EDA Practice Solution

Thanks to Andy Johnson for the use of his code/explanations.

Task 1 - Read in the data and Modify

First, we use the code supplied with the data to load in the two datasets and join them of the specified fields. We should note a few things with this process:

- all the functions used below are from the base R installation,
- though the datafiles end in `.csv`, the delimiter is not a comma, but rather a semicolon,
- the join type specified in the original code is an outer join, and
- the join operation returns 382 rows of data as stored in a data frame.

```
# load the two tables from local files
mat <- read.table("data/student-mat.csv", sep=";", header=TRUE)
por <- read.table("data/student-por.csv", sep=";", header=TRUE)

# join them using the supplied code (an outer join on several fields)
dat_via_merge <- merge(x=mat, y=por,
                       by=c("school","sex","age","address","famsize","Pstatus","Medu",
                            "Fedu","Mjob","Fjob","reason","nursery","internet"))

# show the table and clean up
head(dat_via_merge)
```

```
##   school sex age address famsize Pstatus Medu Fedu     Mjob     Fjob     reason
## 1     GP   F  15       R     GT3       T    1    1  at_home    other       home
## 2     GP   F  15       R     GT3       T    1    1    other    other reputation
## 3     GP   F  15       R     GT3       T    2    2  at_home    other reputation
## 4     GP   F  15       R     GT3       T    2    4 services   health     course
## 5     GP   F  15       R     GT3       T    3    3 services services reputation
## 6     GP   F  15       R     GT3       T    3    4 services   health     course
##   nursery internet guardian.x traveltime.x studytime.x failures.x schoolsup.x
## 1     yes      yes     mother            2           4          1         yes
## 2      no      yes     mother            1           2          2         yes
## 3     yes       no     mother            1           1          0         yes
## 4     yes      yes     mother            1           3          0         yes
## 5     yes      yes      other            2           3          2          no
## 6     yes      yes     mother            1           3          0         yes
##   famsup.x paid.x activities.x higher.x romantic.x famrel.x freetime.x goout.x
## 1      yes    yes          yes      yes         no        3          1       2
## 2      yes     no           no      yes        yes        3          3       4
## 3      yes    yes          yes      yes         no        4          3       1
## 4      yes    yes          yes      yes         no        4          3       2
## 5      yes    yes          yes      yes        yes        4          2       1
## 6      yes    yes          yes      yes         no        4          3       2
```

```
##   Dalc.x Walc.x health.x absences.x G1.x G2.x G3.x guardian.y traveltime.y
## 1      1      1        1          2    7   10   10     mother            2
## 2      2      4        5          2    8    6    5     mother            1
## 3      1      1        2          8   14   13   13     mother            1
## 4      1      1        5          2   10    9    8     mother            1
## 5      2      3        3          8   10   10   10      other            2
## 6      1      1        5          2   12   12   11     mother            1
##   studytime.y failures.y schoolsup.y famsup.y paid.y activities.y higher.y
## 1           4          0         yes      yes    yes          yes      yes
## 2           2          0         yes      yes     no           no      yes
## 3           1          0         yes      yes     no          yes      yes
## 4           3          0         yes      yes     no          yes      yes
## 5           3          0          no      yes    yes          yes      yes
## 6           3          0         yes      yes     no          yes      yes
##   romantic.y famrel.y freetime.y goout.y Dalc.y Walc.y health.y absences.y G1.y
## 1         no        3          1       2      1      1        1          4   13
## 2        yes        3          3       4      2      4        5          2   13
## 3         no        4          3       1      1      1        2          8   14
## 4         no        4          3       2      1      1        5          2   10
## 5        yes        4          2       1      2      3        3          2   13
## 6         no        4          3       2      1      1        5          2   11
##   G2.y G3.y
## 1   13   13
## 2   11   11
## 3   13   12
## 4   11   10
## 5   13   13
## 6   12   12
```

```
rm(dat_via_merge, mat, por)
```

Next we will reload the files and conduct the merge using `tidyverse` functions. Here we note the following:

- we use the `read_delim()` function to import the datafiles, again specifying the semicolon as the delimiter,
- the join type specified here is an inner join,
- we join on all variables other than `G1`, `G2`, `G3`, and `absences`,
- note the suffixes added to the variables not used as join keys which indicate their source tables, and
- the joining operation returns 162 rows of data stored as a tibble.

```
# load the two tables from local files
mat <- read_delim("data/student-mat.csv", delim=";")
por <- read_delim("data/student-por.csv", delim=";")

# create the list of joining variables as the complement of those specified to avoid
inner_join_vars <- colnames(mat)[!colnames(mat) %in% c("G1", "G2", "G3", "absences", "paid")]

# join them as an inner join
dat <- inner_join(x = mat, y=por, by=inner_join_vars, suffix=c("_mat","_por"))

# show the tibble
dat
```

```
## # A tibble: 320 x 38
##    school sex     age address famsize Pstatus  Medu  Fedu Mjob     Fjob   reason
##    <chr>  <chr> <dbl> <chr>   <chr>   <chr>   <dbl> <dbl> <chr>    <chr>  <chr>
##  1 GP     F        18 U       GT3     A           4     4 at_home  teach~ course
##  2 GP     F        17 U       GT3     T           1     1 at_home  other  course
##  3 GP     F        15 U       GT3     T           4     2 health   servi~ home
##  4 GP     F        16 U       GT3     T           3     3 other    other  home
##  5 GP     M        16 U       LE3     T           4     3 services other  reput~
##  6 GP     M        16 U       LE3     T           2     2 other    other  home
##  7 GP     F        17 U       GT3     A           4     4 other    teach~ home
##  8 GP     M        15 U       LE3     A           3     2 services other  home
##  9 GP     M        15 U       GT3     T           3     4 other    other  home
## 10 GP     F        15 U       GT3     T           4     4 teacher  health reput~
## # i 310 more rows
## # i 27 more variables: guardian <chr>, traveltime <dbl>, studytime <dbl>,
## #   failures <dbl>, schoolsup <chr>, famsup <chr>, paid_mat <chr>,
## #   activities <chr>, nursery <chr>, higher <chr>, internet <chr>,
## #   romantic <chr>, famrel <dbl>, freetime <dbl>, goout <dbl>, Dalc <dbl>,
## #   Walc <dbl>, health <dbl>, absences_mat <dbl>, G1_mat <dbl>, G2_mat <dbl>,
## #   G3_mat <dbl>, paid_por <chr>, absences_por <dbl>, G1_por <dbl>, ...
```

Finally, we will select four categorical variables (shared across the `mat`, `por`, and combined `dat` tibbles) for use in the following analyses, and convert them to factors. The specific variables I have chosen are:

- `address`: rural vs urban home location
- 'reason": why the specific school was chosen
- `internet`: if the student has internet access at home
- `higher`: if the student wants to enter into higher education

```
# use mutate() to convert to factor
dat <- mutate(dat, across(c(address, reason, internet, higher), as.factor))
mat <- mutate(mat, across(c(address, reason, internet, higher), as.factor))
por <- mutate(por, across(c(address, reason, internet, higher), as.factor))
```

Task 2 - Summarize the Data

First we can note how the variables are stored via `str()` or the default printing method on the `tibble`. We'll use `str()` here.

```
str(dat)
```

```
## tibble [320 x 38] (S3: tbl_df/tbl/data.frame)
##  $ school      : chr [1:320] "GP" "GP" "GP" "GP" ...
##  $ sex         : chr [1:320] "F" "F" "F" "F" ...
##  $ age         : num [1:320] 18 17 15 16 16 16 17 15 15 15 ...
##  $ address     : Factor w/ 2 levels "R","U": 2 2 2 2 2 2 2 2 2 2 ...
##  $ famsize     : chr [1:320] "GT3" "GT3" "GT3" "GT3" ...
##  $ Pstatus     : chr [1:320] "A" "T" "T" "T" ...
##  $ Medu        : num [1:320] 4 1 4 3 4 2 4 3 3 4 ...
##  $ Fedu        : num [1:320] 4 1 2 3 3 2 4 2 4 4 ...
##  $ Mjob        : chr [1:320] "at_home" "at_home" "health" "other" ...
##  $ Fjob        : chr [1:320] "teacher" "other" "services" "other" ...
```

```
##  $ reason      : Factor w/ 4 levels "course","home",..: 1 1 2 2 4 2 2 2 2 4 ...
##  $ guardian    : chr [1:320] "mother" "father" "mother" "father" ...
##  $ traveltime  : num [1:320] 2 1 1 1 1 1 2 1 1 1 ...
##  $ studytime   : num [1:320] 2 2 3 2 2 2 2 2 2 2 ...
##  $ failures    : num [1:320] 0 0 0 0 0 0 0 0 0 0 ...
##  $ schoolsup   : chr [1:320] "yes" "no" "no" "no" ...
##  $ famsup      : chr [1:320] "no" "yes" "yes" "yes" ...
##  $ paid_mat    : chr [1:320] "no" "no" "yes" "yes" ...
##  $ activities  : chr [1:320] "no" "no" "yes" "no" ...
##  $ nursery     : chr [1:320] "yes" "no" "yes" "yes" ...
##  $ 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 1 2 2 1 2 2 2 ...
##  $ romantic    : chr [1:320] "no" "no" "yes" "no" ...
##  $ famrel      : num [1:320] 4 5 3 4 5 4 4 4 5 3 ...
##  $ freetime    : num [1:320] 3 3 2 3 4 4 1 2 5 3 ...
##  $ goout       : num [1:320] 4 3 2 2 2 4 4 2 1 3 ...
##  $ Dalc        : num [1:320] 1 1 1 1 1 1 1 1 1 1 ...
##  $ Walc        : num [1:320] 1 1 1 2 2 1 1 1 1 2 ...
##  $ health      : num [1:320] 3 3 5 5 5 3 1 1 5 2 ...
##  $ absences_mat: num [1:320] 6 4 2 4 10 0 6 0 0 0 ...
##  $ G1_mat      : num [1:320] 5 5 15 6 15 12 6 16 14 10 ...
##  $ G2_mat      : num [1:320] 6 5 14 10 15 12 5 18 15 8 ...
##  $ G3_mat      : num [1:320] 6 6 15 10 15 11 6 19 15 9 ...
##  $ paid_por    : chr [1:320] "no" "no" "no" "no" ...
##  $ absences_por: num [1:320] 4 2 0 0 6 0 2 0 0 2 ...
##  $ G1_por      : num [1:320] 0 9 14 11 12 13 10 15 12 14 ...
##  $ G2_por      : num [1:320] 11 11 14 13 12 12 13 16 12 14 ...
##  $ G3_por      : num [1:320] 11 11 14 13 13 13 13 17 13 14 ...
```

