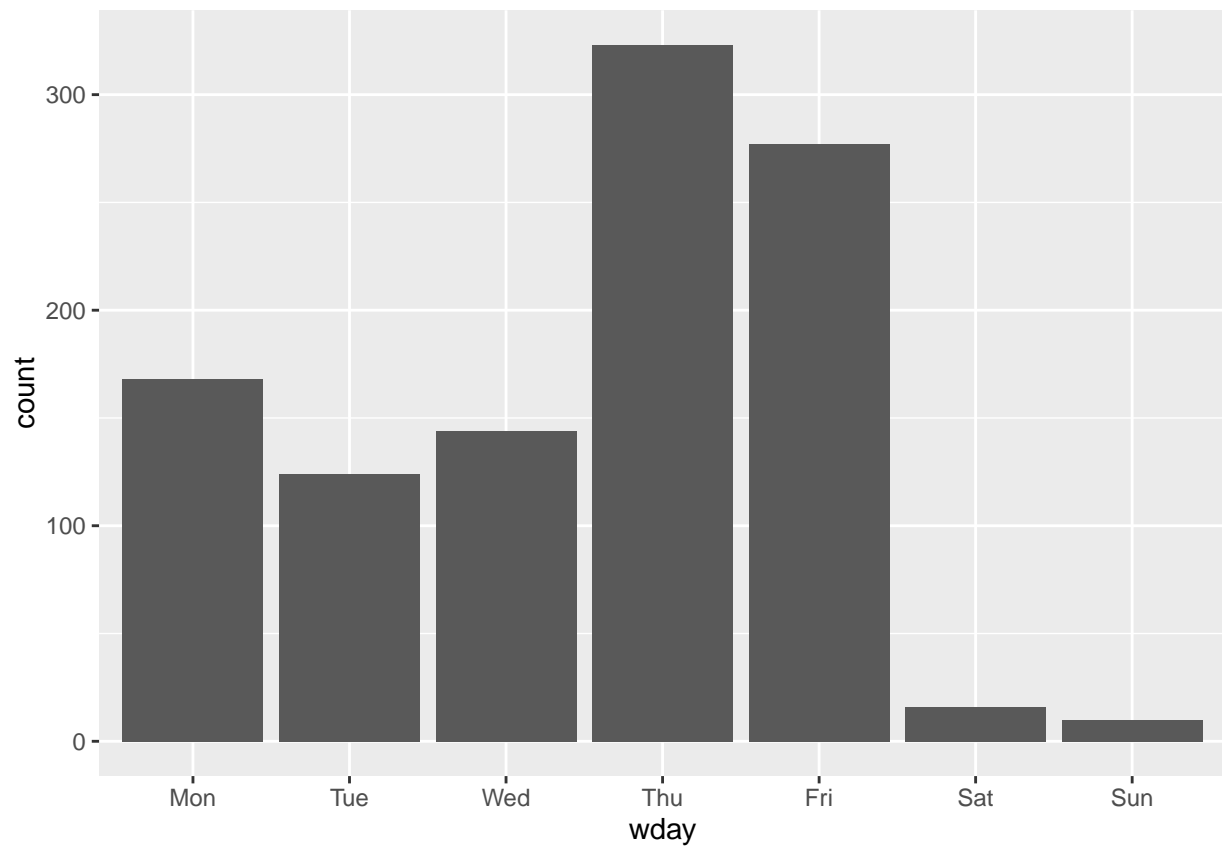
```
qplot(wday, course.delivered, data = dat)
```



