

final

2025-12-06

## Setup

```
# Load packages
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
library(ggplot2)
```

## Data loading

```
dat <- read.csv("student-mat.csv", sep = ";")
str(dat)

## 'data.frame':    395 obs. of  33 variables:
## $ school     : chr  "GP" "GP" "GP" "GP" ...
## $ sex        : chr  "F"  "F"  "F"  "F" ...
## $ age         : int  18 17 15 15 16 16 16 17 15 15 ...
## $ address     : chr  "U"  "U"  "U"  "U" ...
## $ famsize     : chr  "GT3" "GT3" "LE3" "GT3" ...
## $ Pstatus     : chr  "A"  "T"  "T"  "T" ...
## $ Medu        : int  4 1 1 4 3 4 2 4 3 3 ...
## $ Fedu        : int  4 1 1 2 3 3 2 4 2 4 ...
## $ Mjob        : chr  "at_home" "at_home" "at_home" "health" ...
## $ Fjob        : chr  "teacher" "other" "other" "services" ...
## $ reason      : chr  "course" "course" "other" "home" ...
## $ guardian    : chr  "mother" "father" "mother" "mother" ...
## $ traveltimes: int  2 1 1 1 1 1 2 1 1 ...
## $ studytime  : int  2 2 2 3 2 2 2 2 2 ...
## $ failures    : int  0 0 3 0 0 0 0 0 0 0 ...
## $ schoolsup   : chr  "yes" "no" "yes" "no" ...
## $ famsup      : chr  "no"  "yes" "no"  "yes" ...
## $ paid         : chr  "no"  "no" "yes" "yes" ...
## $ activities   : chr  "no"  "no" "no"  "yes" ...
## $ nursery      : chr  "yes" "no" "yes" "yes" ...
## $ higher       : chr  "yes" "yes" "yes" "yes" ...
```

```

## $ internet  : chr  "no" "yes" "yes" "yes" ...
## $ romantic   : chr  "no" "no" "no" "yes" ...
## $ famrel     : int  4 5 4 3 4 5 4 4 4 5 ...
## $ freetime    : int  3 3 3 2 3 4 4 1 2 5 ...
## $ goout       : int  4 3 2 2 2 2 4 4 2 1 ...
## $ Dalc        : int  1 1 2 1 1 1 1 1 1 1 ...
## $ Walc        : int  1 1 3 1 2 2 1 1 1 1 ...
## $ health      : int  3 3 3 5 5 5 3 1 1 5 ...
## $ absences    : int  6 4 10 2 4 10 0 6 0 0 ...
## $ G1          : int  5 5 7 15 6 15 12 6 16 14 ...
## $ G2          : int  6 5 8 14 10 15 12 5 18 15 ...
## $ G3          : int  6 6 10 15 10 15 11 6 19 15 ...

```

## Data preprocessing

The data is pretty clean actually from visual inspection, we just need to cast certain categorical variables into numerical with `factor()`.

```

dat <- dat %>%
  mutate(
    # Core categorical variables
    school    = factor(school),
    sex       = factor(sex),
    address   = factor(address),
    famsize   = factor(famsize),
    Pstatus   = factor(Pstatus),

    # Jobs & reasons
    Mjob      = factor(Mjob),
    Fjob      = factor(Fjob),
    reason    = factor(reason),
    guardian  = factor(guardian),

    # Supports and activities
    schoolsup = factor(schoolsup),
    famsup    = factor(famsup),
    paid      = factor(paid),
    activities= factor(activities),
    nursery   = factor(nursery),
    higher    = factor(higher),
    internet  = factor(internet),
    romantic  = factor(romantic),

    # Study time as ordered-ish factor
    studytime = factor(
      studytime,
      levels = 1:4,
      labels = c("low", "moderate", "high", "very high")
    )
  )

str(dat)

## 'data.frame': 395 obs. of 33 variables:

```

```

## $ school : Factor w/ 2 levels "GP","MS": 1 1 1 1 1 1 1 1 1 1 ...
## $ sex : Factor w/ 2 levels "F","M": 1 1 1 1 1 2 2 1 2 2 ...
## $ age : int 18 17 15 15 16 16 17 15 15 ...
## $ address : Factor w/ 2 levels "R","U": 2 2 2 2 2 2 2 2 2 2 ...
## $ famsize : Factor w/ 2 levels "GT3","LE3": 1 1 2 1 1 2 2 1 2 1 ...
## $ Pstatus : Factor w/ 2 levels "A","T": 1 2 2 2 2 2 2 1 1 2 ...
## $ Medu : int 4 1 1 4 3 4 2 4 3 3 ...
## $ Fedu : int 4 1 1 2 3 3 2 4 2 4 ...
## $ Mjob : Factor w/ 5 levels "at_home","health",...: 1 1 1 2 3 4 3 3 4 3 ...
## $ Fjob : Factor w/ 5 levels "at_home","health",...: 5 3 3 4 3 3 3 5 3 3 ...
## $ reason : Factor w/ 4 levels "course","home",...: 1 1 3 2 2 4 2 2 2 2 ...
## $ guardian : Factor w/ 3 levels "father","mother",...: 2 1 2 2 1 2 2 2 2 2 ...
## $ traveltimes: int 2 1 1 1 1 1 2 1 1 ...
## $ studytime : Factor w/ 4 levels "low","moderate",...: 2 2 2 3 2 2 2 2 2 2 ...
## $ failures : int 0 0 3 0 0 0 0 0 0 0 ...
## $ schoolsup : Factor w/ 2 levels "no","yes": 2 1 2 1 1 1 1 2 1 1 ...
## $ famsup : Factor w/ 2 levels "no","yes": 1 2 1 2 2 2 1 2 2 2 ...
## $ paid : Factor w/ 2 levels "no","yes": 1 1 2 2 2 2 1 1 2 2 ...
## $ activities: Factor w/ 2 levels "no","yes": 1 1 1 2 1 2 1 1 1 2 ...
## $ nursery : Factor w/ 2 levels "no","yes": 2 1 2 2 2 2 2 2 2 2 ...
## $ higher : Factor w/ 2 levels "no","yes": 2 2 2 2 2 2 2 2 2 2 ...
## $ internet : Factor w/ 2 levels "no","yes": 1 2 2 2 1 2 2 1 2 2 ...
## $ romantic : Factor w/ 2 levels "no","yes": 1 1 1 2 1 1 1 1 1 1 ...
## $ famrel : int 4 5 4 3 4 5 4 4 4 5 ...
## $ freetime : int 3 3 3 2 3 4 4 1 2 5 ...
## $ goout : int 4 3 2 2 2 2 4 4 2 1 ...
## $ Dalc : int 1 1 2 1 1 1 1 1 1 1 ...
## $ Walc : int 1 1 3 1 2 2 1 1 1 1 ...
## $ health : int 3 3 3 5 5 5 3 1 1 5 ...
## $ absences : int 6 4 10 2 4 10 0 6 0 0 ...
## $ G1 : int 5 5 7 15 6 15 12 6 16 14 ...
## $ G2 : int 6 5 8 14 10 15 12 5 18 15 ...
## $ G3 : int 6 6 10 15 10 15 11 6 19 15 ...

```

## Exploratory Data Analysis

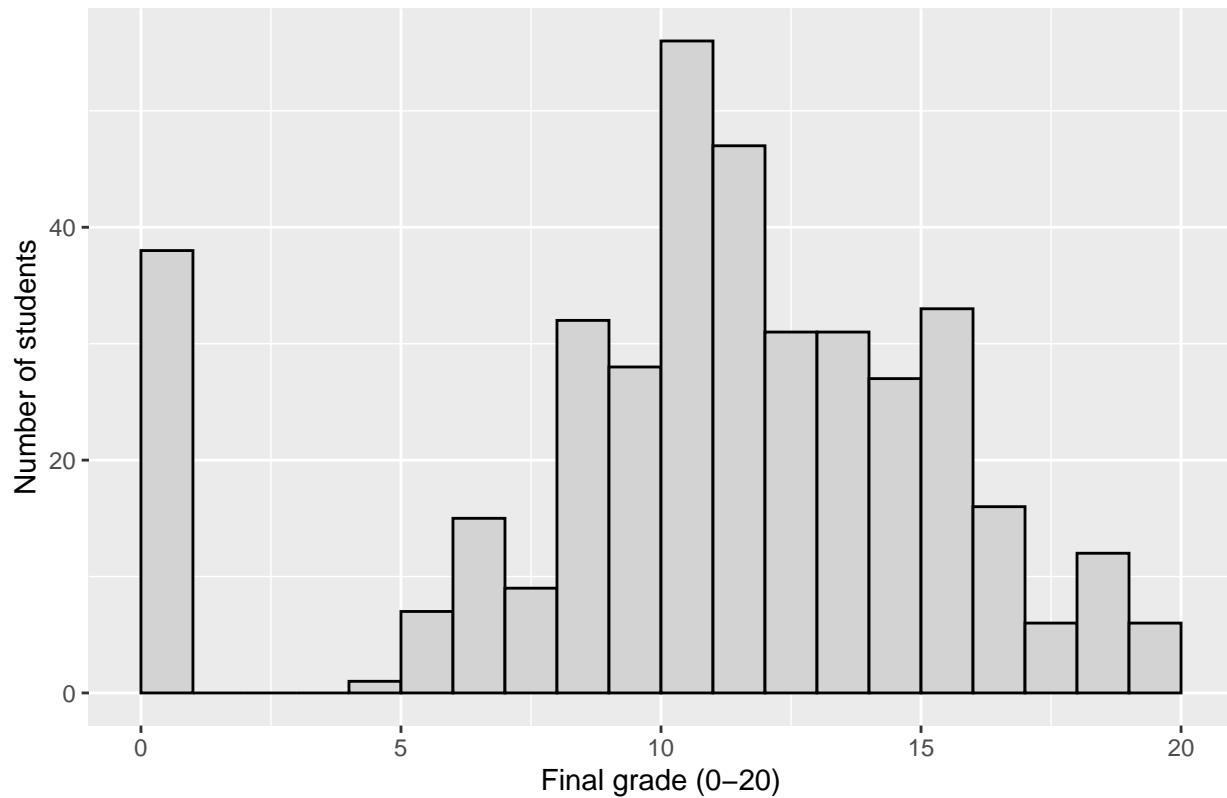
Now we move onto inspecting the data and seeing how different variables look. Firstly, let's look at the grade distribution for the final math grade:

```

ggplot(dat, aes(x = G3)) +
  geom_histogram(binwidth = 1, boundary = 0, closed = "left", color = "black", fill = "lightgray") +
  labs(
    title = "Distribution of Final Grade (G3)",
    x = "Final grade (0-20)",
    y = "Number of students"
  )