- We can note that `Medu` and `Fedu` are education indicator variables. They are ordered but not numeric. We wouldn't want to treat these as numeric! We likely want to make them ordered factors.
- `traveltime` is an indicator variable but a 1 is < 15 minutes, 2 is 15 to 30 minutes, 3 is 30 minutes to 1 hour, and 4 is > 1 hour. This again means we shouldn't treat this as numeric but as some kind of ordered factor (especially since the categories don't have the same length of time in them).
- `studytime` is similar to `traveltime`
- `famrel`, `freetime`, `gout`, `Dalc`, `Walc`, and `health` are Likert scale type data. These can sometimes be treated as numeric. We would need to be careful with those though!

Ok, let's look at the missingness in the data.

```
sum_na <- function(col){
  sum(is.na(col))
}
dat |>
  summarize(across(everything(), sum_na))
```

```
## # A tibble: 1 x 38
##   school   sex   age address famsize Pstatus  Medu  Fedu  Mjob  Fjob reason
##    <int> <int> <int>   <int>   <int>   <int> <int> <int> <int> <int>  <int>
## 1      0     0     0       0       0       0     0     0     0     0      0
## # i 27 more variables: guardian <int>, traveltime <int>, studytime <int>,
## #   failures <int>, schoolsup <int>, famsup <int>, paid_mat <int>,
## #   activities <int>, nursery <int>, higher <int>, internet <int>,
```

```
## #   romantic <int>, famrel <int>, freetime <int>, goout <int>, Dalc <int>,
## #   Walc <int>, health <int>, absences_mat <int>, G1_mat <int>, G2_mat <int>,
## #   G3_mat <int>, paid_por <int>, absences_por <int>, G1_por <int>,
## #   G2_por <int>, G3_por <int>
```

No missing values. Nice!

Categorical variables The following 1-way contingency table shows that the majority of students have internet access available at home:

```
# 1-way contingency table
table("internet at home?" = dat$internet)
```

```
## internet at home?
##  no yes
##  48 272
```

We can add an extra dimension to create a 2-way contingency table. Here, we see that urban students are more likely to have internet access at home versus rural students. Also note that rural students make up a small minority of the total students in these data.

```
# 2-way contingency table
table("internet at home?" = dat$internet,
      "address" = dat$address)
```

```
##                  address
## internet at home?   R   U
##               no   21  27
##               yes  46 226
```

Adding a third dimension will create a 3-way contingency table. As seen below, we've added a variable indicating if students want to pursue higher education. The vast majority of students here do want to pursue higher education. Of the 6 students who do not, all do have internet access at home.

```
# 3-way contingency table
table("internet at home?" = dat$internet,
      "address" = dat$address,
      "want higher ed?" = dat$higher)
```

```
## , , want higher ed? = no
##
##                  address
## internet at home?   R   U
##               no    0   0
##               yes   3   4
##
## , , want higher ed? = yes
##
##                  address
## internet at home?   R   U
##               no   21  27
##               yes  43 222
```

We can also pre-filter a contingency table to get a "slice" of the joint distribution. Here, we subset down to only students who mention the school's reputation as the reason for taking the class. Within this subset, we see that all the rural students have internet access at home versus 8 of 21 urban students.

```
# 2-way contingency table, pre-filtered for condition
fil <- filter(dat, reason == "reputation")
table("internet at home?" = fil$internet,
      "address" = fil$address)
```

```
##                  address
## internet at home?  R  U
##               no   3  9
##               yes 18 55
```

Another approach to arriving at the same results as above is to create a 3-way contingency table first, and then condition the table down to a lower dimension. Here, we condition on the student's reason for choosing the school, and get the same table as above.

```
# 3-way contingency table, conditioned by 1 dimension
tab <- table("internet at home?" = dat$internet,
             "address" = dat$address,
             "why this school?" = dat$reason)
tab[,,"reputation"]
```

```
##                  address
## internet at home?  R  U
##               no   3  9
##               yes 18 55
```

The above examples used the base R `table()` function, but we can create similar contingency tables using functions from `dplyr` (the `tidyverse`). In the table below, we see that rural students are much less likely to use "closeness to home" as the reason for taking the class versus urban students.

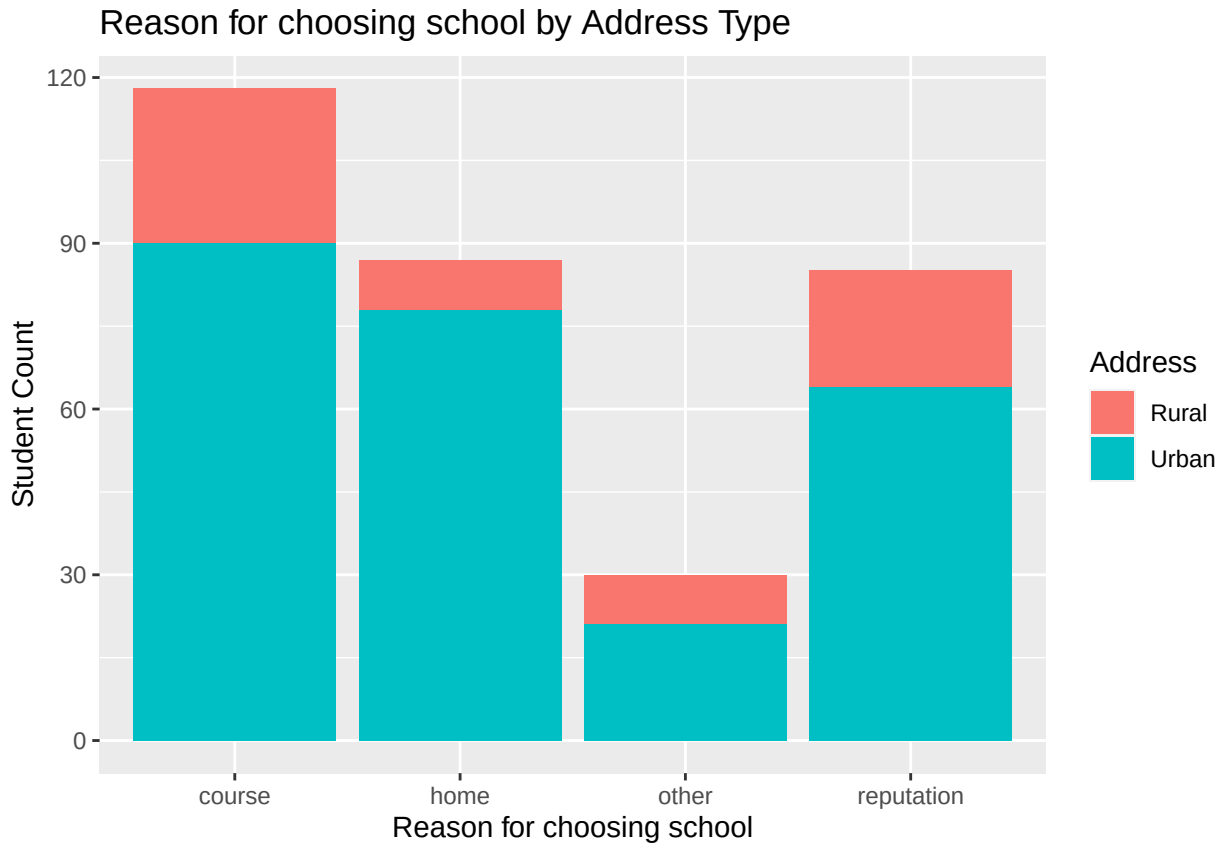
```
# 2-way contingency table using dplyr
dat |>
  group_by(reason, address) |>
  summarize(count = n()) |>
  pivot_wider(names_from = address, values_from = count)
```

```
## # A tibble: 4 x 3
## # Groups:   reason [4]
##   reason         R     U
##   <fct>      <int> <int>
## 1 course        28    90
## 2 home           9    78
## 3 other          9    21
## 4 reputation    21    64
```

Finally, we show the use of bar graphs (both stacked and side-by-side) to visualize the counts of categorical data fields. The first step needed is to calculate the counts by the appropriate groups using a similar approach (`dplyr`-based) as above. Then we create the stacked barplot seen below. The results seen here match those derived from the immediately prior table: rural students are much less likely to use "closeness to home" as the reason for selecting the course. We can also see that "course preference" is the most popular reason given for taking the course, with the "other reasons" category being the smallest.