for 1D categorical, use bar

```
ggplot(data=dat, aes(x=wday)) + geom_bar()
```

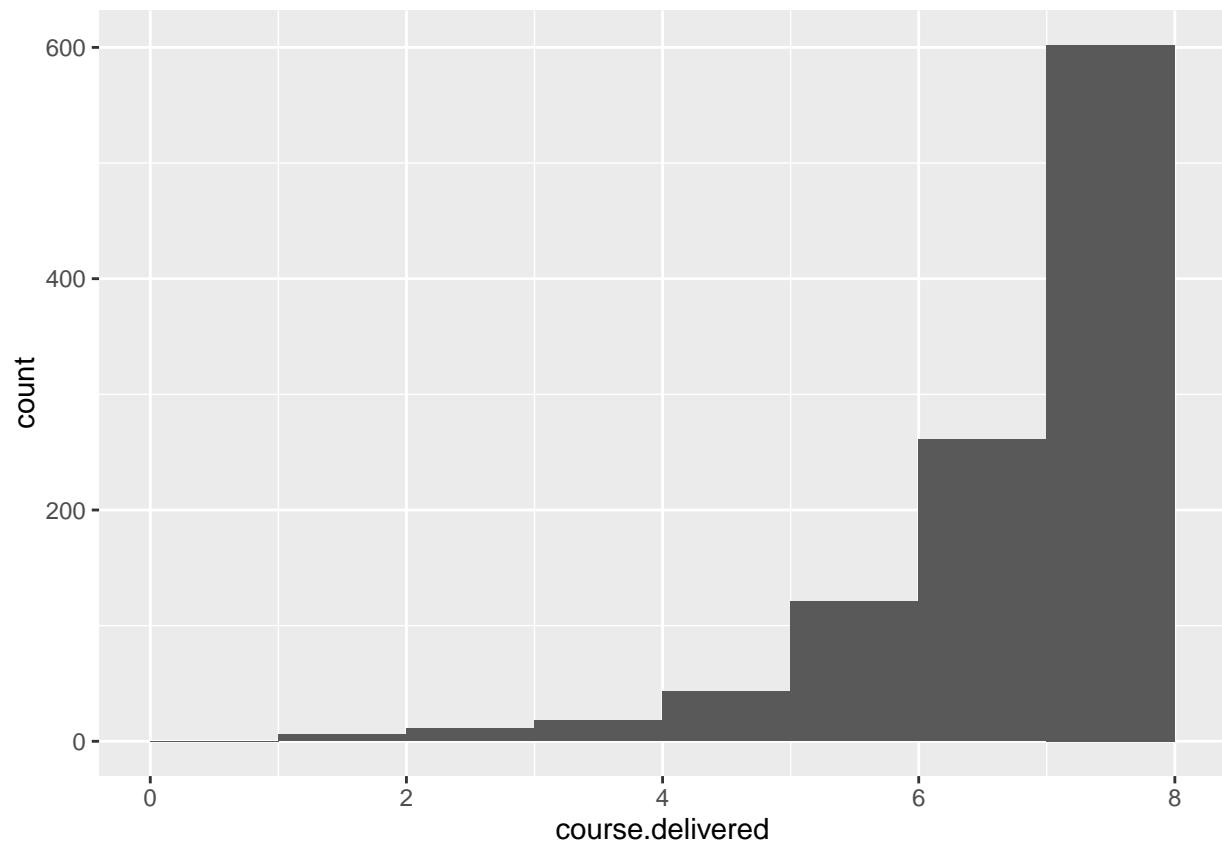




for 1D continuous, use `hist`

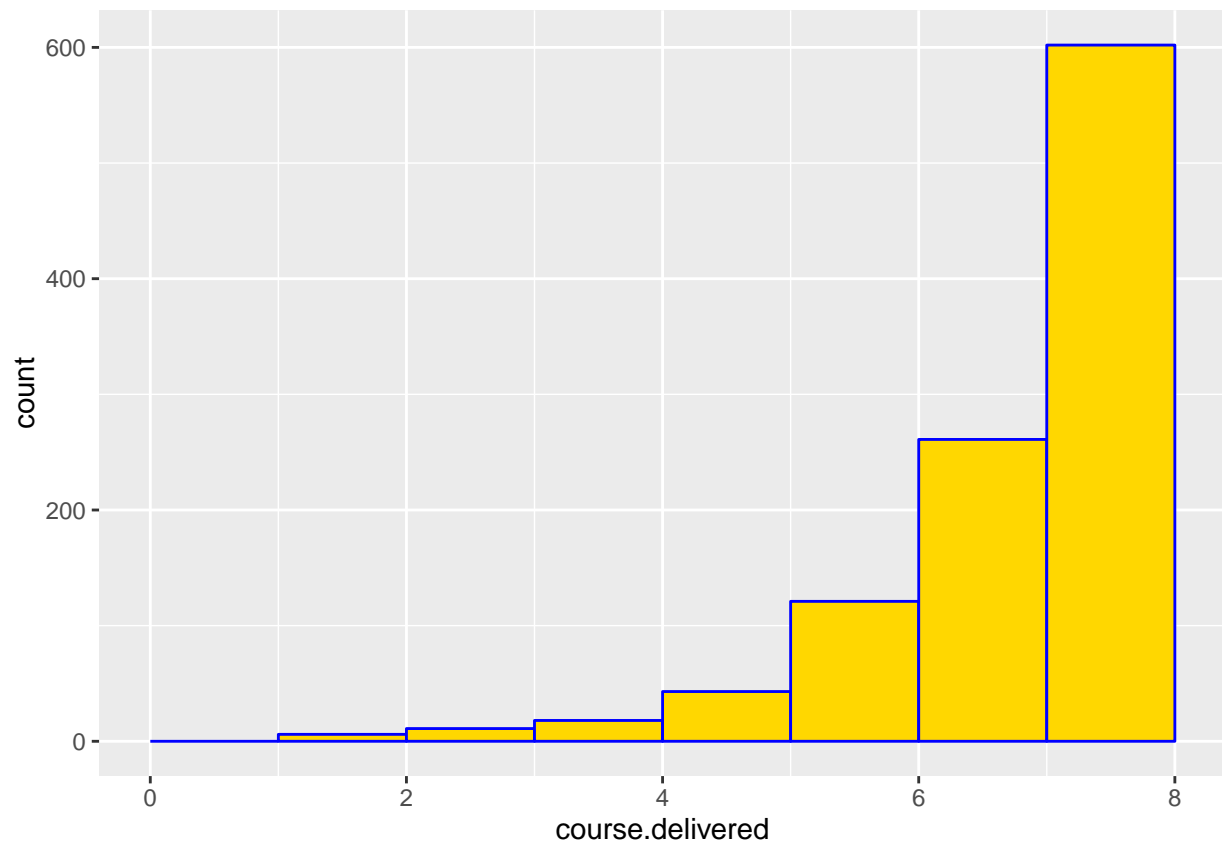
this is really just convenience for `geom_bar(stat = 'bin')`, as opposed to bar plots, whose `stat` is `'count'`

```
ggplot(data=dat, aes(x=course.delivered)) +  
  geom_histogram(binwidth=1)
```



you can add color to this plot

```
ggplot(data=dat, aes(x=course.delivered)) +  
  geom_histogram(binwidth=1, fill = 'gold', colour= 'blue')
```



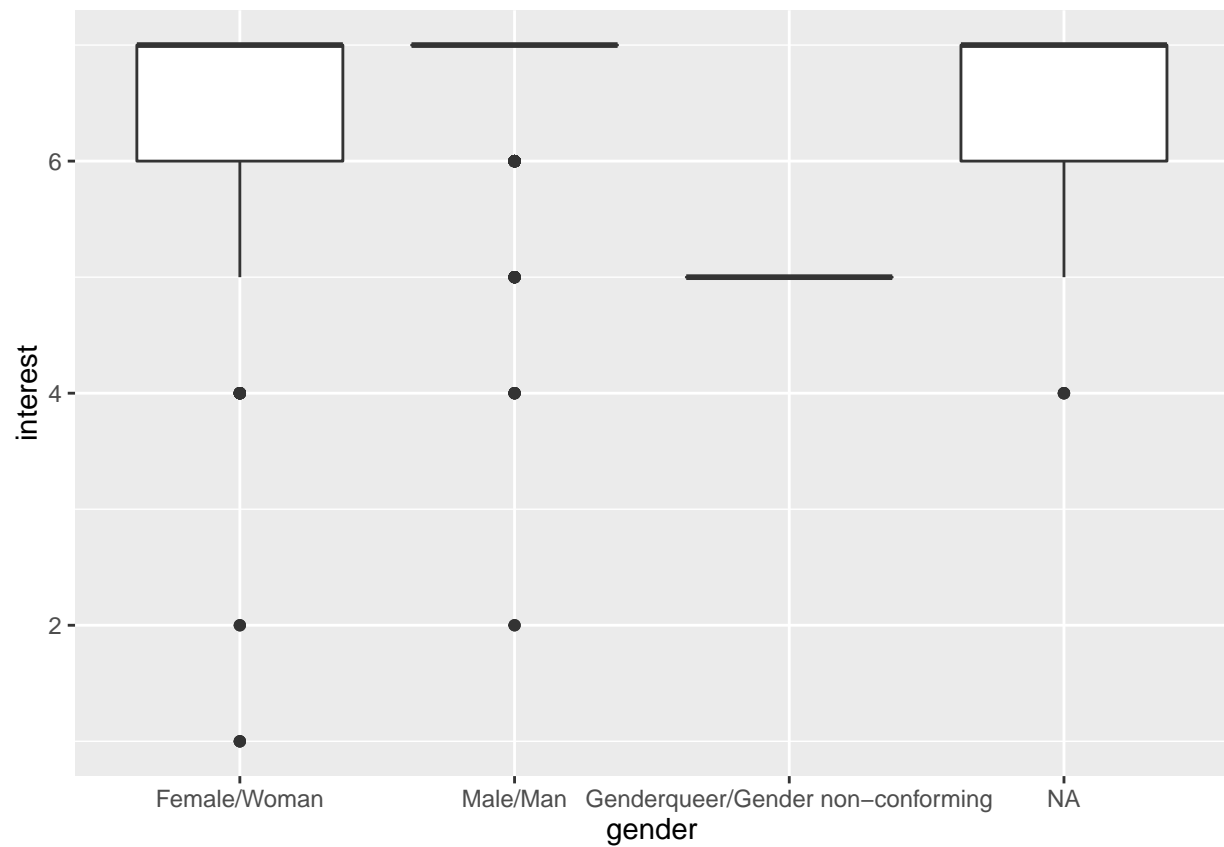
GO BEARS

**for many 1D variables, use a box plot**

these are handy for a whole bunch of reasons, and you should make them your close associates

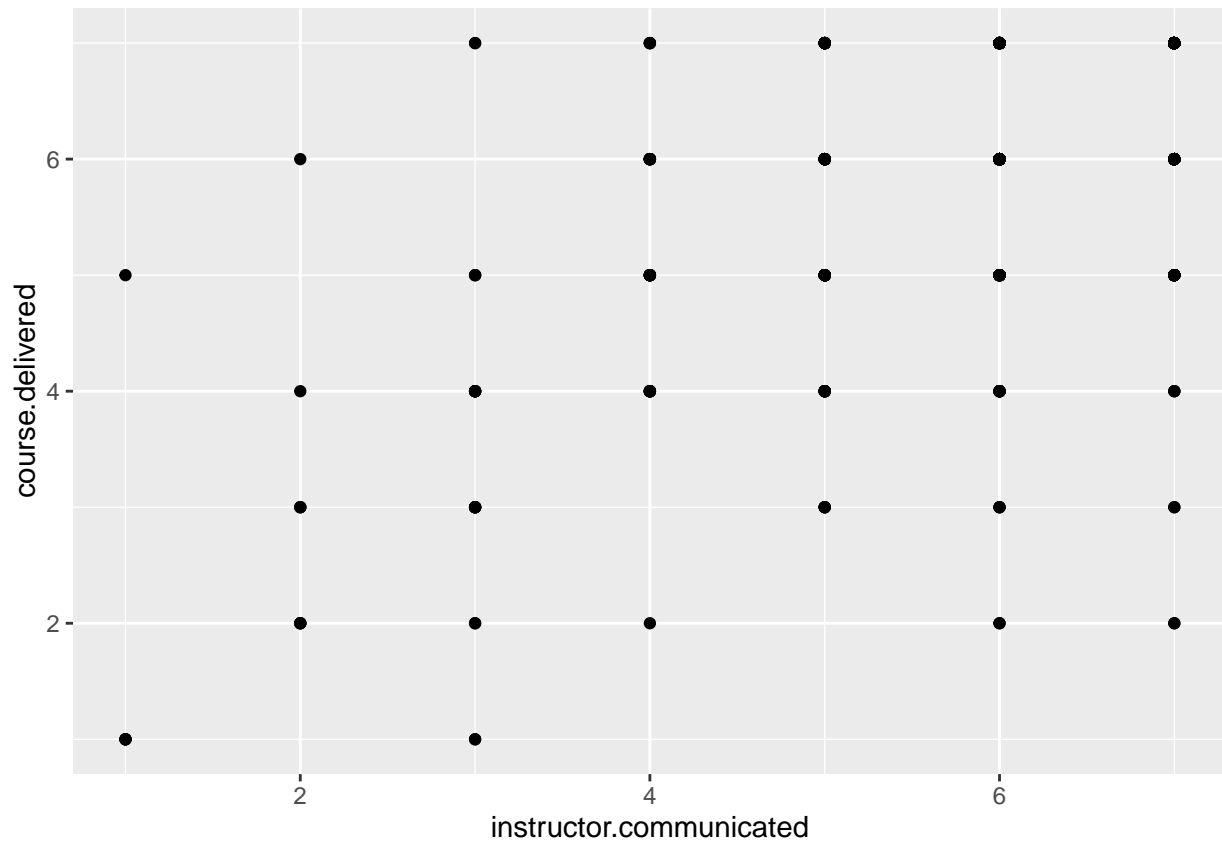
```
ggplot(data=dat, aes(x=gender,y=interest)) + geom_boxplot()
```

```
## Warning: Removed 15 rows containing non-finite values (stat_boxplot).
```