```

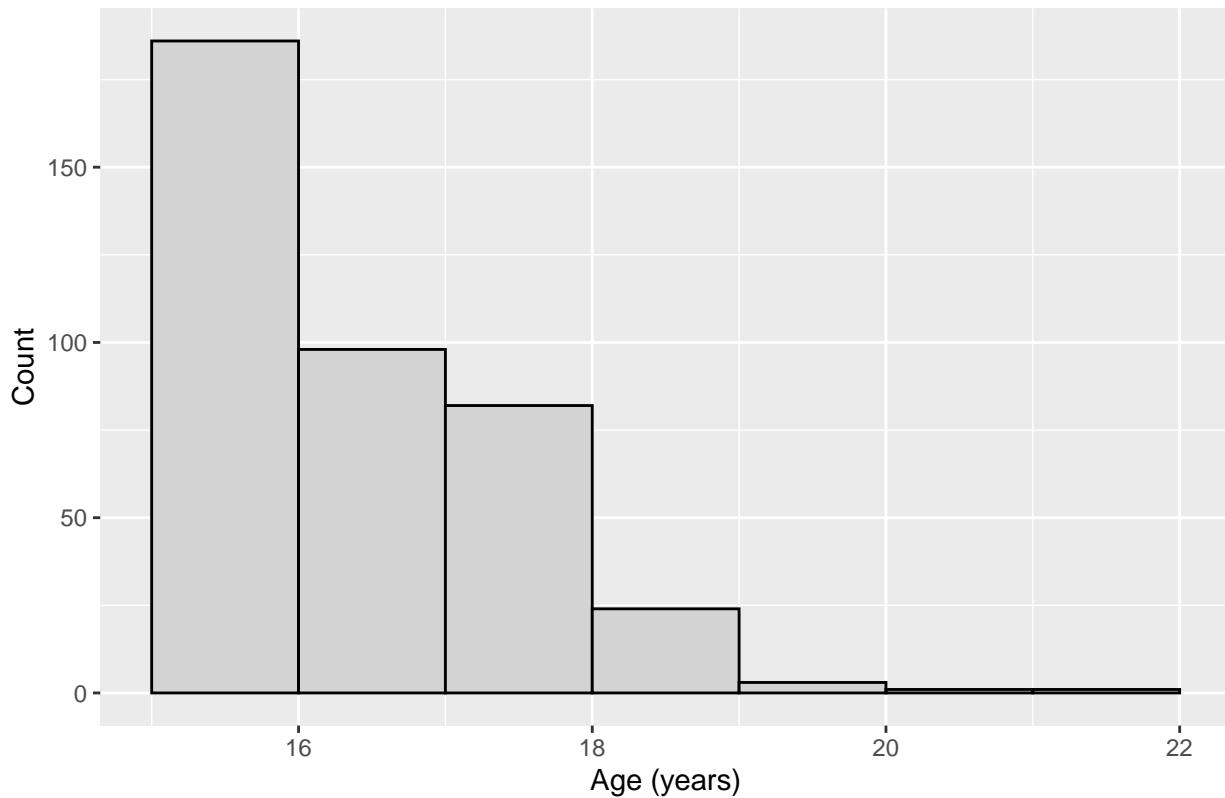
## Distribution of Final Grade (G3)



Observing distributions of some numeric variables like age, absences, failures.

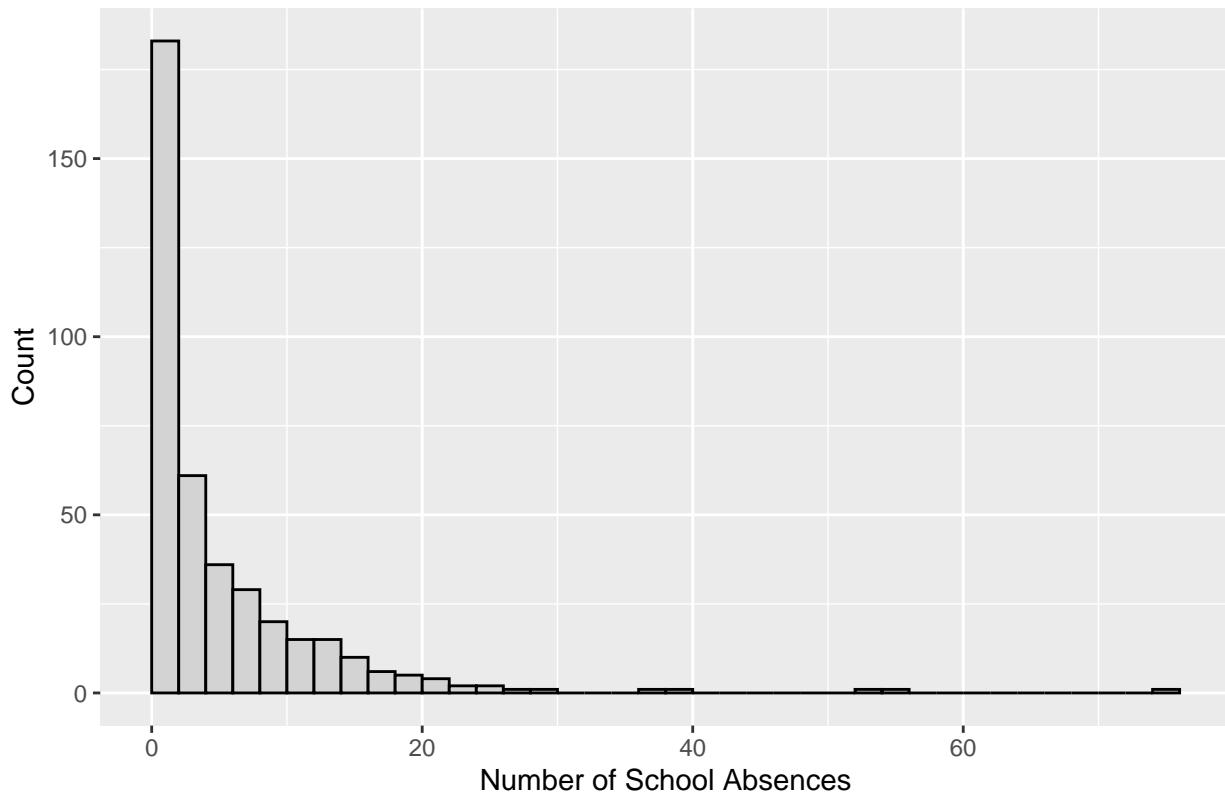
```
ggplot(dat, aes(x = age)) +
  geom_histogram(
    binwidth = 1,
    boundary = 0,
    color = "black",
    fill = "lightgray"
  ) +
  labs(
    title = "Distribution of Age",
    x = "Age (years)",
    y = "Count"
  )
```