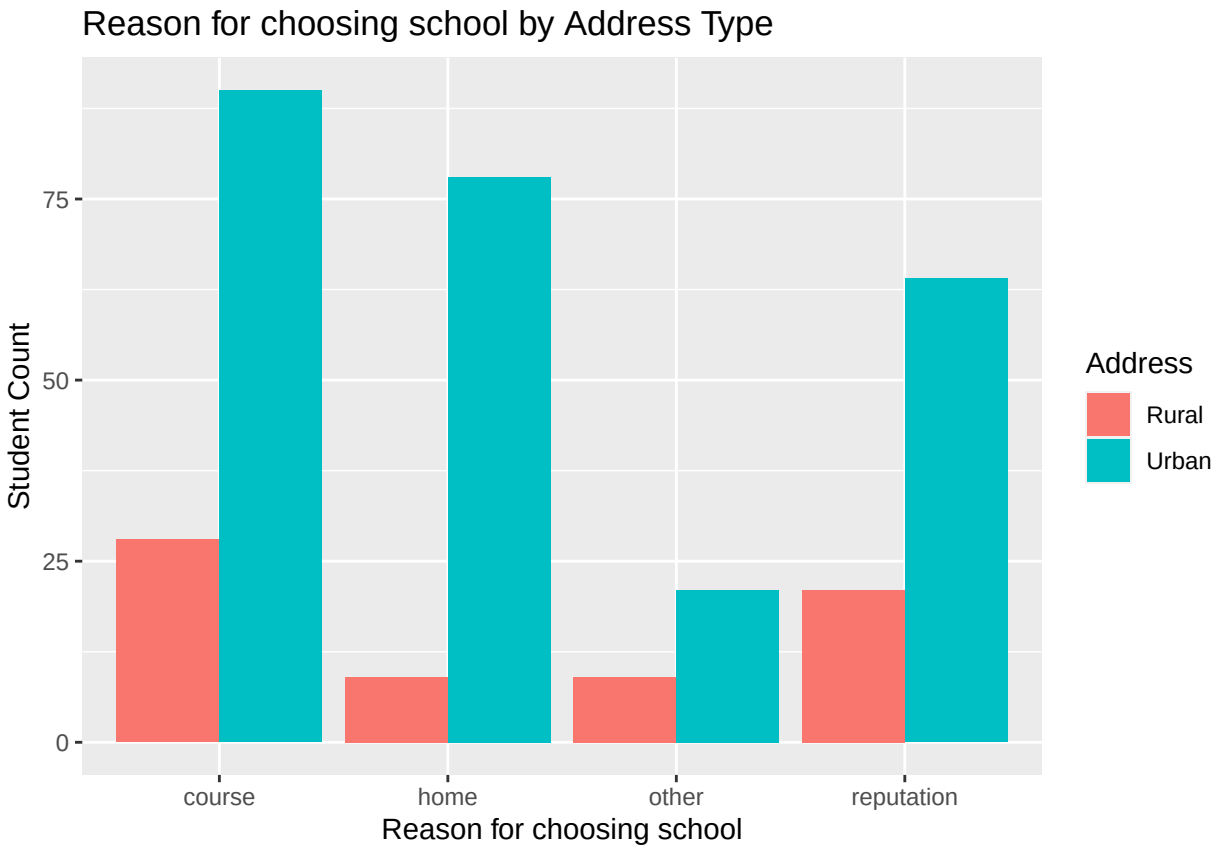
```r
# data counts for plot
dat_barplot <- dat |>
  group_by(reason, address) |>
  summarize(count = n())

# stacked barplot
ggplot(data=dat_barplot, aes(x=reason, y=count, fill=address)) +
  geom_bar(stat="identity") +
  xlab("Reason for choosing school") +
  ylab("Student Count") +
  ggtitle("Reason for choosing school by Address Type") +
  scale_fill_discrete(name="Address", labels=c("Rural", "Urban"))
```



We can also recreate the same visualization using side-by-side bars. The interpretations are the same as above

```r
# side-by-side barplot
ggplot(data=dat_barplot, aes(x=reason, y=count, fill=address)) +
  geom_bar(stat="identity", position="dodge") +
  xlab("Reason for choosing school") +
  ylab("Student Count") +
  ggtitle("Reason for choosing school by Address Type") +
  scale_fill_discrete(name="Address", labels=c("Rural", "Urban"))
```



Numeric variables The code below summarizes selected numeric variables using a variety of approaches and visualizations. First, we calculate the means and standard deviations for all of the various term grades (G1, G2, and G3) for both classes (Math and Portuguese). Then we repeat this while subsetting down to only students who want to pursue higher education. Interesting findings include:

- the mean Portuguese term grades increase slightly over time,
- the standard deviations of the Math term scores increase (widen) over time, and
- mean grades for students wanting to pursue higher education are slightly higher for all grading terms in both classes.

```
# unconditioned
dat |>
  summarise(across(matches("G[0-9]"), list(mean = mean, sd = sd), .names = "{.col}.{.fn}"))
```

```
## # A tibble: 1 x 12
##   G1_mat.mean G1_mat.sd G2_mat.mean G2_mat.sd G3_mat.mean G3_mat.sd G1_por.mean
##         <dbl>     <dbl>       <dbl>     <dbl>       <dbl>     <dbl>       <dbl>
## 1        11.3      3.20        11.2      3.67        11.0      4.30        12.4
## # i 5 more variables: G1_por.sd <dbl>, G2_por.mean <dbl>, G2_por.sd <dbl>,
## #   G3_por.mean <dbl>, G3_por.sd <dbl>
```

```
# again, but subset down to only students who want to attend higher ed
dat |>
  filter(higher == "yes") |>
  summarise(across(matches("G[0-9]"), list(mean = mean, sd = sd), .names = "{.col}.{.fn}"))
```

```
## # A tibble: 1 x 12
##   G1_mat.mean G1_mat.sd G2_mat.mean G2_mat.sd G3_mat.mean G3_mat.sd G1_por.mean
##         <dbl>     <dbl>       <dbl>     <dbl>       <dbl>     <dbl>       <dbl>
## 1        11.3      3.19        11.3      3.63        11.1      4.25        12.5
## # i 5 more variables: G1_por.sd <dbl>, G2_por.mean <dbl>, G2_por.sd <dbl>,
## #   G3_por.mean <dbl>, G3_por.sd <dbl>
```

If we condition the above summary statistics on the address type (rural versus urban), we can see that urban students have higher grades (on average) across all terms/periods and in both classes. Rural students' grades also have greater spread/variance across all grading terms and both classes.

```
# conditioned on address
dat |>
  group_by(address) |>
  summarise(across(matches("G[0-9]"), list(mean = mean, sd = sd), .names = "{.col}.{.fn}"))
```

```
## # A tibble: 2 x 13
##   address G1_mat.mean G1_mat.sd G2_mat.mean G2_mat.sd G3_mat.mean G3_mat.sd
##   <fct>         <dbl>     <dbl>       <dbl>     <dbl>       <dbl>     <dbl>
## 1 R              10.7      3.35        10.1      4.02        9.90      4.59
## 2 U              11.5      3.15        11.5      3.53       11.3       4.18
## # i 6 more variables: G1_por.mean <dbl>, G1_por.sd <dbl>, G2_por.mean <dbl>,
## #   G2_por.sd <dbl>, G3_por.mean <dbl>, G3_por.sd <dbl>
```

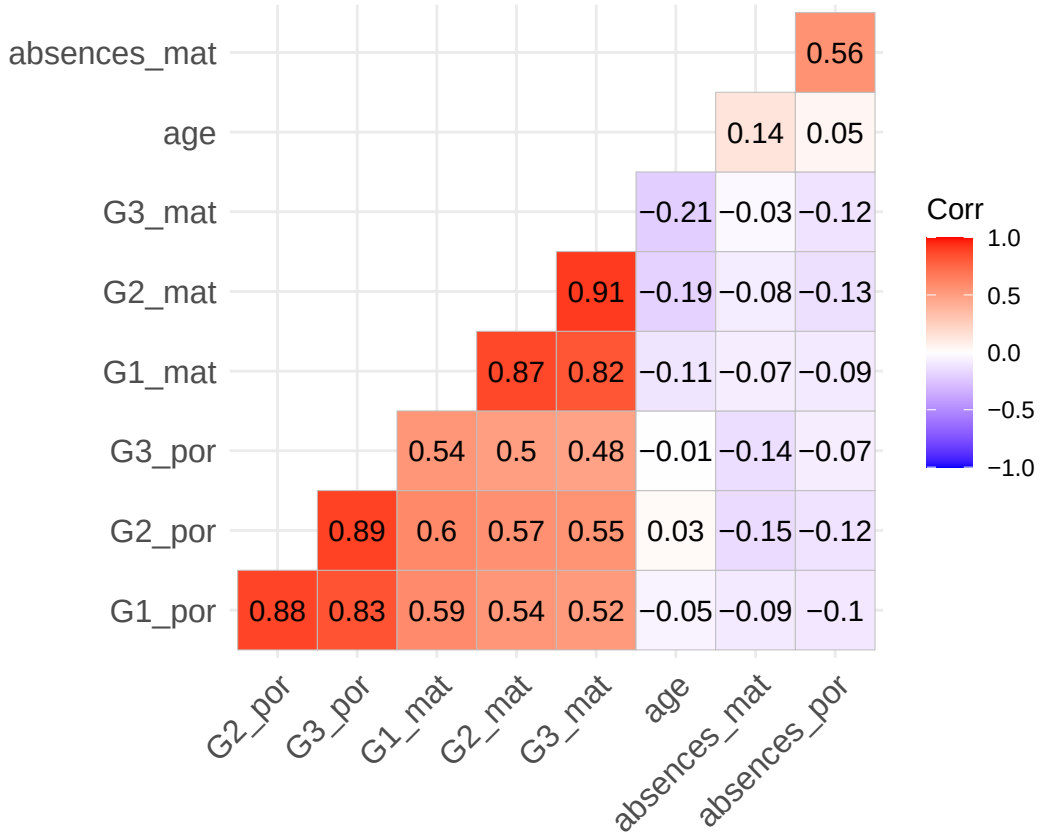
We can simultaneously condition on two categorical variables (here, address and reason for taking the course). Interestingly, we find that rural students taking the Math class because of its "closeness to home" tend to have the lowest mean grades, while urban students giving the same reason have among the highest mean grades, and this persists across all grading periods.

```
dat |>
  group_by(address, reason) |>
  summarise(across(c(G1_mat, G2_mat, G3_mat),
                   list(mean = mean, sd = sd), .names = "{.col}.{.fn}"))
```

```
## # A tibble: 8 x 8
## # Groups:   address [2]
##   address reason     G1_mat.mean G1_mat.sd G2_mat.mean G2_mat.sd G3_mat.mean
##   <fct>   <fct>            <dbl>     <dbl>       <dbl>     <dbl>       <dbl>
## 1 R       course            11.1      2.99        10.0      4.12        9.57
## 2 R       home               9.22     2.49         9.44     2.70        9.89
## 3 R       other             10.1      2.85         9.56     2.79        9.22
## 4 R       reputation        11.0      4.22        10.7      4.85       10.6
## 5 U       course            10.9      3.35        10.8      3.96       10.6
## 6 U       home              11.5      3.19        11.5      3.41       11.2
## 7 U       other             11.5      3.27        12.3      3.34       12.3
## 8 U       reputation        12.1      2.69        12.1      2.92       12.2
## # i 1 more variable: G3_mat.sd <dbl>
```