to plot two continuous variables, use points

```
ggplot(data=dat, aes(x=instructor.communicated, y=course.delivered)) + geom_point()
```

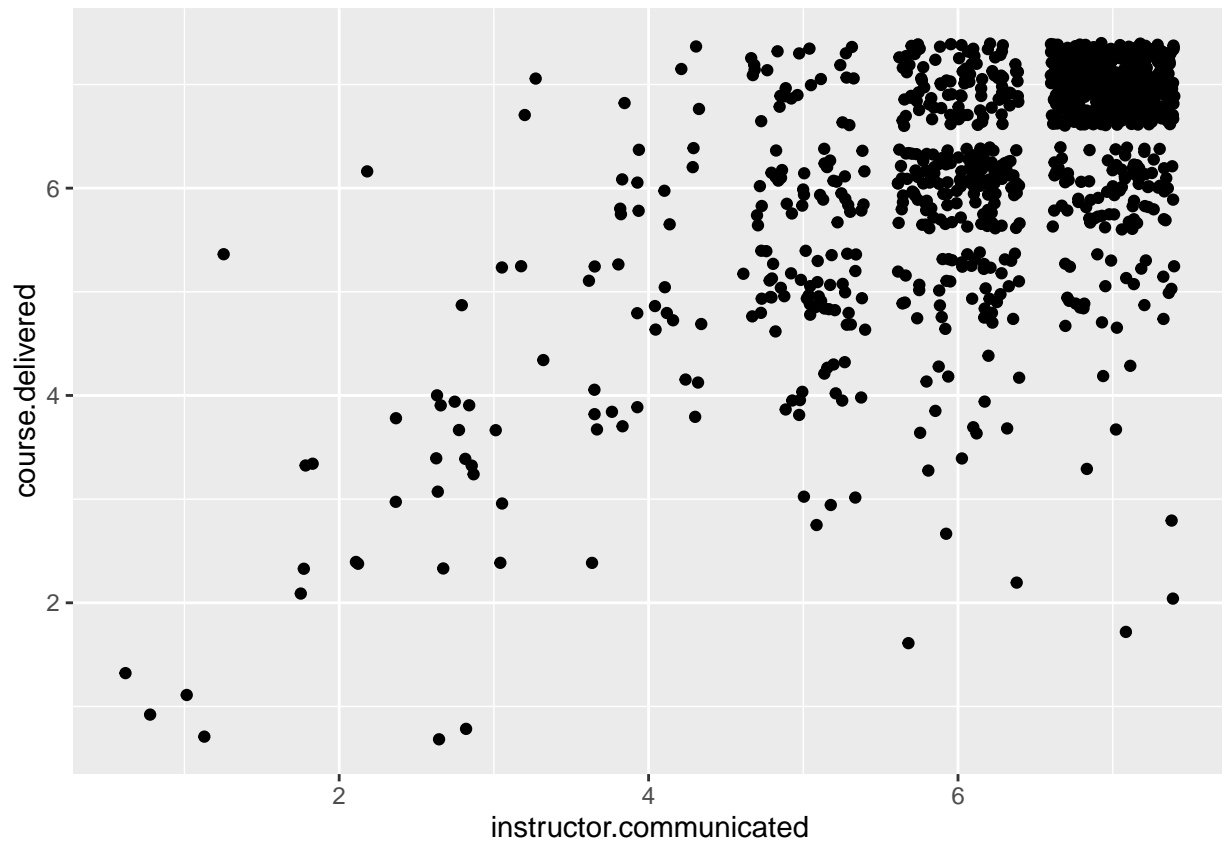


all of these values are discrete, which makes them hard to see

**to scatter points randomly, use jitter**

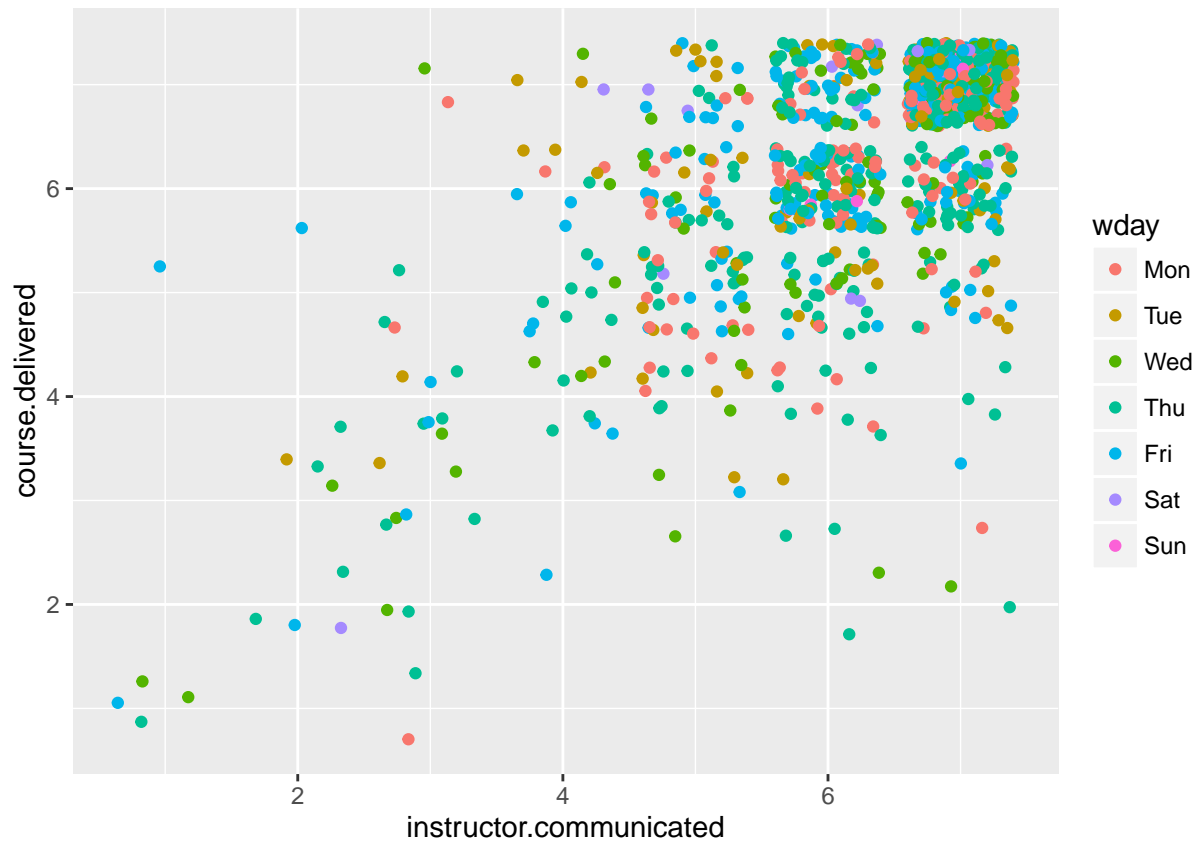
this is really just convenience for `geom_point(position = jitter())`

```
ggplot(data=dat, aes(x=instructor.communicated, y=course.delivered)) +  
  geom_jitter()
```