## Distribution of Age



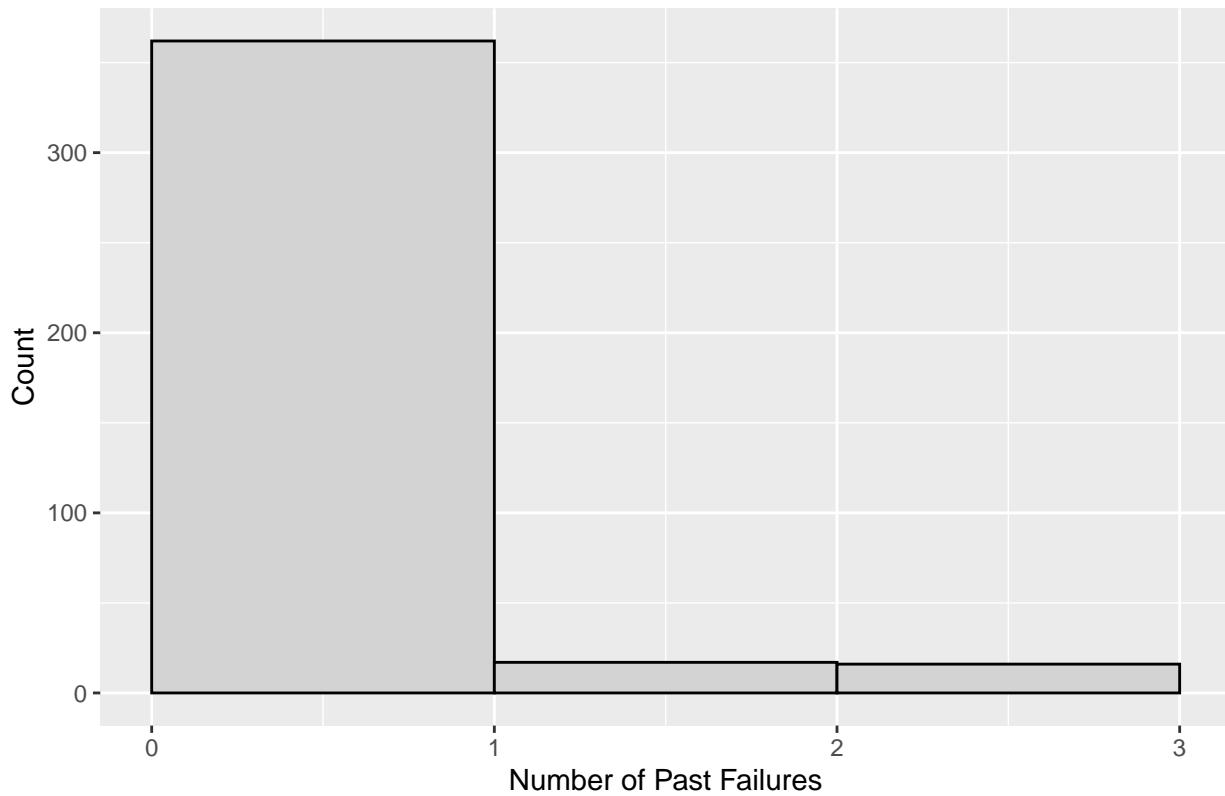
```
ggplot(dat, aes(x = absences)) +  
  geom_histogram(  
    binwidth = 2,  
    boundary = 0,  
    color = "black",  
    fill = "lightgray"  
) +  
  labs(  
    title = "Distribution of Absences",  
    x = "Number of School Absences",  
    y = "Count"  
)
```

## Distribution of Absences



```
ggplot(dat, aes(x = failures)) +
  geom_histogram(
    binwidth = 1,
    boundary = 0,
    color = "black",
    fill = "lightgray"
  ) +
  labs(
    title = "Distribution of Past Failures",
    x = "Number of Past Failures",
    y = "Count"
  )
```

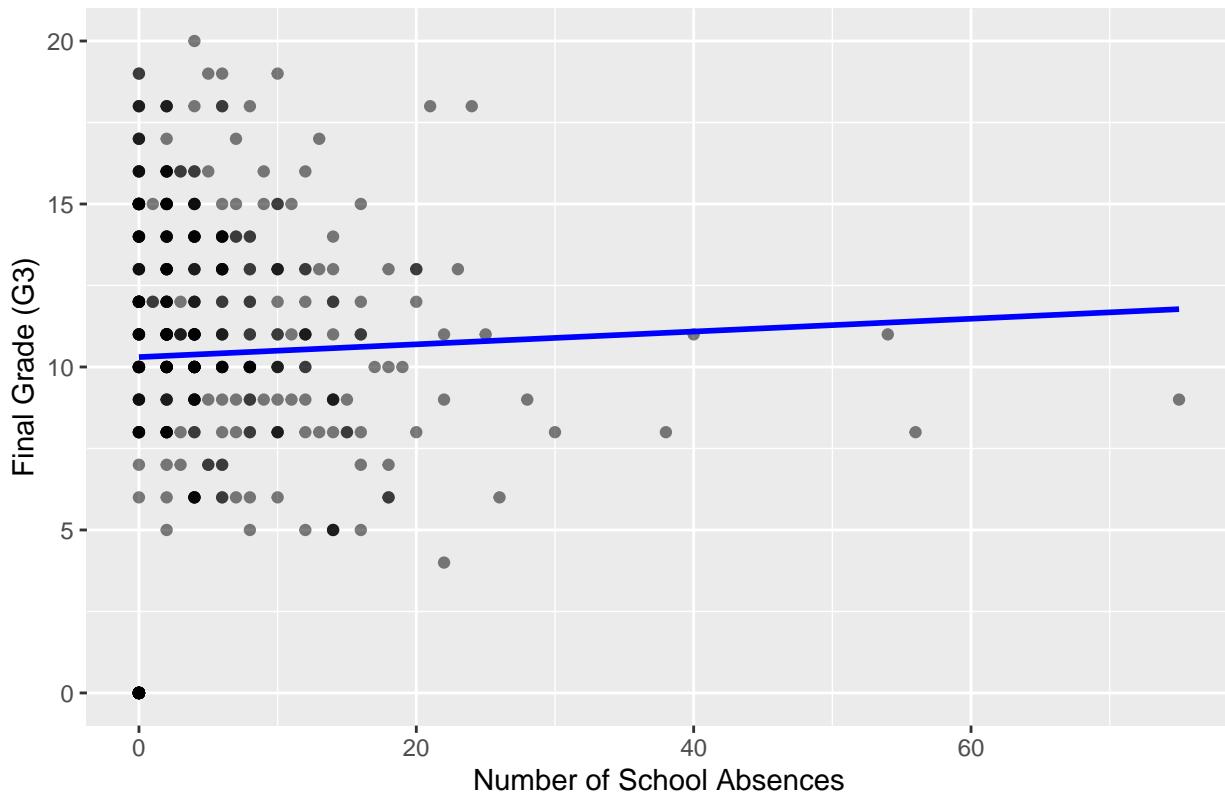
## Distribution of Past Failures



Now to contrast the final grade with some of the variables of interest.

```
ggplot(dat, aes(x = absences, y = G3)) +  
  geom_point(alpha = 0.5) +  
  geom_smooth(method = "lm", se = FALSE, color = "blue") +  
  labs(  
    title = "Final Grade (G3) vs Absences",  
    x = "Number of School Absences",  
    y = "Final Grade (G3)"  
)  
  
## `geom_smooth()` using formula = 'y ~ x'
```

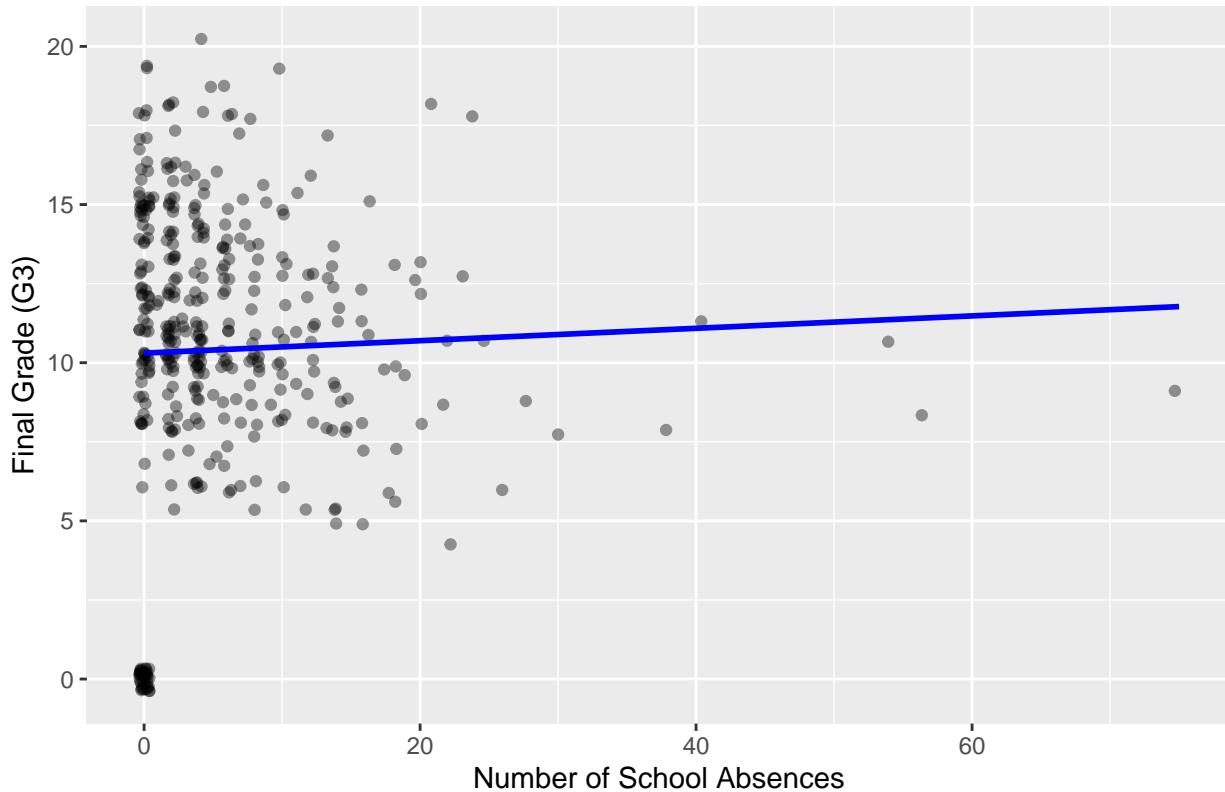
## Final Grade (G3) vs Absences



Adding some jitter for a better visual understanding of the quantity of different data points.

```
ggplot(dat, aes(x = absences, y = G3)) +  
  geom_jitter(width = 0.4, alpha = 0.4) +  
  geom_smooth(method = "lm", se = FALSE, color = "blue") +  
  labs(  
    title = "Final Grade (G3) vs Absences (Jittered)",  
    x = "Number of School Absences",  
    y = "Final Grade (G3)"  
)  
  
## `geom_smooth()` using formula = 'y ~ x'
```