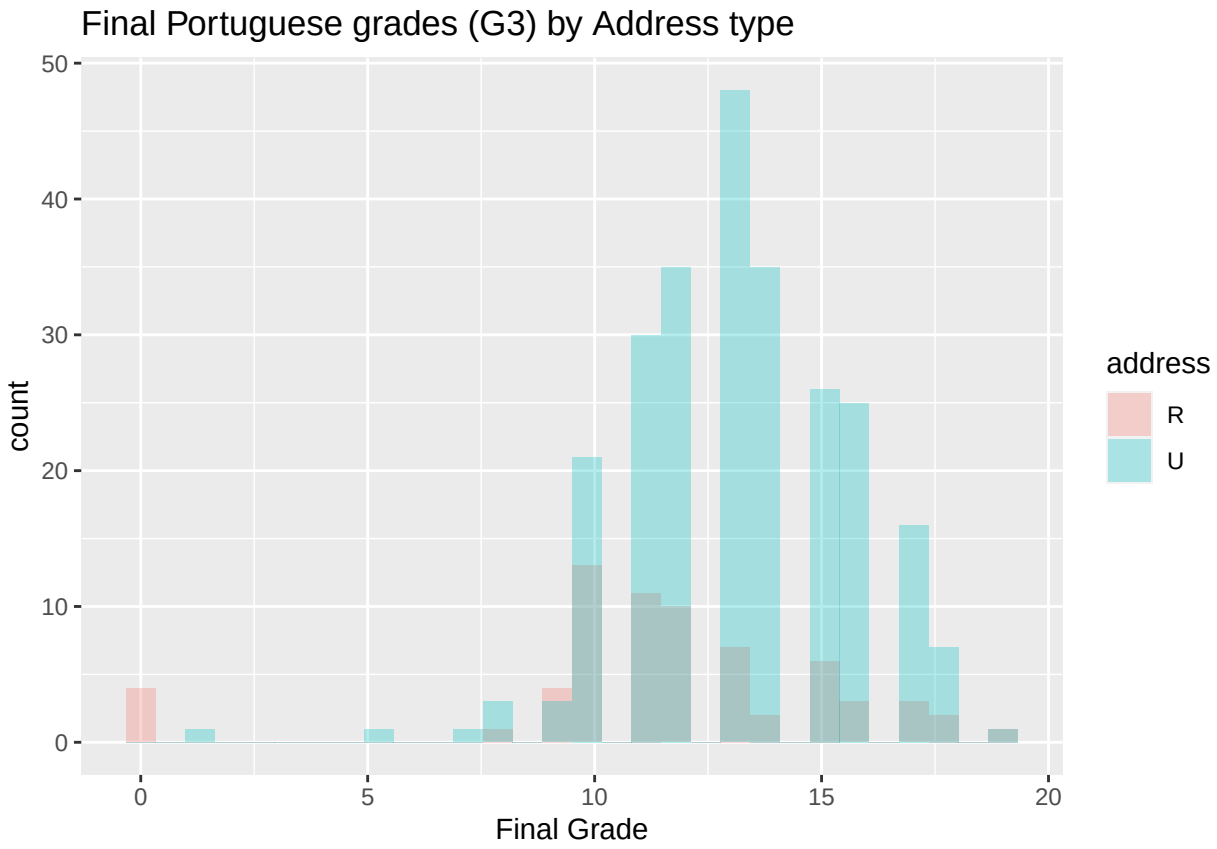
We can also create a correlation matrix to examine associations between all the numeric variables in the dataset. The highest observed positive correlations are between the various term grades. These are especially high between the first and second term grades for a given class, and grades *between* courses are also positively correlated to a lesser degree. There is also a high correlation between the number of absences in the two classes (as you would expect). Finally, there is a very slight negative correlation between age and the various grades in both classes.

```r
dat |>
  select(age, ends_with("_mat"), ends_with("_por")) |>
  select(-paid_mat, - paid_por) |>
  cor() |>
  ggcorrplot(hc.order = TRUE, type = "lower", lab = TRUE)
```

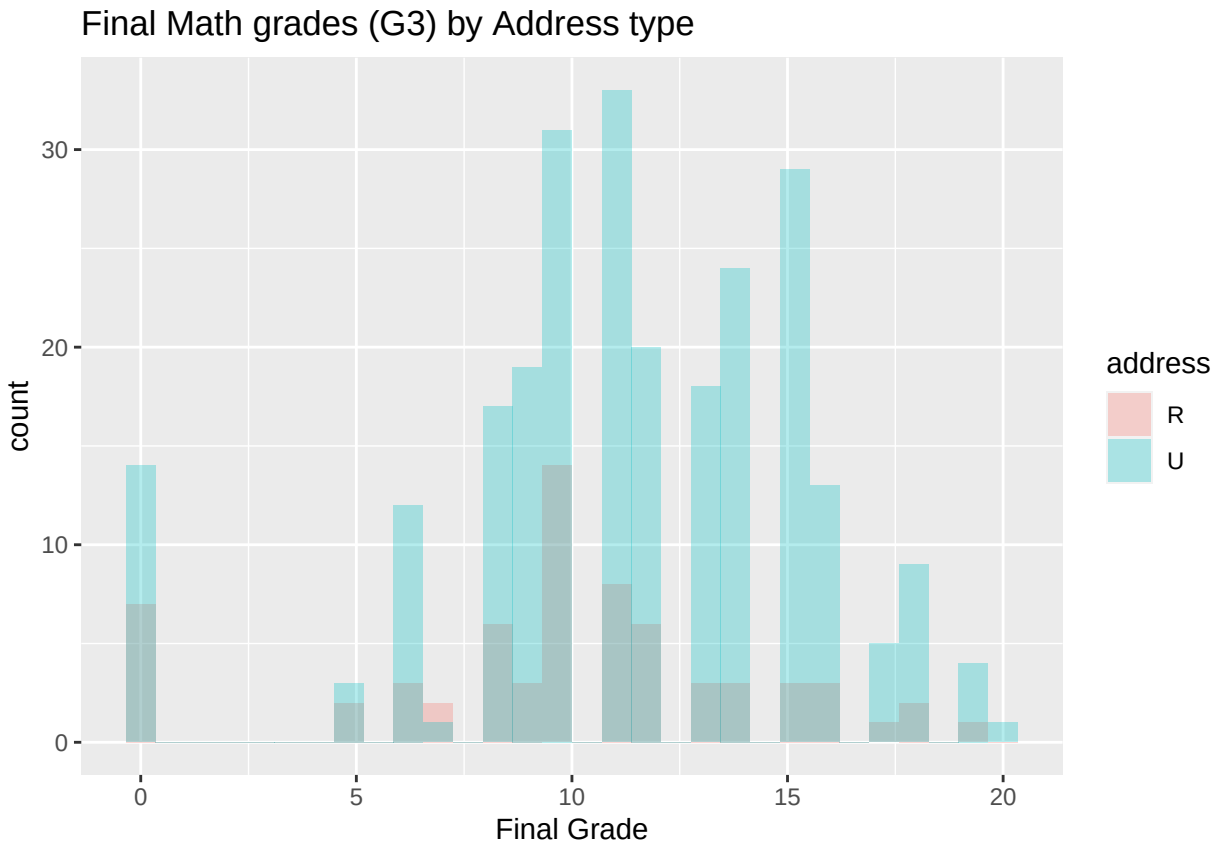


The following series of plots visualize the relationship between the final (G3) grades in the Math and Portuguese classes across the levels of address type (rural versus urban). We can explore these joint distributions using histograms, boxplots, and kernel density plots. The findings from these graphs are similar: urban students tend to have higher final grades in both classes. We can also see there are a handful of final grades in both classes = 0.