not only can you add color, you can make the color a mapping of other variables

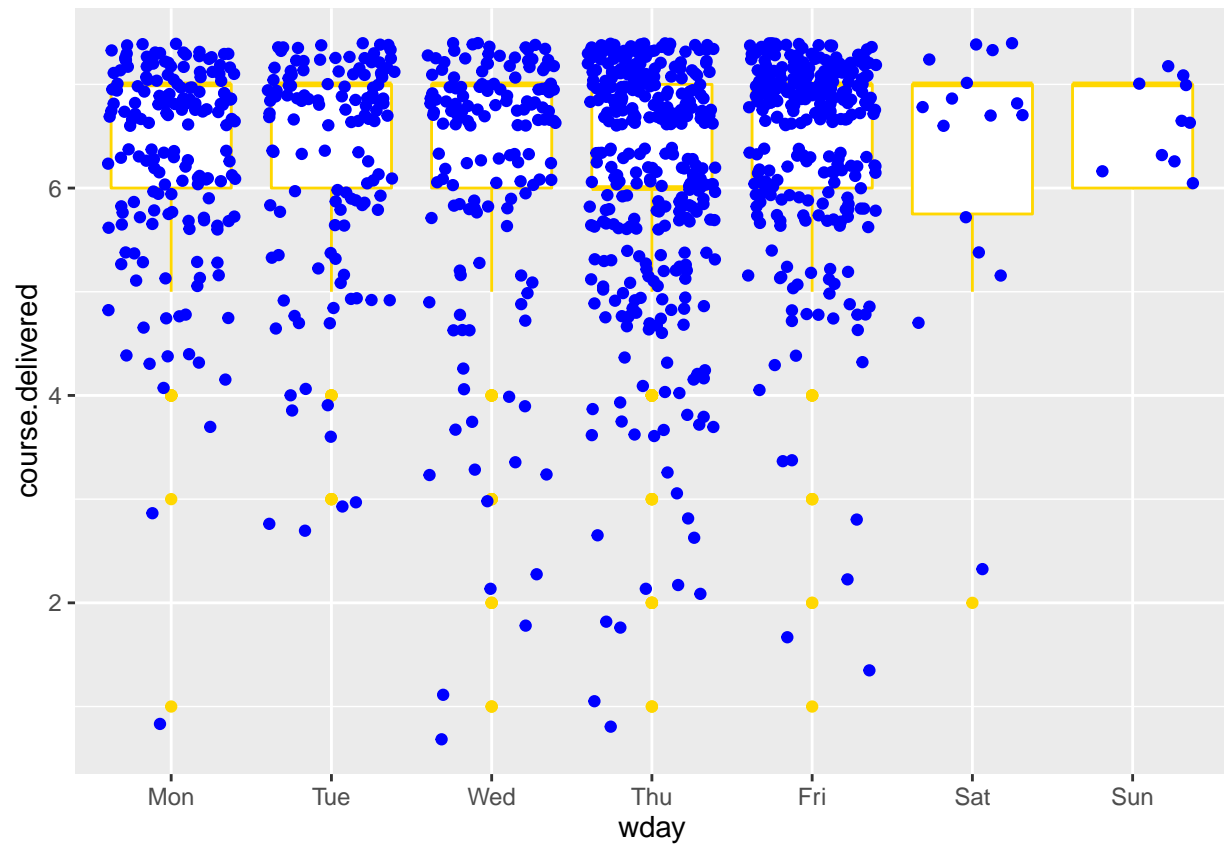
```
ggplot(data=dat, aes(x=instructor.communicated, y=course.delivered)) +  
  geom_jitter(aes(colour = wday))
```



the last time we used colour it was not an aesthetic - why is it now?

you can stack layers until your eyes hurt

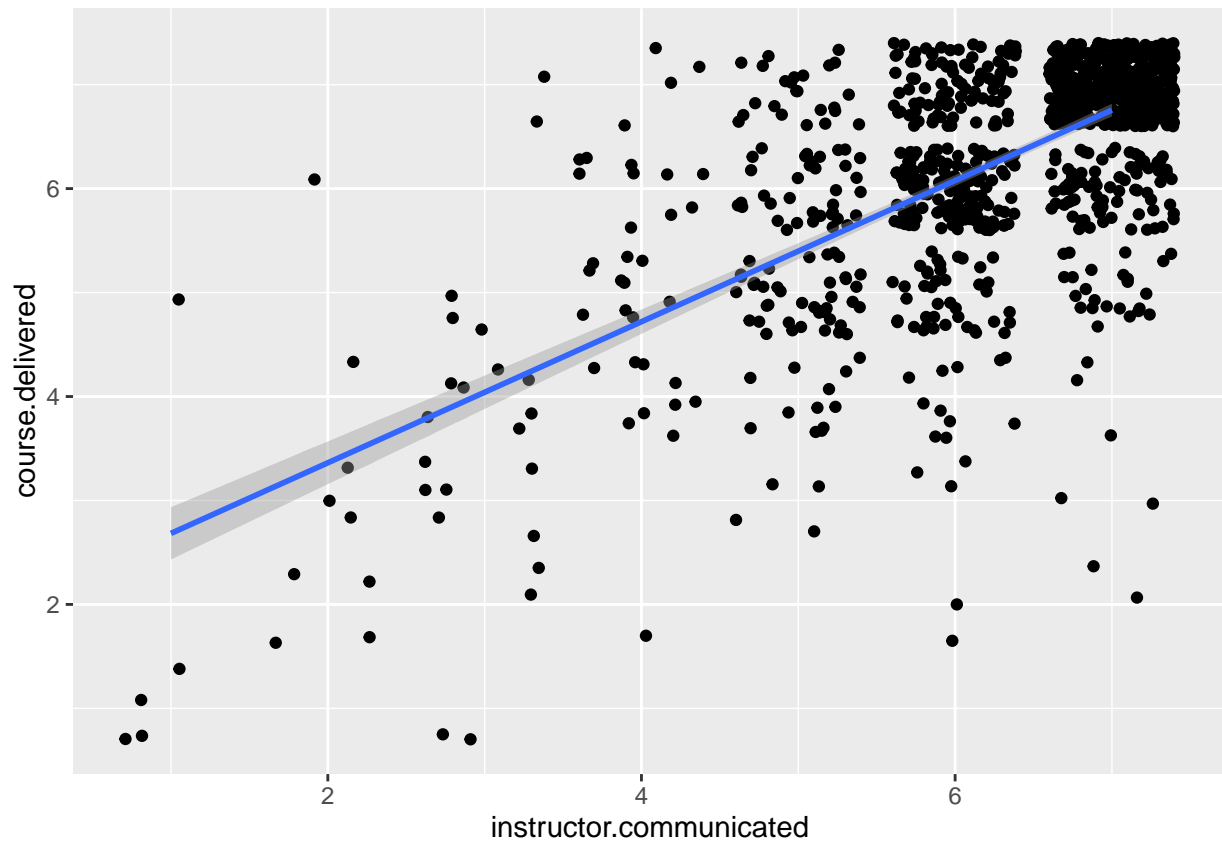
```
ggplot(data=dat, aes(x=wday, y=course.delivered)) +
  geom_boxplot(colour = 'gold') +
  geom_jitter(colour = 'blue')
```