## Final Grade (G3) vs Absences (Jittered)

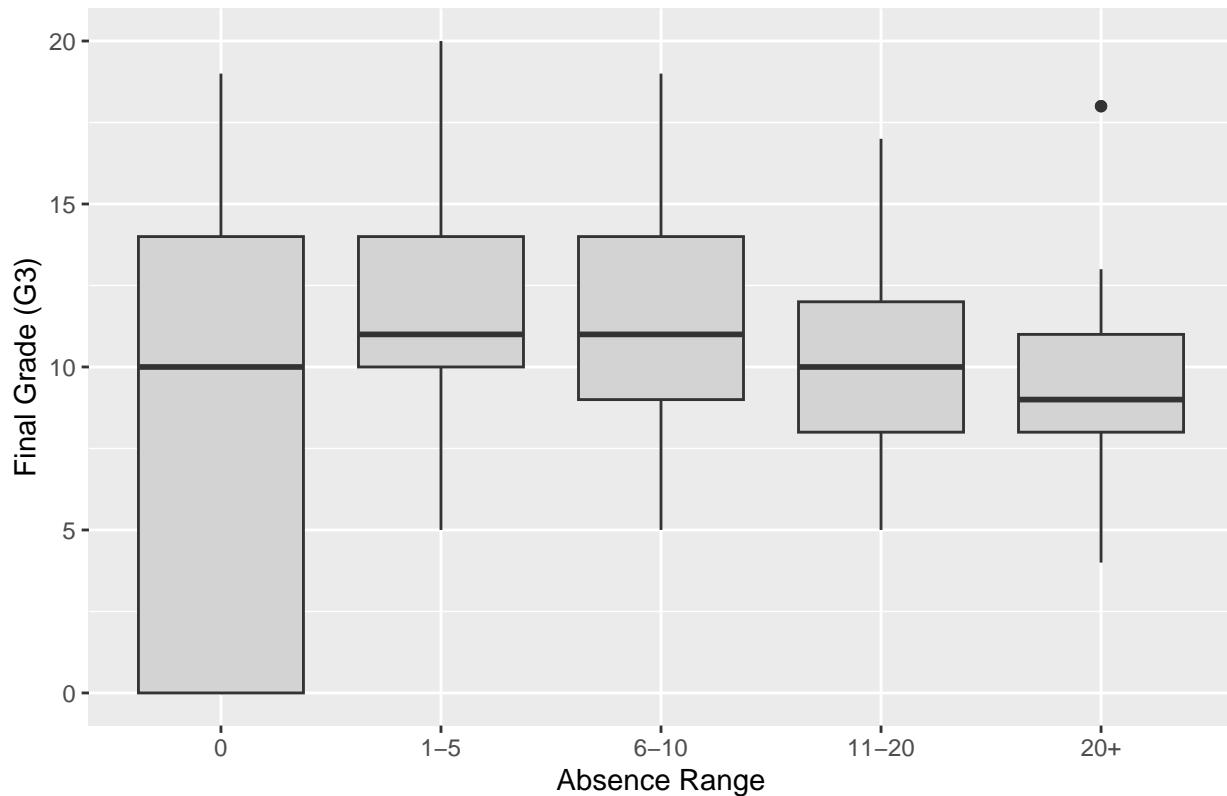


We can also visually show the disparities between different categorical groups with a box plot.

```
dat <- dat %>%
  mutate(abs_bin = cut(
    absences,
    breaks = c(-1, 0, 5, 10, 20, 100),
    labels = c("0", "1-5", "6-10", "11-20", "20+")
  ))

ggplot(dat, aes(x = abs_bin, y = G3)) +
  geom_boxplot(fill = "lightgray") +
  labs(
    title = "Final Grade (G3) by Absence Category",
    x = "Absence Range",
    y = "Final Grade (G3)"
  )
```

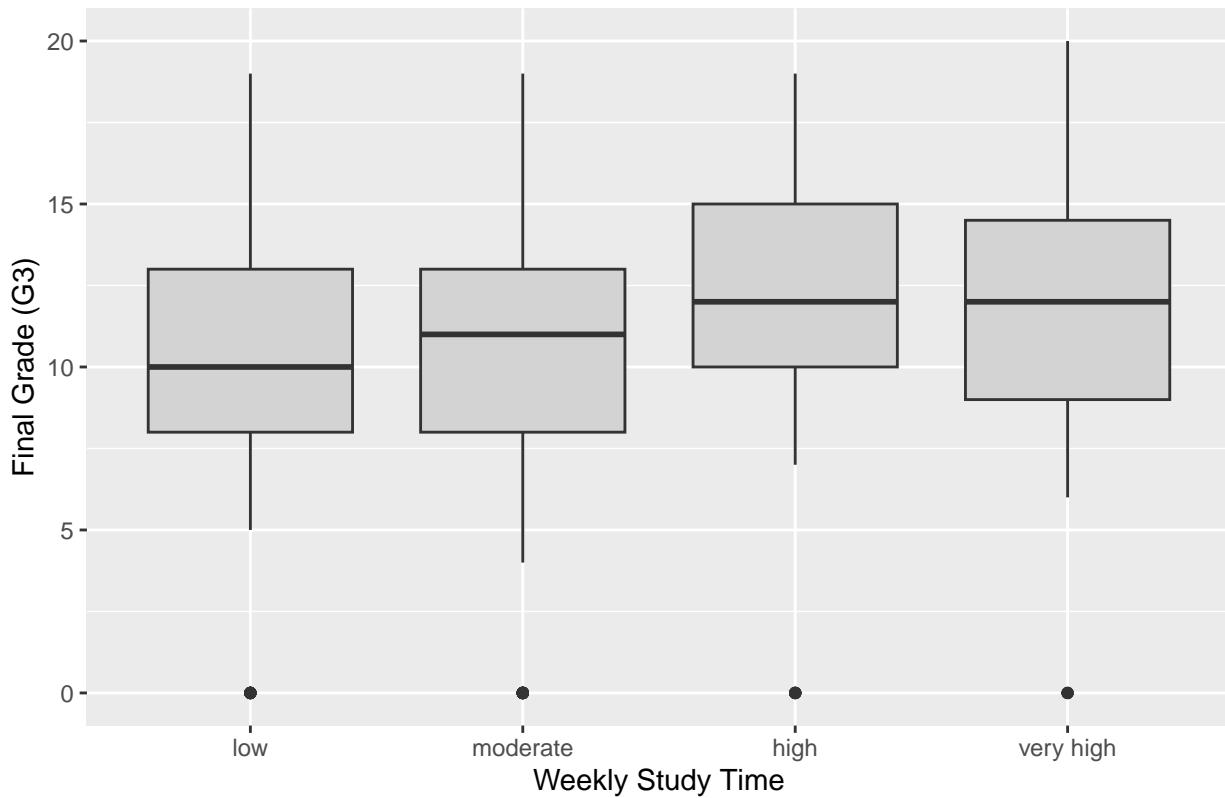
## Final Grade (G3) by Absence Category



Let's look at other variables like study time:

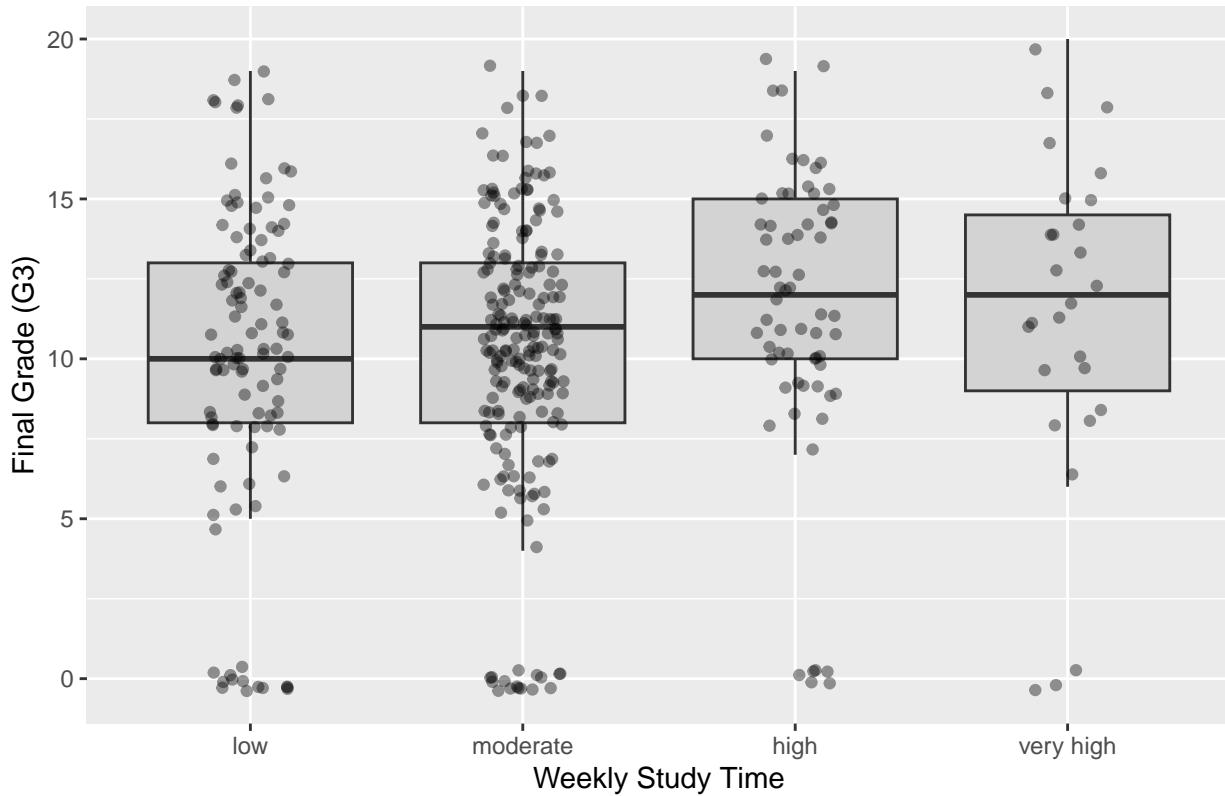
```
# Normal box plot
ggplot(dat, aes(x = studytime, y = G3)) +
  geom_boxplot(fill = "lightgray") +
  labs(
    title = "Final Grade (G3) by Study Time Category",
    x = "Weekly Study Time",
    y = "Final Grade (G3)"
  )
```