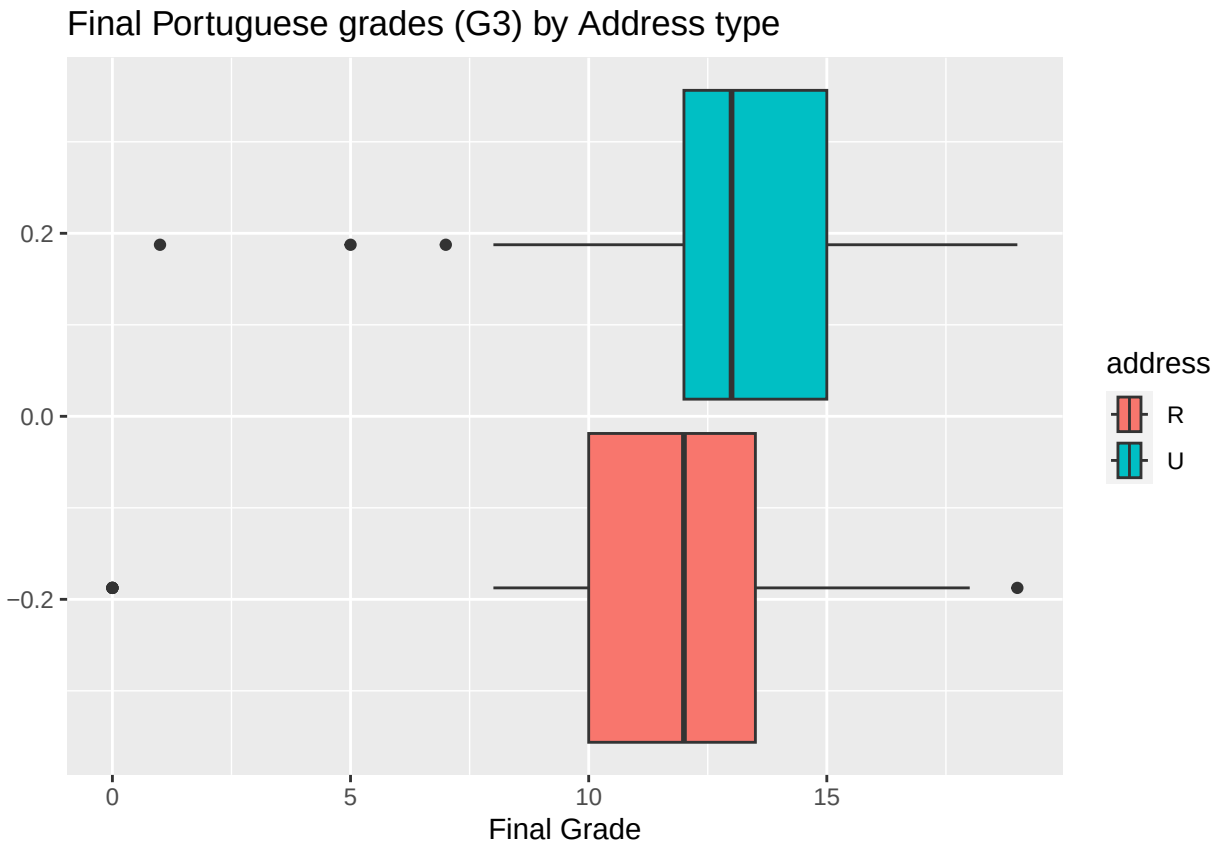
```r
# histograms for G3_mat and G3_por across levels of address
ggplot(dat, aes(x=G3_por, fill=address)) +
  geom_histogram(alpha=0.3, position = 'identity') +
  ggtitle("Final Portuguese grades (G3) by Address type") +
  xlab("Final Grade")
```



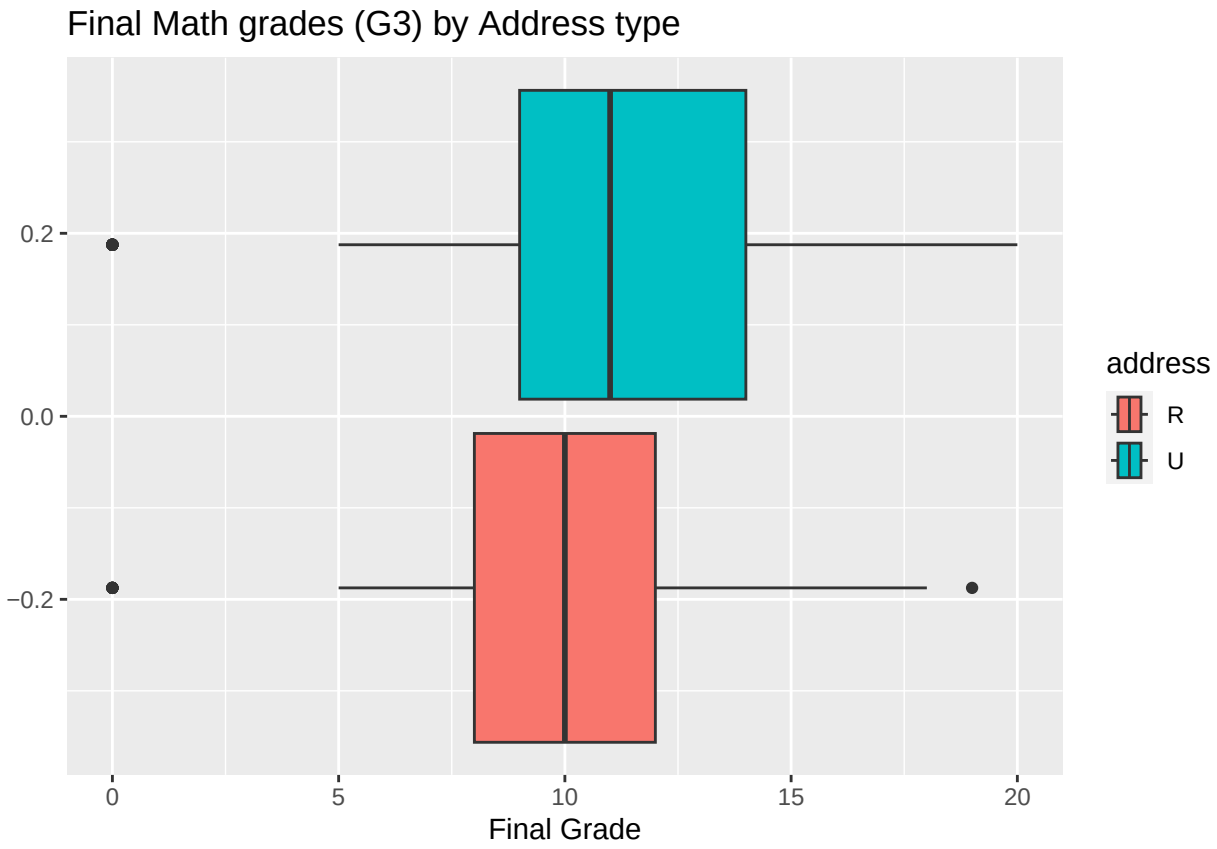

```
ggplot(dat, aes(x=G3_mat, fill=address)) +
  geom_histogram(alpha=0.3, position = 'identity') +
  ggtitle("Final Math grades (G3) by Address type") +
  xlab("Final Grade")
```



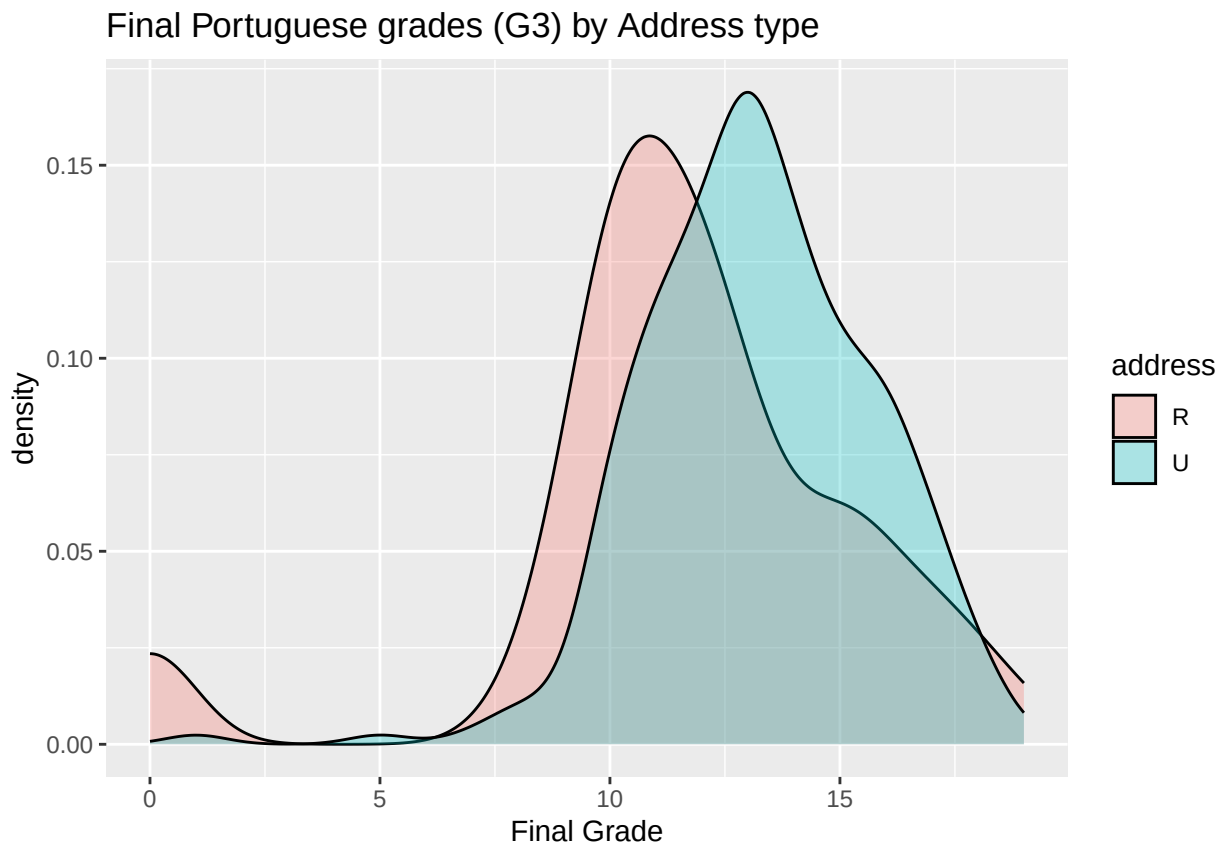

```
# boxplots for G3_mat and G3_por across levels of address
ggplot(dat, aes(x=G3_por, fill=address)) +
  geom_boxplot() +
  ggtitle("Final Portuguese grades (G3) by Address type") +
  xlab("Final Grade")
```