add summary functions with smooth

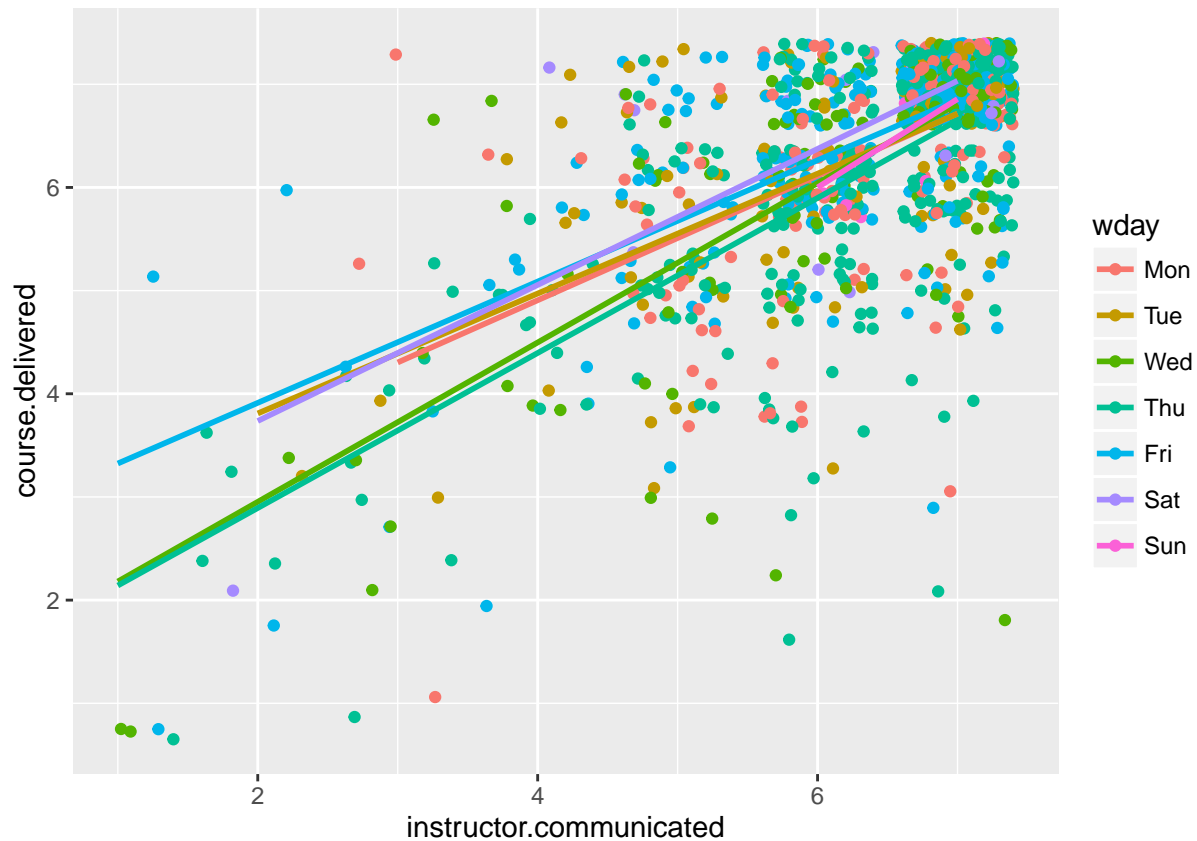
```
ggplot(data=dat, aes(x=instructor.communicated, y=course.delivered)) +  
  geom_jitter() +  
  stat_smooth(method = 'lm')
```





if you are using colour as an aesthetic, you'll produce stats for each color

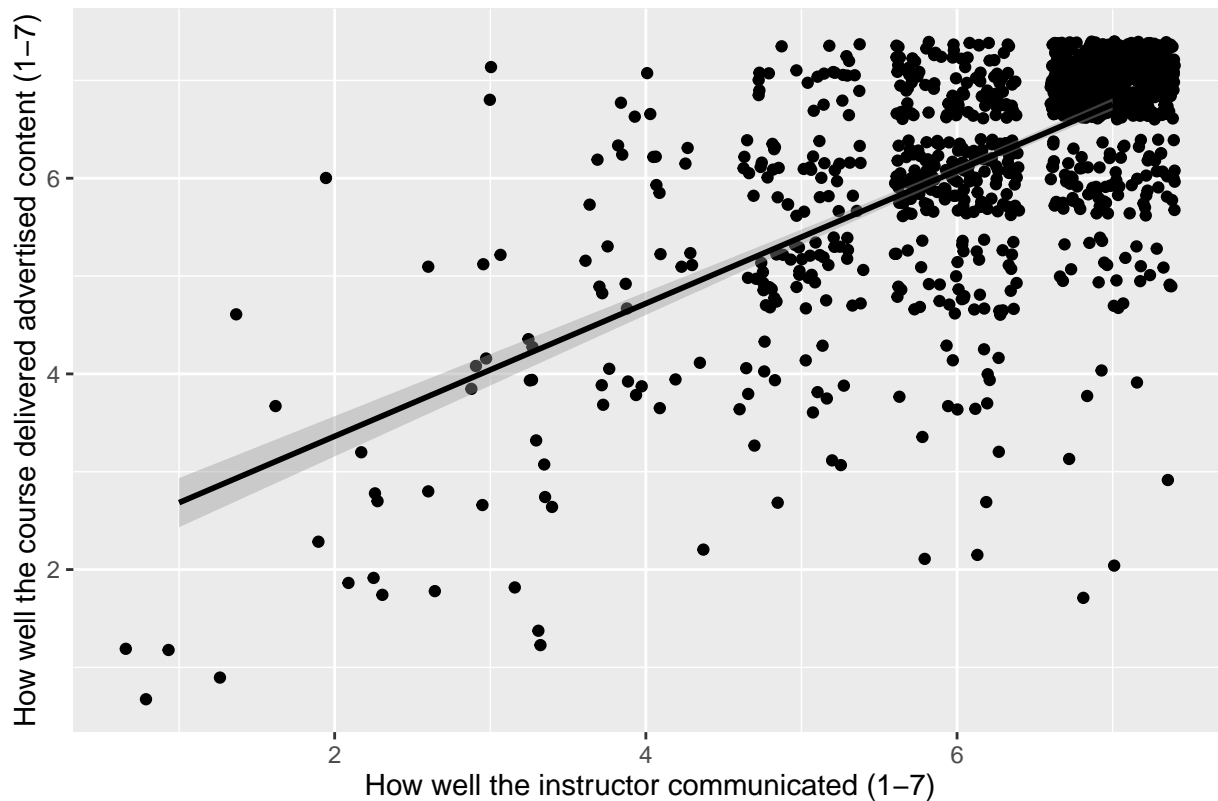
```
ggplot(data=dat, aes(x=instructor.communicated, y=course.delivered, colour = wday)) +  
  geom_jitter() +  
  stat_smooth(method = 'lm', se = FALSE)
```



good scientists put units on their axes

```
ggplot(data=dat, aes(x=instructor.communicated, y=course.delivered)) +
  geom_jitter() +
  stat_smooth(method = 'lm', colour = 'black') +
  xlab('How well the instructor communicated (1-7)') +
  ylab('How well the course delivered advertised content (1-7)') +
  ggtitle("I have no idea what I'm doing")
```

I have no idea what I'm doing



the general point here is that every single object on this graph is customizable

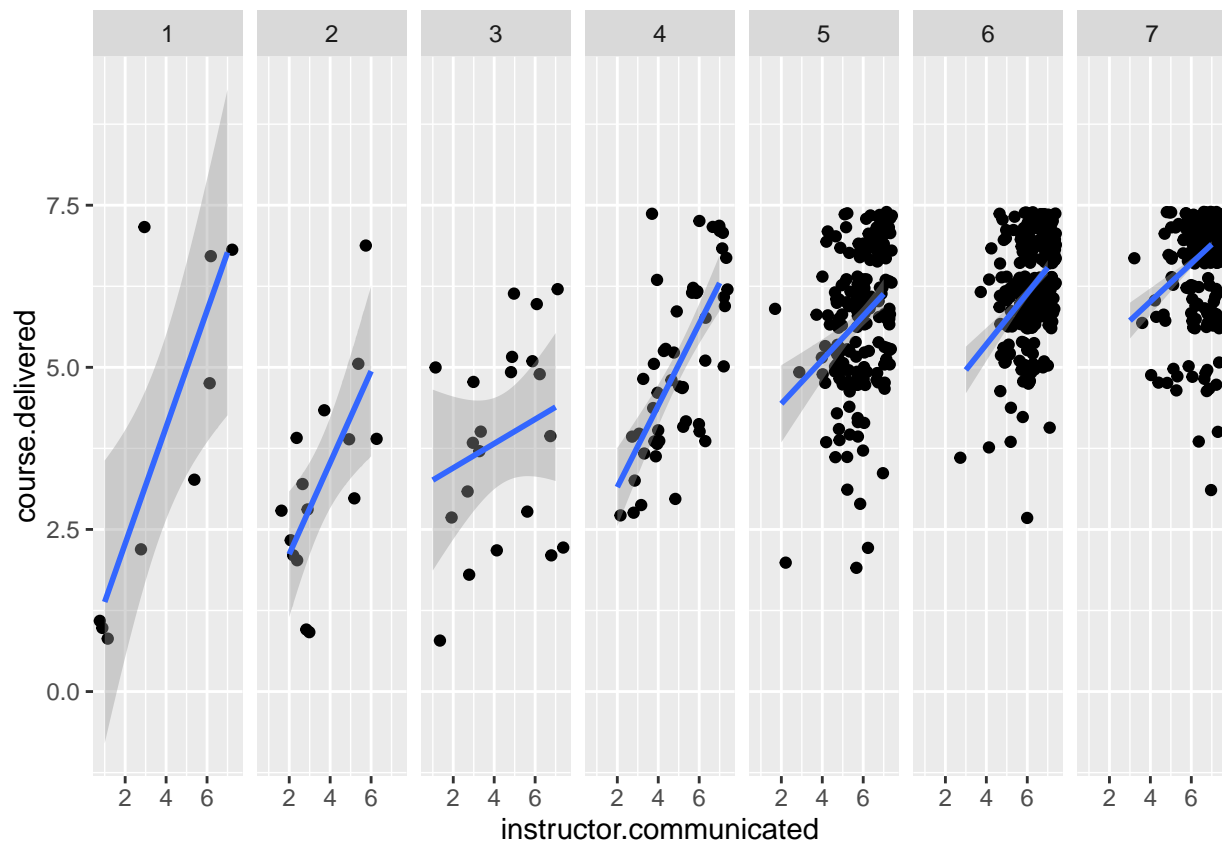
frequent customizations are very simple to add