### Final Grade (G3) by Study Time Category



```
# Box plot including the jittered data points overlaid
ggplot(dat, aes(x = studystime, y = G3)) +
  geom_boxplot(fill = "lightgray", outlier.shape = NA) +
  geom_jitter(width = 0.15, alpha = 0.4) +
  labs(
    title = "Final Grade (G3) by Study Time Category",
    x = "Weekly Study Time",
    y = "Final Grade (G3)"
  )
```

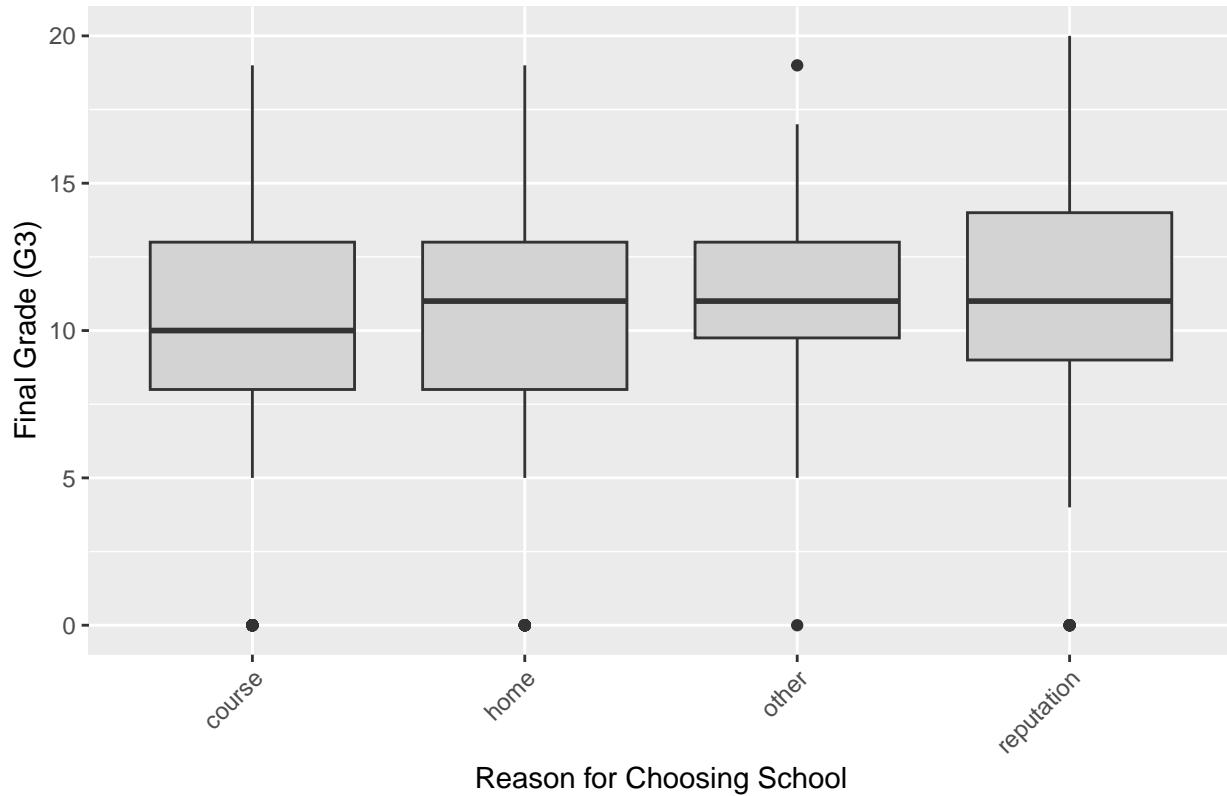
## Final Grade (G3) by Study Time Category



And also variables like reason for doing school:

```
ggplot(dat, aes(x = reason, y = G3)) +
  geom_boxplot(fill = "lightgray") +
  labs(
    title = "Final Grade (G3) by Reason for Choosing School",
    x = "Reason for Choosing School",
    y = "Final Grade (G3)"
  ) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

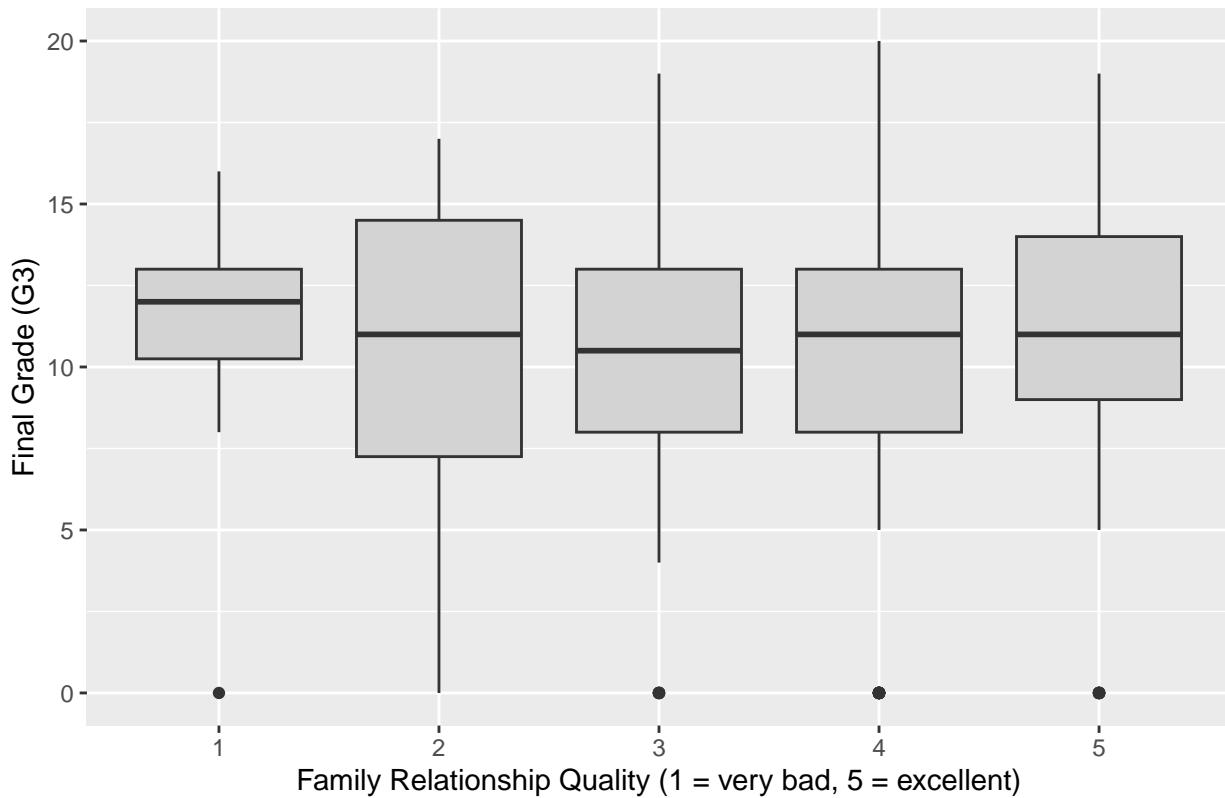
### Final Grade (G3) by Reason for Choosing School



Family relations:

```
ggplot(dat, aes(x = factor(famrel), y = G3)) +
  geom_boxplot(fill = "lightgray") +
  labs(
    title = "Final Grade (G3) by Family Relationship Quality",
    x = "Family Relationship Quality (1 = very bad, 5 = excellent)",
    y = "Final Grade (G3)"
  )
```

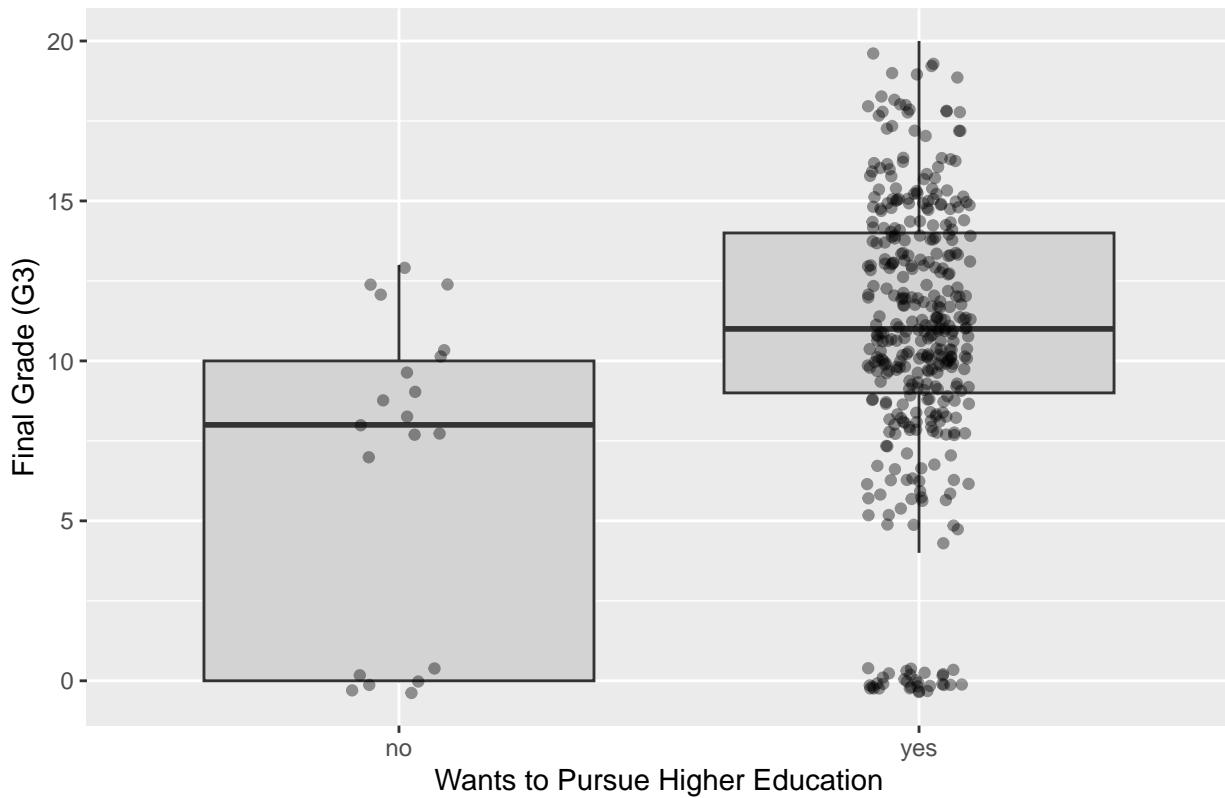
### Final Grade (G3) by Family Relationship Quality



A few more binary variables of interest like whether they want to pursue higher education, Internet access, rural/urban, and whether they have a romantic relationship.

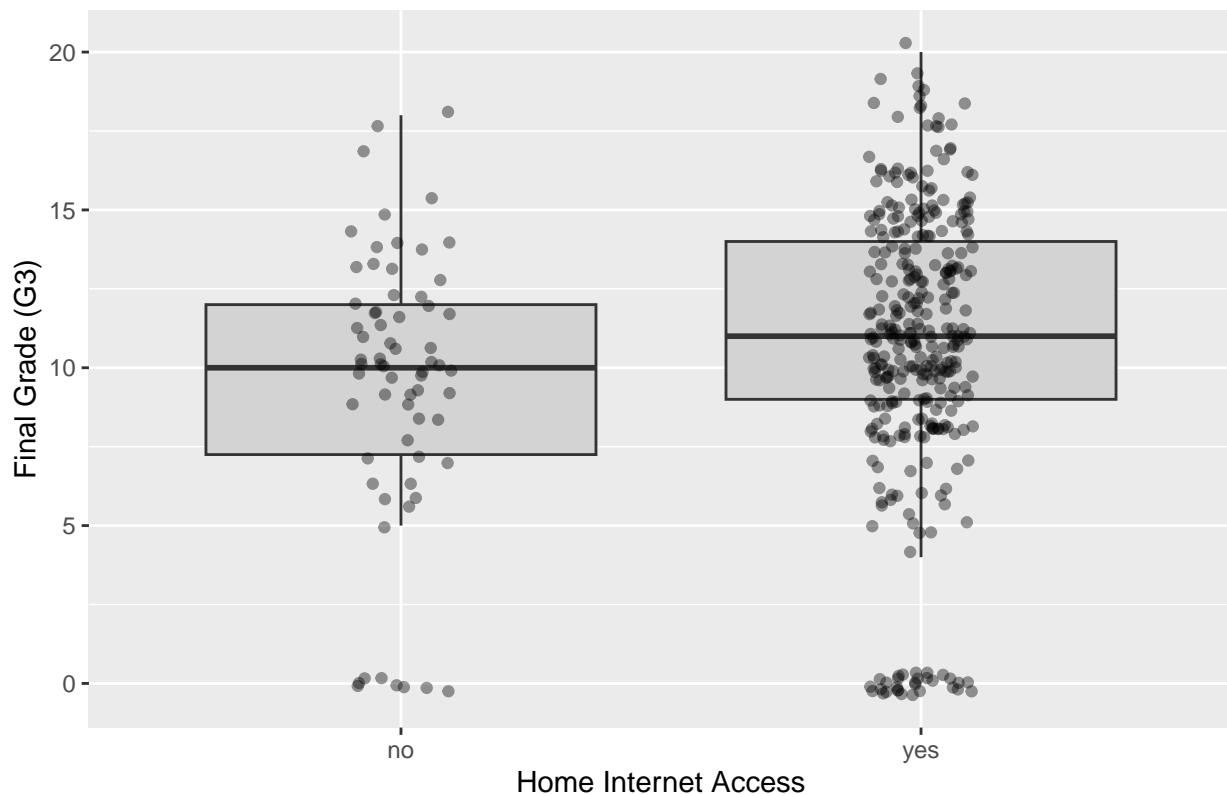
```
ggplot(dat, aes(x = higher, y = G3)) +
  geom_boxplot(fill = "lightgray", outlier.shape = NA) +
  geom_jitter(width = 0.1, alpha = 0.4) +
  labs(
    title = "Final Grade (G3) by Intention to Pursue Higher Education",
    x = "Wants to Pursue Higher Education",
    y = "Final Grade (G3)"
  )
```

### Final Grade (G3) by Intention to Pursue Higher Education



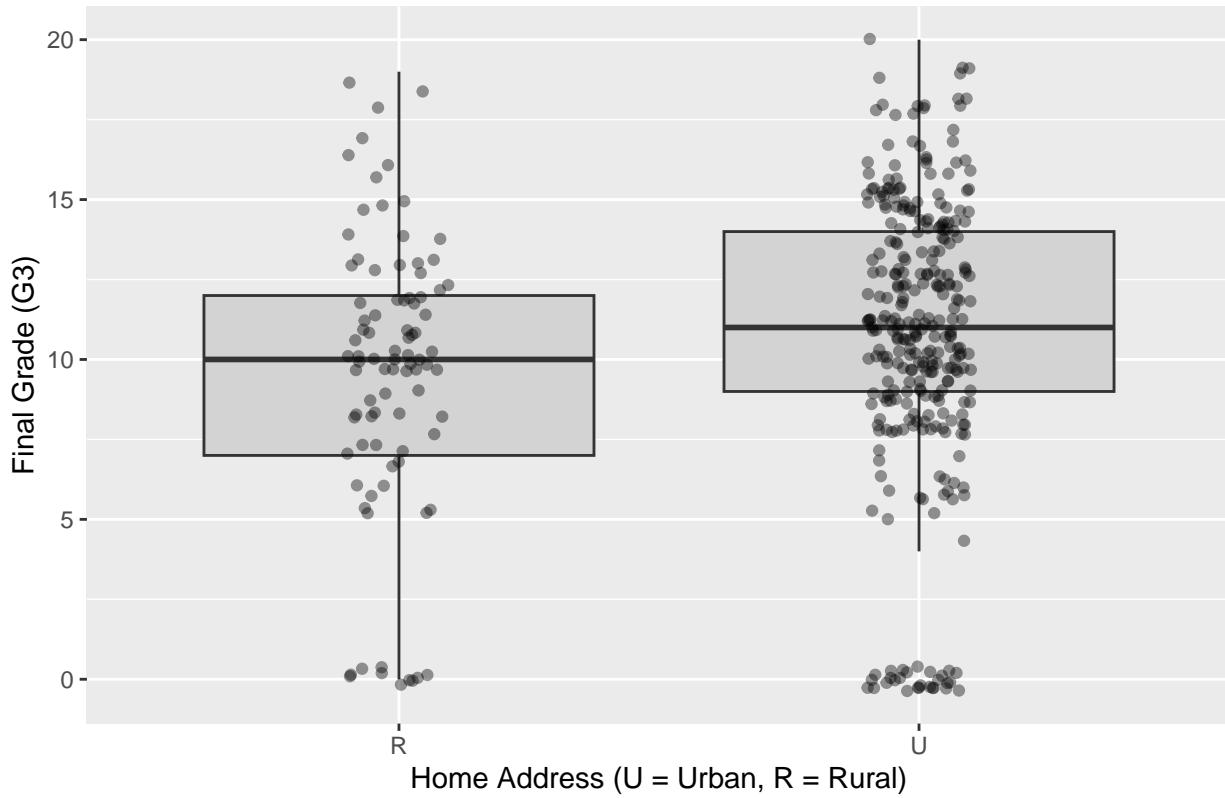
```
ggplot(dat, aes(x = internet, y = G3)) +  
  geom_boxplot(fill = "lightgray", outlier.shape = NA) +  
  geom_jitter(width = 0.1, alpha = 0.4) +  
  labs(  
    title = "Final Grade (G3) by Home Internet Access",  
    x = "Home Internet Access",  
    y = "Final Grade (G3)"  
)
```

### Final Grade (G3) by Home Internet Access



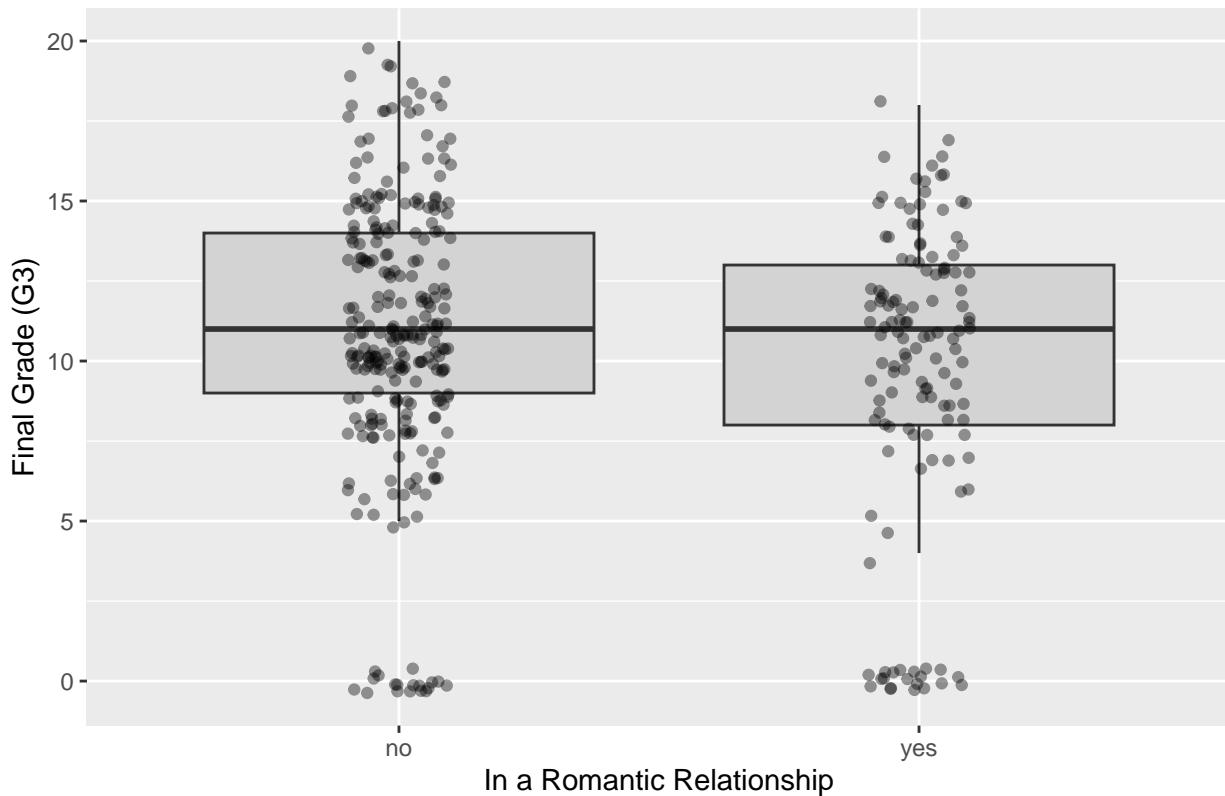
```
ggplot(dat, aes(x = address, y = G3)) +  
  geom_boxplot(fill = "lightgray", outlier.shape = NA) +  
  geom_jitter(width = 0.1, alpha = 0.4) +  
  labs(  
    title = "Final Grade (G3) by Home Address Type",  
    x = "Home Address (U = Urban, R = Rural)",  
    y = "Final Grade (G3)"  
)
```

### Final Grade (G3) by Home Address Type