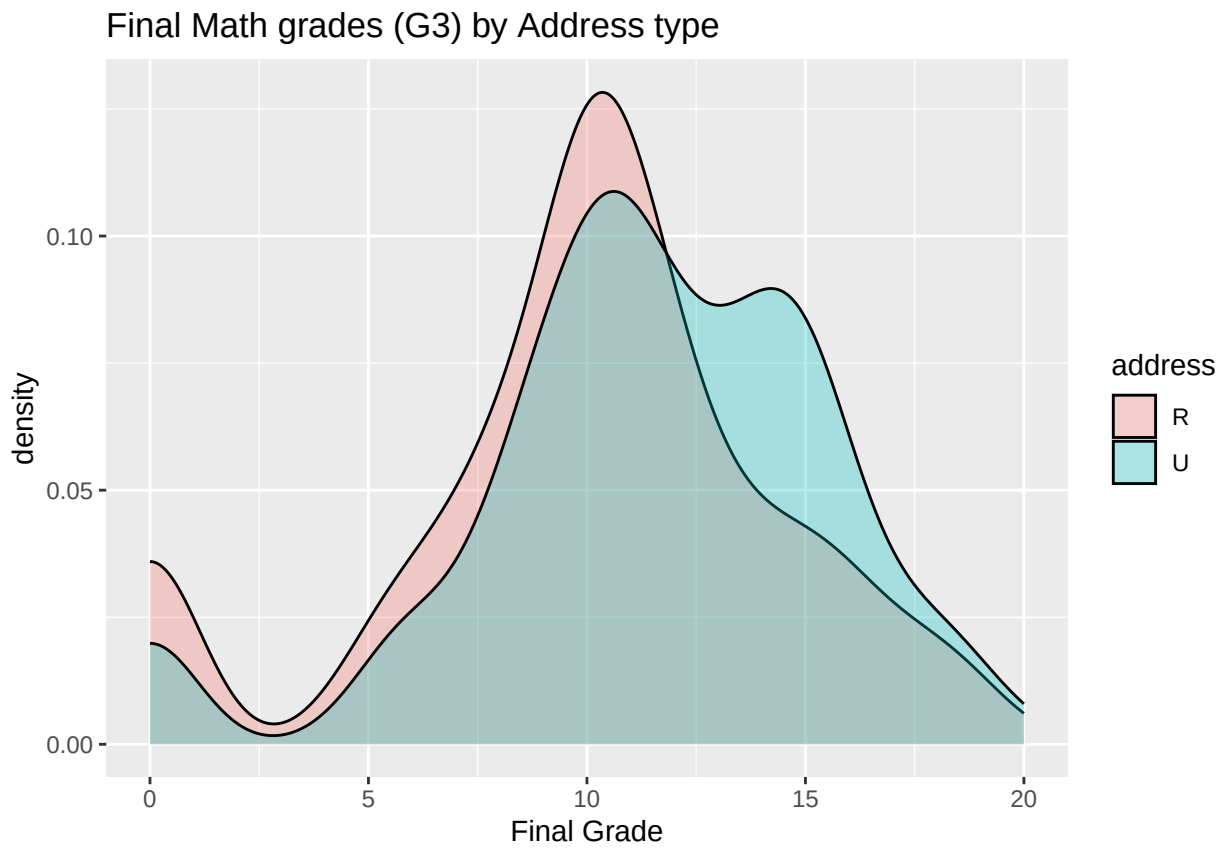

```
ggplot(dat, aes(x=G3_mat, fill=address)) +
  geom_boxplot() +
  ggtitle("Final Math grades (G3) by Address type") +
  xlab("Final Grade")
```




```
# KDplots for G3_mat and G3_por across levels of address
ggplot(dat, aes(x=G3_por, fill=address, cut=address)) +
  geom_density(alpha=0.3) +
  ggtitle("Final Portuguese grades (G3) by Address type") +
  xlab("Final Grade")
```

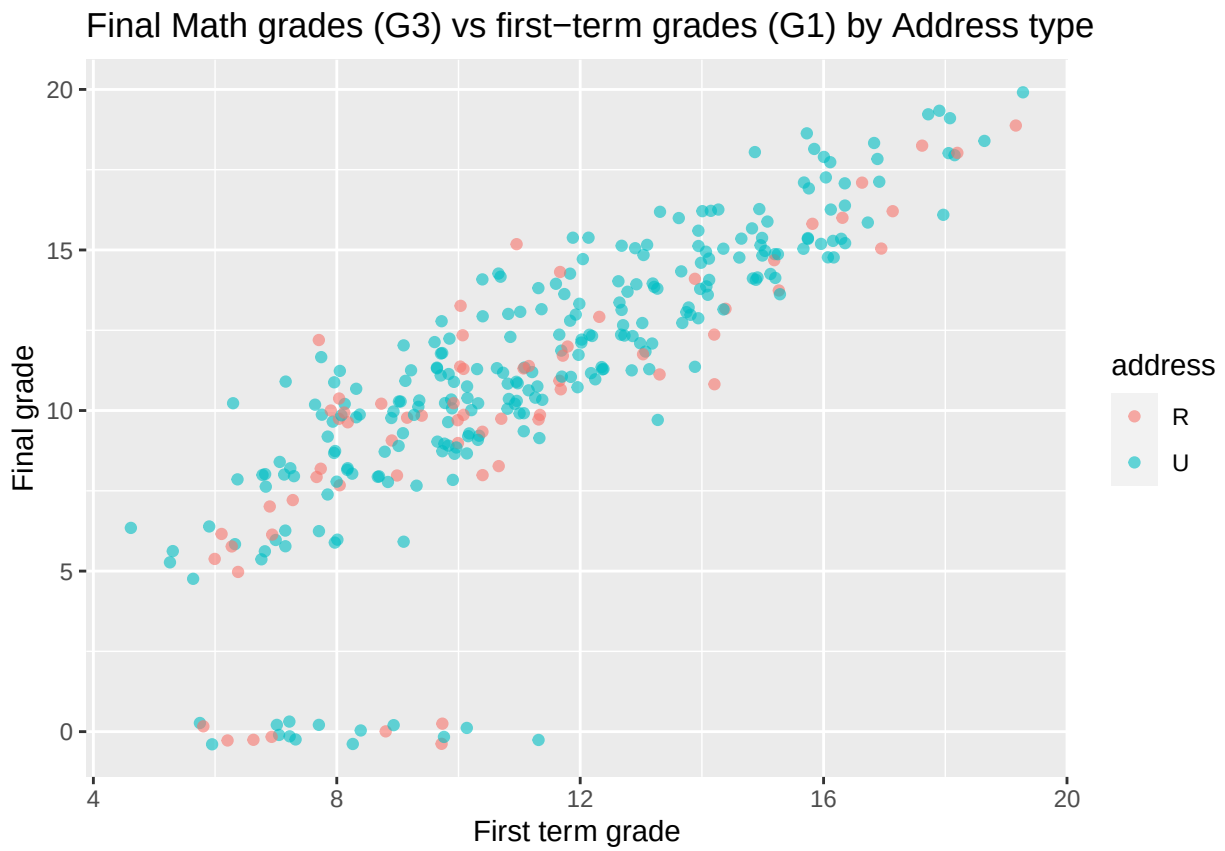



```
ggplot(dat, aes(x=G3_mat, fill=address)) +
  geom_density(alpha=0.3) +
  ggtitle("Final Math grades (G3) by Address type") +
  xlab("Final Grade")
```