infrequent customizations will take a lot of tinkering on your part

## facetting

often useful for looking at relationships between three variables at the same time

```
ggplot(data=dat, aes(x=instructor.communicated, y=course.delivered)) +  
  geom_jitter() +  
  stat_smooth(method = 'lm') +  
  facet_grid(. ~ useful)
```



## Mean testing

a picture is worth 1,000 words, but a p-value is worth a dissertation

basically, inferential statistics is the application of probability theory to decide what is real and what isn't

we'll start by trying to tell whether differences between group summaries are real

### t.test with two vectors (default method)

```
t.test(dat$inside.barriers, dat$outside.barriers)
```

```
##
##  Welch Two Sample t-test
##
## data:  dat$inside.barriers and dat$outside.barriers
## t = -16.638, df = 1356.8, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.9092224 -0.7174269
## sample estimates:
## mean of x mean of y
##  1.259301  2.072626
```

note that R takes care of the defaults for you - what it is really computing is `t.test(dat$inside.barriers, dat$outside.barriers, alternative = "two.sided", paired = FALSE, var.equal = FALSE, mu = 0, conf.level = 0.95)`

how would you find this out for yourself?

## t.test with subsets of one vector (default method)

```
t.test(dat$outside.barriers[dat$gender == "Male/Man"], dat$outside.barriers[dat$gender == "Female/Woman"])

##
## Welch Two Sample t-test
##
## data: dat$outside.barriers[dat$gender == "Male/Man"] and dat$outside.barriers[dat$gender == "Female/Woman"]
## t = -6.9925, df = 748.19, p-value = 5.993e-12
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.7650033 -0.4296142
## sample estimates:
## mean of x mean of y
## 1.702875 2.300184
```

recall that we mentioned inconsistency on day one - here it is, and in a big way

## t.test with S3 method

```
t.test(outside.barriers ~ gender, data = dat, subset = dat$gender %in% c("Male/Man", "Female/Woman"))

##
## Welch Two Sample t-test
##
## data: outside.barriers by gender
## t = 6.9925, df = 748.19, p-value = 5.993e-12
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## 0.4296142 0.7650033
## sample estimates:
## mean in group Female/Woman mean in group Male/Man
## 2.300184 1.702875
```

## aov

first, you would think anova would be called by `anova`, but that's reserved for conducting F-tests on `lm` objects

second, you really shouldn't be using `anova`, but if you must do it in R, the syntax looks like this

side note - ANOVA was invented by Ron Fisher to make it easy to do linear models with only a pencil and paper, and has been superseded by regression since the advent of computation in the 70s

```
aov(outside.barriers ~ gender, data = dat)
```

```
## Call:
## aov(formula = outside.barriers ~ gender, data = dat)
##
## Terms:
##              gender Residuals
## Sum of Squares    79.3444 1363.4374
## Deg. of Freedom      2      854
##
## Residual standard error: 1.263539
## Estimated effects may be unbalanced
## 205 observations deleted due to missingness
```

this isn't particularly helpful, but remember that it is an object, and we can call other, more helpful functions, on that object

remember our old friend `summary`? it works on almost everything

```
model.1 <- aov(outside.barriers ~ gender, data = dat)
summary(model.1)
```

```
##              Df Sum Sq Mean Sq F value    Pr(>F)
## gender          2   79.3   39.67    24.85 3.24e-11 ***
## Residuals      854 1363.4    1.60
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 205 observations deleted due to missingness
```

that's a little better - but what about post-hoc testing?

```
TukeyHSD(model.1)
```

```
## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = outside.barriers ~ gender, data = dat)
##
## $gender
##              diff            lwr
## Male/Man-Female/Woman -0.5973088 -0.8078392
## Genderqueer/Gender non-conforming-Female/Woman 2.6998158 -0.2694533
## Genderqueer/Gender non-conforming-Male/Man 3.2971246 0.3258507
##              upr            p adj
## Male/Man-Female/Woman -0.3867784 0.000000
## Genderqueer/Gender non-conforming-Female/Woman 5.6690850 0.083531
## Genderqueer/Gender non-conforming-Male/Man 6.2683985 0.025285
```

side note - apparently Stata stores all of the models that you generate, whether you assign them names or not; in R, you must explicitly give your models names or they will disappear into the ether

## linear models

mean tests are really just a subset of linear models where your predictor is a category

### cor.test (Pearson)

earlier, we were looking at differences between the means of two variables

but those variables were both continuous, so we can ask whether they are related

```
cor.test(dat$outside.barriers, dat$inside.barriers)

##
## Pearson's product-moment correlation
##
## data: dat$outside.barriers and dat$inside.barriers
## t = 15.558, df = 882, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.4106679 0.5142422
## sample estimates:
##          cor
## 0.4640396
```

okay, so they're related - now what?

### lm

this is probably the closest you will get to building a linear model by hand

this means lm is a powerful tool, but you have to know what you're doing

the basic call is the S3 method

```
model.1 <- lm(inside.barriers ~ outside.barriers, data = dat)
summary(model.1)

##
## Call:
## lm(formula = inside.barriers ~ outside.barriers, data = dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.98483 -0.24569  0.00069  0.00069  3.01517
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.75292    0.03842   19.60  <2e-16 ***
## outside.barriers 0.24638    0.01584   15.56  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.6041 on 882 degrees of freedom
## (178 observations deleted due to missingness)
## Multiple R-squared: 0.2153, Adjusted R-squared: 0.2144
## F-statistic: 242 on 1 and 882 DF, p-value: < 2.2e-16
```