```
ggplot(dat, aes(x = romantic, y = G3)) +  
  geom_boxplot(fill = "lightgray", outlier.shape = NA) +  
  geom_jitter(width = 0.1, alpha = 0.4) +  
  labs(  
    title = "Final Grade (G3) by Romantic Relationship Status",  
    x = "In a Romantic Relationship",  
    y = "Final Grade (G3)"  
)
```

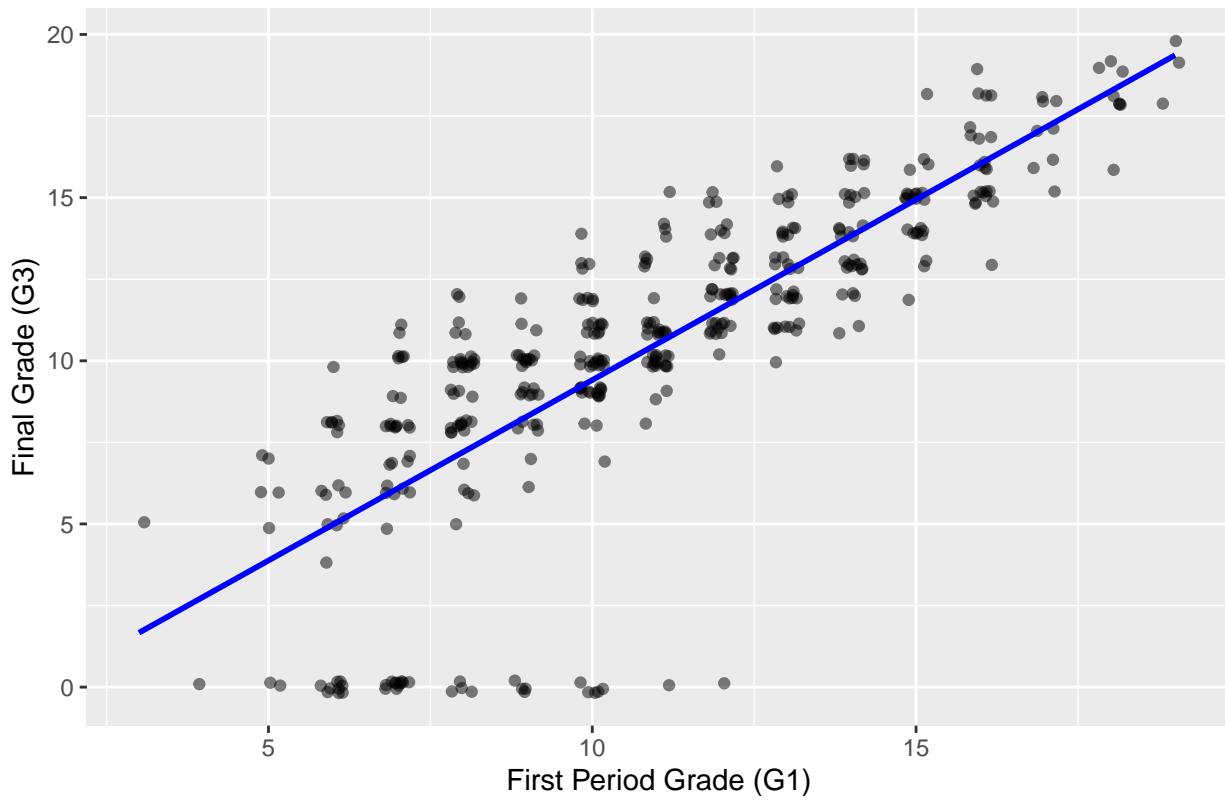
## Final Grade (G3) by Romantic Relationship Status



Also let's do a scatter plot analysis between G1 and G2, because this is the few strictly numeric vs numeric comparisons we have:

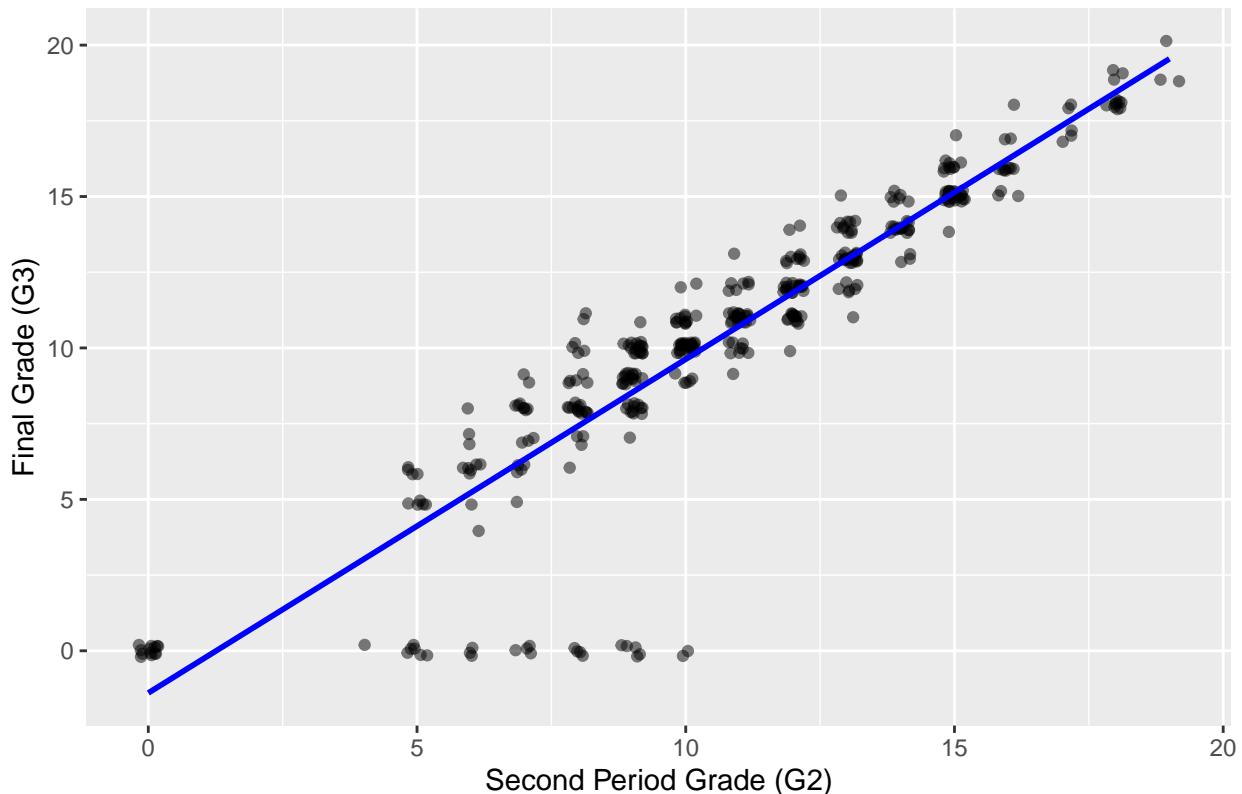
```
ggplot(dat, aes(x = G1, y = G3)) +  
  geom_jitter(width = 0.2, height = 0.2, alpha = 0.5) +  
  geom_smooth(method = "lm", se = FALSE, color = "blue") +  
  labs(  
    title = "Final Grade (G3) vs First Period Grade (G1)",  
    x = "First Period Grade (G1)",  
    y = "Final Grade (G3)"  
  )  
  