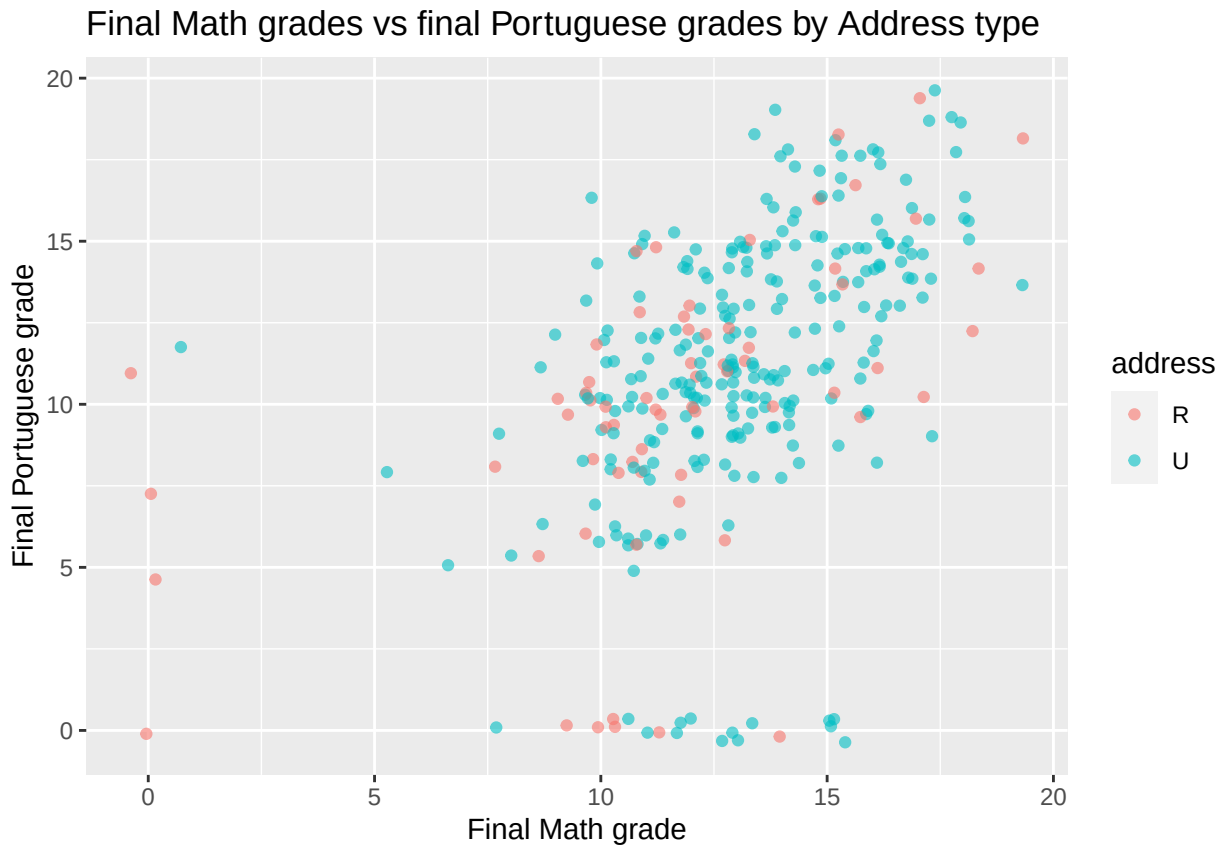


Using scatter plots, we examine the relationship between students' first and final Math grades, followed by the relationship between their final Portuguese and Math scores. Using different colored markers, we can also add another visual dimension to represent the students' address types. In both plots, we see a generally positive association between the variables on the x- and y-axes: between both first and final Math grades, as well as between final Math and Portuguese grades (with the former association appearing "tighter". Though we see no clear difference in these associations across address types, we do see a handful of students with 0 grades on both classes. Also note that all of the final Math grades = 0 are from rural students.

```
ggplot(dat, aes(x=G1_mat, y=G3_mat, color=address)) +
  geom_jitter(alpha=0.6) +
  ggtitle("Final Math grades (G3) vs first-term grades (G1) by Address type") +
  xlab("First term grade") +
  ylab("Final grade")
```

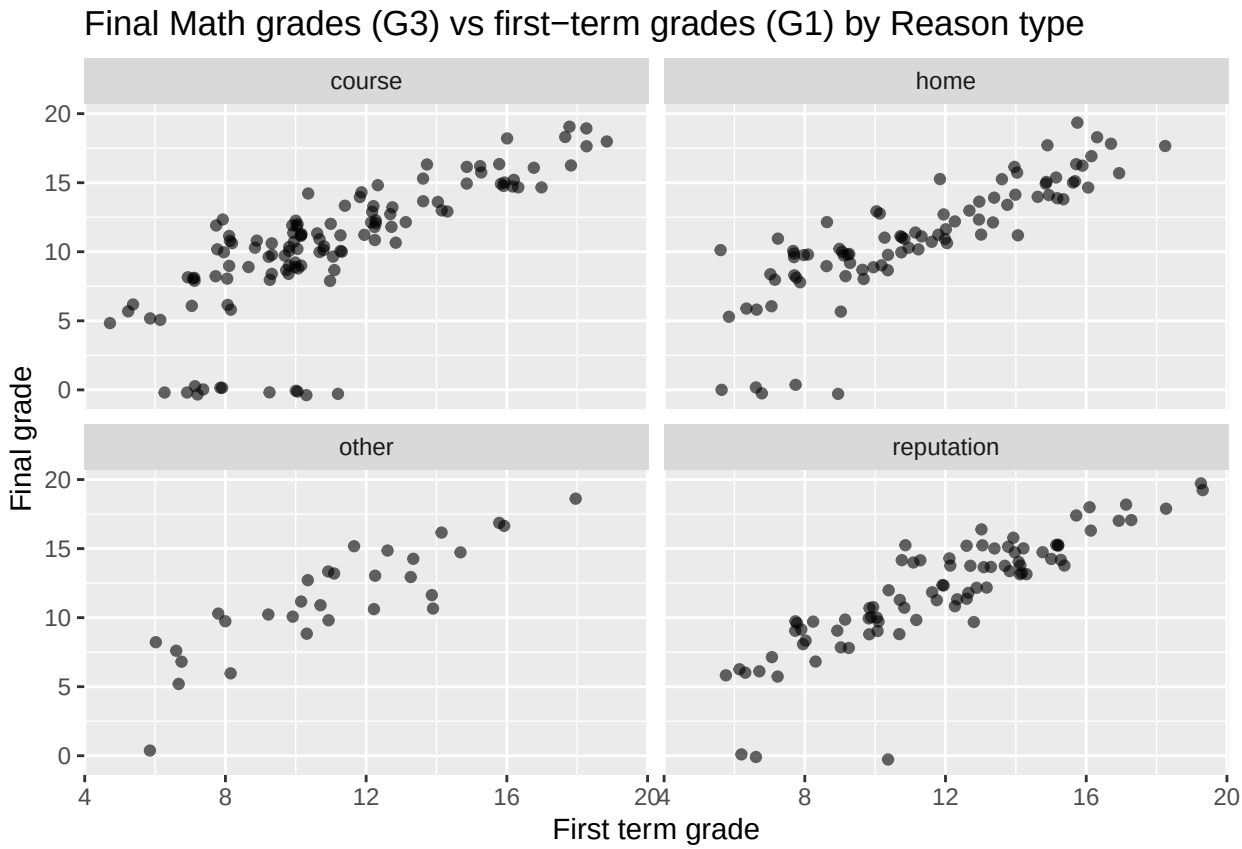



```
ggplot(dat, aes(x=G3_por, y=G3_mat, color=address)) +
  geom_jitter(alpha=0.6) +
  ggtitle("Final Math grades vs final Portuguese grades by Address type") +
  xlab("Final Math grade") +
  ylab("Final Portuguese grade")
```

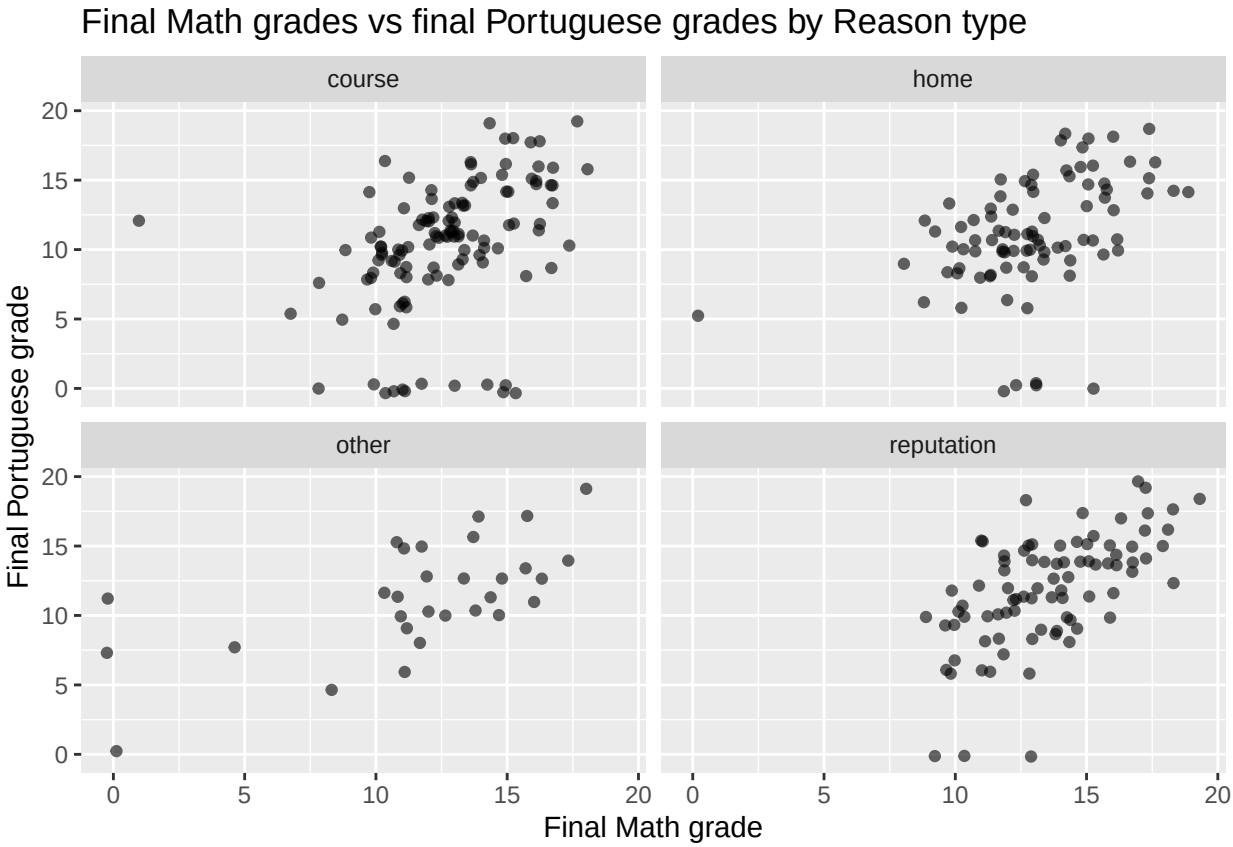


Now we remake the above plots, but use faceting instead of marker color as an additional dimension to show variation from a third variable. In these plots, we use the student reason type as the faceting dimension. I see no clear trends in the Math G1 versus Math G3 scores across reason types, but it does appear that most of the 0 grades in both sets of plots are coming from students who gave "course preference" as the reason for taking the class.

```
ggplot(dat, aes(x=G1_mat, y=G3_mat)) +
  geom_jitter(alpha=0.6) +
  ggtitle("Final Math grades (G3) vs first-term grades (G1) by Reason type") +
  xlab("First term grade") +
  ylab("Final grade") +
  facet_wrap(~reason)
```