## R automatically one-hot encodes your categories

```
model.2 <- lm(inside.barriers ~ outside.barriers + department, data = dat)
summary(model.2)
```

```
##
## Call:
## lm(formula = inside.barriers ~ outside.barriers + department,
##     data = dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.20049 -0.36011 -0.04989  0.17705  2.91702
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.91782    0.14467   6.344 5.57e-10
## outside.barriers    0.27713    0.02492  11.122 < 2e-16
## departmentAg & Resource Econ & Pol -0.50167    0.19758  -2.539  0.0115
## departmentAnthropology -0.05175    0.25719  -0.201  0.8406
## departmentApp Sci & Tech Grad Grp  0.11828    0.26693   0.443  0.6579
## departmentBiostatistics Grad Grp -0.06243    0.26679  -0.234  0.8151
## departmentCity & Regional Planning -0.20133    0.20909  -0.963  0.3361
## departmentEconomics -0.33051    0.19965  -1.655  0.0986
## departmentEducation -0.10298    0.19602  -0.525  0.5996
## departmentEnergy & Resources Group -0.44436    0.24646  -1.803  0.0721
## departmentEnv Sci, Policy, & Mgmt -0.04236    0.21656  -0.196  0.8450
## departmentEthnic Studies Grad Grp -0.47207    0.66073  -0.714  0.4753
## departmentHistory      0.16488    0.21638   0.762  0.4465
## departmentIndustrial Eng & Ops Rsch -0.22207    0.35128  -0.632  0.5276
## departmentInformation -0.21906    0.25570  -0.857  0.3921
## departmentIntegrative Biology -0.32510    0.18972  -1.714  0.0873
## departmentJSP Grad Pgm   0.09721    0.35124   0.277  0.7821
## departmentLaw -0.37970    0.25570  -1.485  0.1383
## departmentLinguistics -0.28064    0.25582  -1.097  0.2732
## departmentMusic -0.47207    0.47727  -0.989  0.3231
## departmentNeuroscience -0.26423    0.35148  -0.752  0.4526
## departmentPolitical Science -0.14505    0.17595  -0.824  0.4102
## departmentPsychology -0.11197    0.18571  -0.603  0.5469
## departmentPublic Health -0.37200    0.15691  -2.371  0.0182
## departmentPublic Policy -0.16255    0.17016  -0.955  0.3399
## departmentRhetoric      0.17521    0.24153   0.725  0.4686
## departmentSlavic Languages & Lit -0.19495    0.26748  -0.729  0.4665
## departmentSociology -0.34162    0.17664  -1.934  0.0537
##
## (Intercept) ***
## outside.barriers ***
```



```
## departmentAg & Resource Econ & Pol *
## departmentAnthropology
## departmentApp Sci & Tech Grad Grp
## departmentBiostatistics Grad Grp
## departmentCity & Regional Planning
## departmentEconomics .
## departmentEducation
## departmentEnergy & Resources Group .
## departmentEnv Sci, Policy, & Mgmt
## departmentEthnic Studies Grad Grp
## departmentHistory
## departmentIndustrial Eng & Ops Rsch
## departmentInformation
## departmentIntegrative Biology .
## departmentJSP Grad Pgm
## departmentLaw
## departmentLinguistics
## departmentMusic
## departmentNeuroscience
## departmentPolitical Science
## departmentPsychology
## departmentPublic Health *
## departmentPublic Policy
## departmentRhetoric
## departmentSlavic Languages & Lit
## departmentSociology .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6462 on 440 degrees of freedom
## (594 observations deleted due to missingness)
## Multiple R-squared:  0.2759, Adjusted R-squared:  0.2314
## F-statistic: 6.209 on 27 and 440 DF, p-value: < 2.2e-16
```

R does not assume you want the full factorial model

```
model.3 <- lm(inside.barriers ~ outside.barriers + department + outside.barriers*department, data = dat)
summary(model.3)
```

```
##
## Call:
## lm(formula = inside.barriers ~ outside.barriers + department +
##     outside.barriers * department, data = dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.75495 -0.25924  0.00000  0.05784  2.80608
##
## Coefficients: (3 not defined because of singularities)
##                                Estimate Std. Error
## (Intercept)                   0.3378995  0.2274560
## outside.barriers               0.6042618  0.1072238
```

## departmentAg & Resource Econ & Pol	0.5964070	0.3649460
## departmentAnthropology	0.1087024	0.4637595
## departmentApp Sci & Tech Grad Grp	0.0001286	0.5189858
## departmentBiostatistics Grad Grp	-0.7015359	0.5198322
## departmentCity & Regional Planning	0.4121005	0.4931678
## departmentEconomics	0.7321813	0.3636619
## departmentEducation	0.1234904	0.3435377
## departmentEnergy & Resources Group	0.6621005	0.4114066
## departmentEnv Sci, Policy, & Mgmt	0.1485869	0.3921866
## departmentEthnic Studies Grad Grp	-0.5464231	0.5996170
## departmentHistory	-0.1648226	0.3431664
## departmentIndustrial Eng & Ops Rsch	0.2454338	0.6053634
## departmentInformation	0.2750037	0.5174054
## departmentIntegrative Biology	0.5698762	0.3364553
## departmentJSP Grad Pgm	-0.4288086	0.7210600
## departmentLaw	0.6621005	0.3866430
## departmentLinguistics	0.9274066	0.4800139
## departmentMusic	-0.5464231	0.4334402
## departmentNeuroscience	0.6621005	0.9235061
## departmentPolitical Science	0.3541044	0.2943577
## departmentPsychology	0.6858647	0.3178332
## departmentPublic Health	0.4345019	0.2604548
## departmentPublic Policy	0.2930528	0.2905775
## departmentRhetoric	-7.3378995	1.3298262
## departmentSlavic Languages & Lit	0.0578387	0.2557124
## departmentSociology	0.6621005	0.3254963
## outside.barriers:departmentAg & Resource Econ & Pol	-0.4947727	0.1356053
## outside.barriers:departmentAnthropology	-0.1819317	0.1627797
## outside.barriers:departmentApp Sci & Tech Grad Grp	0.0013720	0.2240160
## outside.barriers:departmentBiostatistics Grad Grp	0.3230109	0.2478656
## outside.barriers:departmentCity & Regional Planning	-0.3542618	0.3517710
## outside.barriers:departmentEconomics	-0.5880893	0.1732220
## outside.barriers:departmentEducation	-0.1930649	0.1370691
## outside.barriers:departmentEnergy & Resources Group	-0.6042618	0.1858492
## outside.barriers:departmentEnv Sci, Policy, & Mgmt	-0.1448023	0.1699881
## outside.barriers:departmentEthnic Studies Grad Grp	NA	NA
## outside.barriers:departmentHistory	0.1601613	0.1545218
## outside.barriers:departmentIndustrial Eng & Ops Rsch	-0.2709285	0.2621456
## outside.barriers:departmentInformation	-0.2816812	0.2476624
## outside.barriers:departmentIntegrative Biology	-0.4541714	0.1387724
## outside.barriers:departmentJSP Grad Pgm	0.3048291	0.3692532
## outside.barriers:departmentLaw	-0.6042618	0.1815371
## outside.barriers:departmentLinguistics	-0.6246700	0.2074324
## outside.barriers:departmentMusic	NA	NA
## outside.barriers:departmentNeuroscience	-0.6042618	0.6850431
## outside.barriers:departmentPolitical Science	-0.2878748	0.1320162
## outside.barriers:departmentPsychology	-0.4341097	0.1425093
## outside.barriers:departmentPublic Health	-0.4340109	0.1185779
## outside.barriers:departmentPublic Policy	-0.2649761	0.1327705
## outside.barriers:departmentRhetoric	2.1457382	0.4109273
## outside.barriers:departmentSlavic Languages & Lit	NA	NA
## outside.barriers:departmentSociology	-0.4996106	0.1372998
##	t value	Pr(> t )
## (Intercept)	1.486	0.138151

## outside.barriers	5.636	3.22e-08	***
## departmentAg & Resource Econ & Pol	1.634	0.102964	
## departmentAnthropology	0.234	0.814794	
## departmentApp Sci & Tech Grad Grp	0.000	0.999802	
## departmentBiostatistics Grad Grp	-1.350	0.177895	
## departmentCity & Regional Planning	0.836	0.403848	
## departmentEconomics	2.013	0.044719	*
## departmentEducation	0.359	0.719428	
## departmentEnergy & Resources Group	1.609	0.108295	
## departmentEnv Sci, Policy, & Mgmt	0.379	0.704979	
## departmentEthnic Studies Grad Grp	-0.911	0.362671	
## departmentHistory	-0.480	0.631266	
## departmentIndustrial Eng & Ops Rsch	0.405	0.685368	
## departmentInformation	0.532	0.595352	
## departmentIntegrative Biology	1.694	0.091057	.
## departmentJSP Grad Pgm	-0.595	0.552372	
## departmentLaw	1.712	0.087560	.
## departmentLinguistics	1.932	0.054032	.
## departmentMusic	-1.261	0.208134	
## departmentNeuroscience	0.717	0.473811	
## departmentPolitical Science	1.203	0.229669	
## departmentPsychology	2.158	0.031503	*
## departmentPublic Health	1.668	0.096018	.
## departmentPublic Policy	1.009	0.313790	
## departmentRhetoric	-5.518	6.03e-08	***
## departmentSlavic Languages & Lit	0.226	0.821167	
## departmentSociology	2.034	0.042571	*
## outside.barriers:departmentAg & Resource Econ & Pol	-3.649	0.000297	***
## outside.barriers:departmentAnthropology	-1.118	0.264358	
## outside.barriers:departmentApp Sci & Tech Grad Grp	0.006	0.995116	
## outside.barriers:departmentBiostatistics Grad Grp	1.303	0.193236	
## outside.barriers:departmentCity & Regional Planning	-1.007	0.314480	
## outside.barriers:departmentEconomics	-3.395	0.000752	***
## outside.barriers:departmentEducation	-1.409	0.159722	
## outside.barriers:departmentEnergy & Resources Group	-3.251	0.001242	**
## outside.barriers:departmentEnv Sci, Policy, & Mgmt	-0.852	0.394793	
## outside.barriers:departmentEthnic Studies Grad Grp	NA	NA	
## outside.barriers:departmentHistory	1.036	0.300571	
## outside.barriers:departmentIndustrial Eng & Ops Rsch	-1.034	0.301967	
## outside.barriers:departmentInformation	-1.137	0.256041	
## outside.barriers:departmentIntegrative Biology	-3.273	0.001154	**
## outside.barriers:departmentJSP Grad Pgm	0.826	0.409544	
## outside.barriers:departmentLaw	-3.329	0.000950	***
## outside.barriers:departmentLinguistics	-3.011	0.002758	**
## outside.barriers:departmentMusic	NA	NA	
## outside.barriers:departmentNeuroscience	-0.882	0.378243	
## outside.barriers:departmentPolitical Science	-2.181	0.029771	*
## outside.barriers:departmentPsychology	-3.046	0.002465	**
## outside.barriers:departmentPublic Health	-3.660	0.000284	***
## outside.barriers:departmentPublic Policy	-1.996	0.046612	*
## outside.barriers:departmentRhetoric	5.222	2.80e-07	***
## outside.barriers:departmentSlavic Languages & Lit	NA	NA	
## outside.barriers:departmentSociology	-3.639	0.000308	***
## ---			