## `geom_smooth()` using formula = 'y ~ x'
```

### Final Grade (G3) vs First Period Grade (G1)



```
ggplot(dat, aes(x = G2, y = G3)) +
  geom_jitter(width = 0.2, height = 0.2, alpha = 0.5) +
  geom_smooth(method = "lm", se = FALSE, color = "blue") +
  labs(
    title = "Final Grade (G3) vs Second Period Grade (G2)",
    x = "Second Period Grade (G2)",
    y = "Final Grade (G3)"
  )
## `geom_smooth()` using formula = 'y ~ x'
```

## Final Grade (G3) vs Second Period Grade (G2)



So it looks like we have many variables that do show pretty big gaps in performance between different categorical groups. Now let's turn to exploring some regressions to see if we can quantify these relationships with the final math grade.

## Regressions

We'll reuse `dat` from above and fit several linear regression models

### Model 1: Background and behavior only (no prior grades)

Here we first focus on non-grade predictors to see how demographics, family background, and behaviors relate to the final grade, without using G1 or G2.

```
model1 <- lm(
  G3 ~ sex + age + Medu + Fedu +
    studytime + failures + absences +
    higher + goout + Walc + internet,
  data = dat
)

summary(model1)

## 
## Call:
## lm(formula = G3 ~ sex + age + Medu + Fedu + studytime + failures +
##     absences + higher + goout + Walc + internet, data = dat)
## 
```

```

## Residuals:
##      Min      1Q Median      3Q     Max
## -13.2241 -2.0116  0.4896  2.8137  8.7655
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)            11.56584   3.42278   3.379 0.000802 ***
## sexM                  1.37284   0.47235   2.906 0.003870 **
## age                   -0.21051   0.18045  -1.167 0.244091
## Medu                  0.49122   0.25664   1.914 0.056365 .
## Fedu                 -0.11465   0.25433  -0.451 0.652406
## studytimemoderate    0.07606   0.53510   0.142 0.887040
## studytimehigh         1.30893   0.73093   1.791 0.074124 .
## studytimeevery high  0.55157   0.94906   0.581 0.561468
## failures              -1.78689   0.31490  -5.674 2.76e-08 ***
## absences               0.04123   0.02779   1.484 0.138686
## higheryes              1.64756   1.04281   1.580 0.114955
## goout                  -0.45638   0.21340  -2.139 0.033104 *
## Walc                  0.09027   0.19390   0.466 0.641807
## internetyes             0.44558   0.58660   0.760 0.447962
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.178 on 381 degrees of freedom
## Multiple R-squared:  0.1959, Adjusted R-squared:  0.1685
## F-statistic: 7.141 on 13 and 381 DF,  p-value: 1.74e-12

```

## Model 2: Prior grades only (G1 and G2)

```

model2 <- lm(G3 ~ G1 + G2, data = dat)
summary(model2)

##
## Call:
## lm(formula = G3 ~ G1 + G2, data = dat)
##
## Residuals:
##      Min      1Q Median      3Q     Max
## -9.5713 -0.3888  0.2885  0.9725  3.7089
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.83001   0.33531  -5.458 8.57e-08 ***
## G1           0.15327   0.05618   2.728  0.00665 **
## G2           0.98687   0.04957  19.909 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.937 on 392 degrees of freedom
## Multiple R-squared:  0.8222, Adjusted R-squared:  0.8213
## F-statistic: 906.1 on 2 and 392 DF,  p-value: < 2.2e-16

```

### Model 3: Combined model (prior grades + background and behavior)

```

model3_full <- lm(
  G3 ~ G1 + G2 +
  sex + age + Medu + Fedu +
  studytime + failures + absences +
  higher + goout + Walc + internet,
  data = dat
)

summary(model3_full)

##
## Call:
## lm(formula = G3 ~ G1 + G2 + sex + age + Medu + Fedu + studytime +
##     failures + absences + higher + goout + Walc + internet, data = dat)
##
## Residuals:
##    Min      1Q  Median      3Q     Max 
## -9.0225 -0.4694  0.2749  0.9955  3.6924 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 1.25742   1.58303   0.794  0.427511  
## G1          0.16100   0.05741   2.804  0.005302 ** 
## G2          0.96538   0.05040  19.153 < 2e-16 *** 
## sexM        0.19145   0.21834   0.877  0.381128  
## age         -0.19603   0.08299  -2.362  0.018681 *  
## Medu        0.09254   0.11745   0.788  0.431278  
## Fedu        -0.16847   0.11634  -1.448  0.148402  
## studytimemoderate -0.11125   0.24381  -0.456  0.648438 
## studytimehigh   0.13159   0.33603   0.392  0.695578 
## studytimeevery high -0.78727   0.43380  -1.815  0.070342 . 
## failures      -0.25375   0.14943  -1.698  0.090297 .  
## absences       0.04264   0.01266   3.368  0.000834 *** 
## higheryes     0.24349   0.47667   0.511  0.609776  
## goout         0.06398   0.09822   0.651  0.515153  
## Walc          0.06668   0.08887   0.750  0.453523  
## internetyes   -0.29108   0.26863  -1.084  0.279252 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.903 on 379 degrees of freedom
## Multiple R-squared:  0.834, Adjusted R-squared:  0.8274 
## F-statistic:  127 on 15 and 379 DF,  p-value: < 2.2e-16

```

### Stepwise model

```

model_step <- step(model3_full, direction = "both", trace = FALSE)
summary(model_step)

##
## Call:
## lm(formula = G3 ~ G1 + G2 + age + failures + absences + Walc,

```

```

##      data = dat)
##
## Residuals:
##      Min    1Q Median    3Q   Max
## -8.9517 -0.4347  0.2737  0.9542  3.7652
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.80205   1.35383   0.592  0.55390
## G1          0.16845   0.05628   2.993  0.00294 **
## G2          0.95808   0.04962  19.309 < 2e-16 ***
## age         -0.17375   0.07963  -2.182  0.02971 *
## failures    -0.21104   0.14266  -1.479  0.13987
## absences     0.03974   0.01224   3.247  0.00127 **
## Walc        0.11149   0.07635   1.460  0.14504
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.901 on 388 degrees of freedom
## Multiple R-squared:  0.8305, Adjusted R-squared:  0.8279
## F-statistic: 316.9 on 6 and 388 DF,  p-value: < 2.2e-16

```

## Comparing the models

To summarize how much each model explains, we can extract the  $R^2$  and adjusted  $R^2$  values:

```

get_fit <- function(mod) {
  s <- summary(mod)
  data.frame(
    R2 = s$r.squared,
    Adj_R2 = s$adj.r.squared
  )
}

model_fits <- rbind(
  Model1_no_grades = get_fit(model1),
  Model2_grades_only = get_fit(model2),
  Model3_full      = get_fit(model3_full),
  Model_stepwise   = get_fit(model_step)
)

model_fits

##           R2     Adj_R2
## Model1_no_grades 0.1959106 0.1684745
## Model2_grades_only 0.8221632 0.8212559
## Model3_full       0.8340173 0.8274481
## Model_stepwise    0.8305031 0.8278820

```