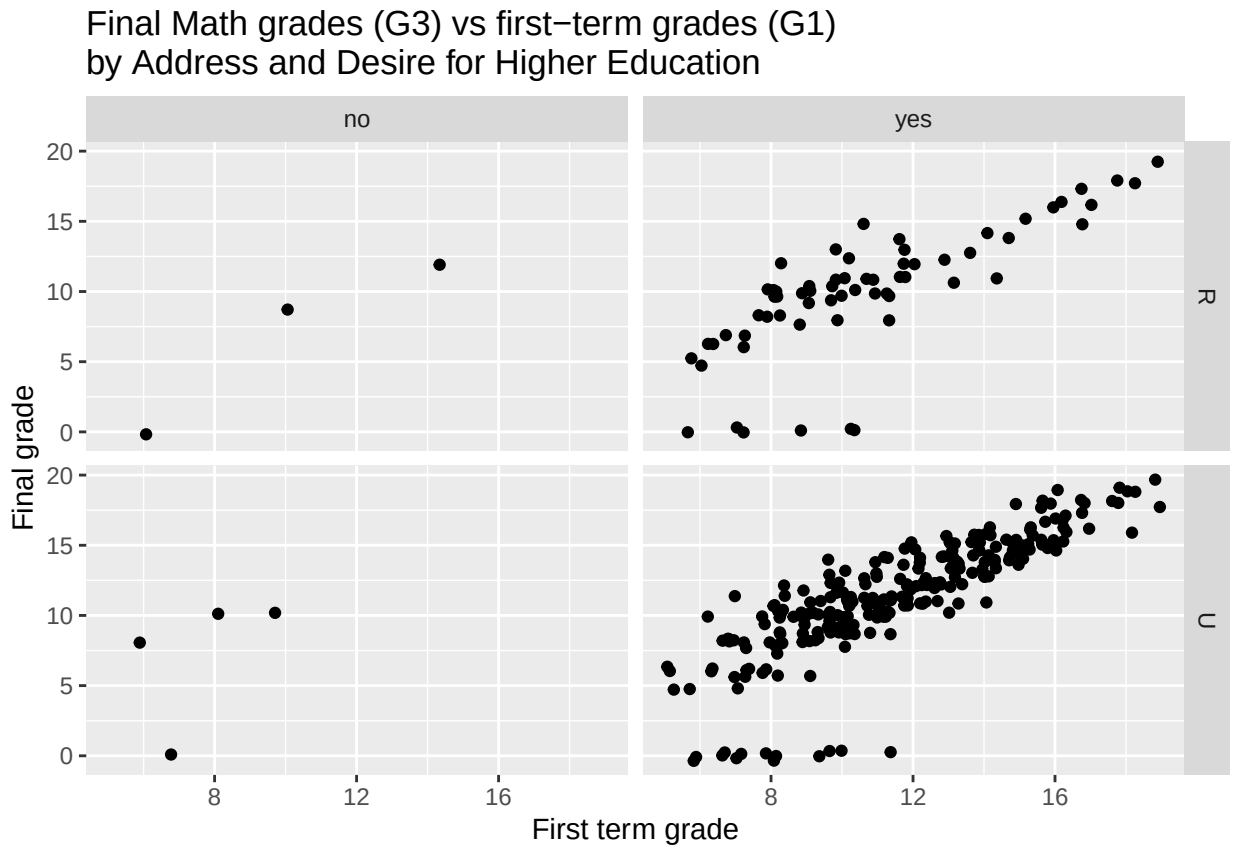



```
ggplot(dat, aes(x=G3_por, y=G3_mat)) +
  geom_jitter(alpha=0.6) +
  ggtitle("Final Math grades vs final Portuguese grades by Reason type") +
  xlab("Final Math grade") +
  ylab("Final Portuguese grade") +
  facet_wrap(~reason)
```

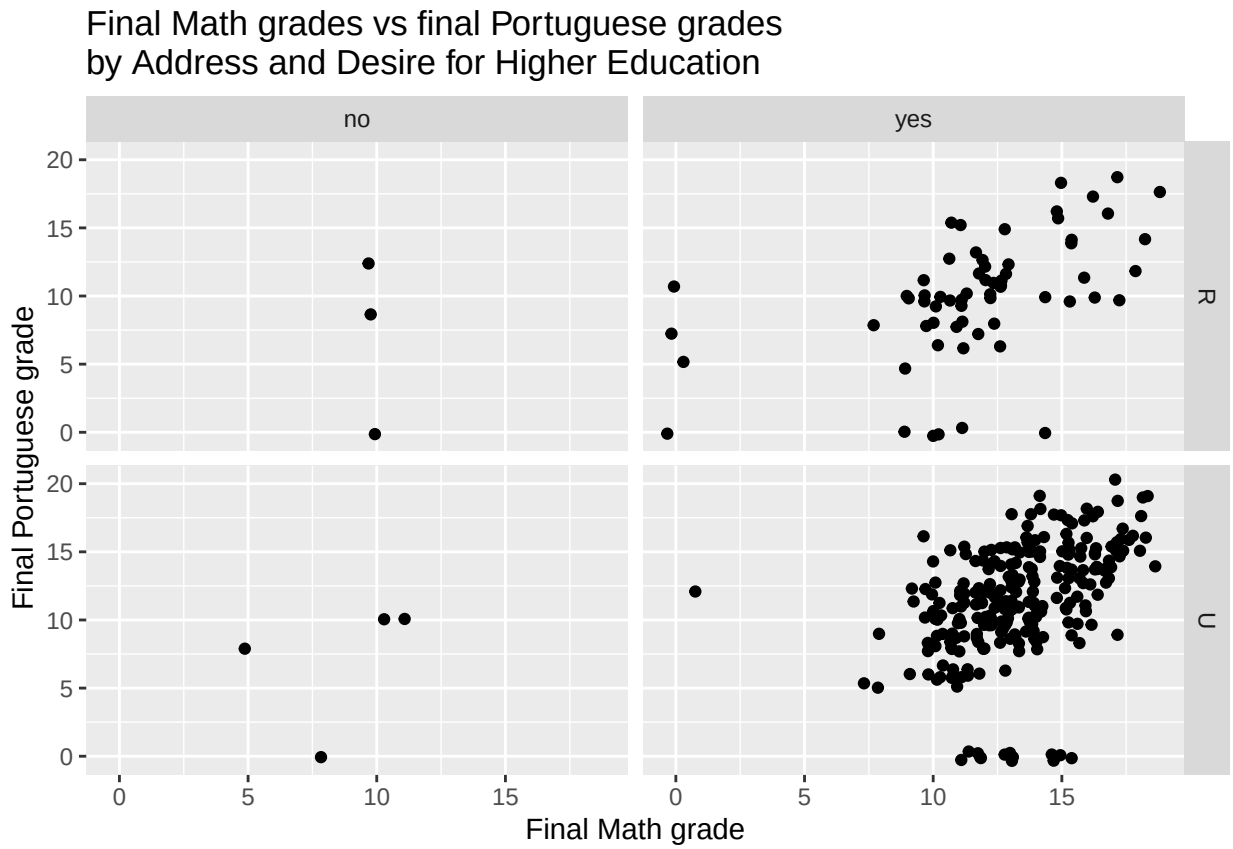


Finally, we show an example of using two faceting variables to create a grid of plots. We use the same y- and x-axis pairs as above, but now use address tyoe and the desire to pursue higher education to create the facet grid. Within each facet, the patterns are similar to those previously noted (particularly the tighter relationship within a set of course grades versus between courses), but now we can see the *very few* observations coming from students having no desire for higher education

```
ggplot(dat, aes(x=G1_mat, y=G3_mat)) +
  geom_jitter() +
  ggtitle("Final Math grades (G3) vs first-term grades (G1)\nby Address and Desire for Higher Education
  xlab("First term grade") +
  ylab("Final grade") +
  facet_grid(address ~ higher)
```




```
ggplot(dat, aes(x=G3_por, y=G3_mat)) +
  geom_jitter() +
  ggtitle("Final Math grades vs final Portuguese grades\nby Address and Desire for Higher Education") +
  xlab("Final Math grade") +
  ylab("Final Portuguese grade") +
  facet_grid(address ~ higher)
```


Final Math grades vs final Portuguese grades
by Address and Desire for Higher Education
no
yes
R
U
Final Portuguese grade
Final Math grade
0
5
10
15
20