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.586 on 417 degrees of freedom
##   (594 observations deleted due to missingness)
## Multiple R-squared:  0.4357, Adjusted R-squared:  0.368
## F-statistic: 6.439 on 50 and 417 DF,  p-value: < 2.2e-16
```

extract model parameters with `$`

```
model.1$coefficients
```

```
##      (Intercept) outside.barriers
##      0.7529250      0.2463815
```

```
model.1$coefficients[[2]]
```

```
## [1] 0.2463815
```

this is useful if you want to plot residuals

```
dat$residuals <- model.1$residuals
```

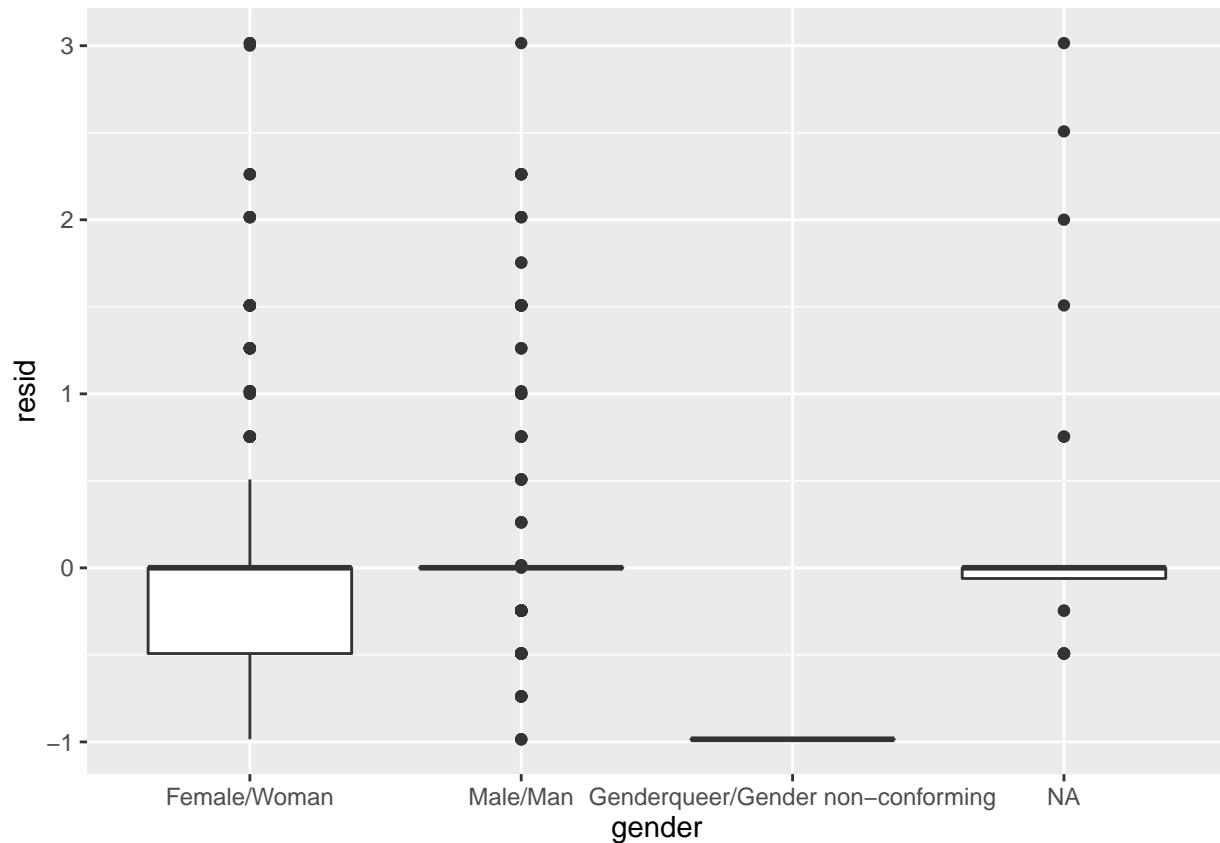
oh boy golly gee gosh darn! remember how we talked about R having casewise deletion + bad indexing? this is one place where it makes your life difficult

we have to do something like this:

```
dat.listwise <- dat[!is.na(dat$inside.barriers) & !is.na(dat$outside.barriers), ]
dat.listwise$resid <- model.1$residuals
```

then we can do this

```
ggplot(data = dat.listwise, aes(x=gender,y=resid)) +
  geom_boxplot()
```



## Nonparametric

parametric refers to using means, deviations, and other estimates of population parameters

*BUT* what if you don't want to make assumptions about the structure of the population?

or what if you **gasp** can't?

## ranked variables

a simple case is where means don't have meaning

above we were looking at correlations between Likert variables

all Likerts are really rank variables, which means they don't act like actual number-y numbers

in the real world, a 6 foot tall person is twice as tall as a 3 foot tall person

but is a level ‘6’ really twice as many barriers to access as a ‘3’?

NOPE

we know that 6 is more than 3, but can't really say how much - in that sense then, a scale of 1-7 is exactly the same thing as a scale of a-g.

median testing ranks

we use Mann-Whitney sums to test that the ranks are centered the same way

```
wilcox.test(dat$outside.barriers, dat$inside.barriers, alternative = "two.sided", paired = FALSE, mu = 0)
```

```
##  
## Wilcoxon rank sum test with continuity correction  
##  
## data: dat$outside.barriers and dat$inside.barriers  
## W = 541240, p-value < 2.2e-16  
## alternative hypothesis: true location shift is not equal to 0
```

see how this setup looks exactly like a t-test? that's not an accident

## correlating ranks

this is just like the `cor.test` you did above, but with `method` set to equal 'spearman' instead of pearson

```
cor.test(dat$outside.barriers, dat$inside.barriers, method = 'spearman')
```

```
## Warning in cor.test.default(dat$outside.barriers, dat$inside.barriers,  
## method = "spearman"): Cannot compute exact p-value with ties
```

```
##  
## Spearman's rank correlation rho  
##  
## data: dat$outside.barriers and dat$inside.barriers  
## S = 63037000, p-value < 2.2e-16  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho  
## 0.4524909
```

rho is pretty close to the r from above

## chisq

what if both of your variables are categories? we can test their counts with R's built in `chisq.test` function  
i.e. what if we want to know if gender is distributed evenly over departments?

```
chisq.test(dat$gender, dat$department)
```

```
## Warning in chisq.test(dat$gender, dat$department): Chi-squared  
## approximation may be incorrect
```

```
##  
## Pearson's Chi-squared test  
##  
## data: dat$gender and dat$department  
## X-squared = 76.442, df = 26, p-value = 7.326e-07
```

## Practice

### Assignment

There were a lot of variables in this dataset that we did not look at today:

```
names(data)
```

```
## NULL
```

Choose two of those variables, and explore their distribution and relationship to each other. Can you conclude anything about the D-Lab based on the feedback?

## Acknowledgements

Materials taken from:

[D-Lab's Feedback Analytics](#)