

Association and Correlation

Matt Lawrence

September 30, 2019

Getting Started

Today we will be using data from Chetty et al's 2014 paper "Where Is The Land Of Opportunity?". The `commuting_zones.csv` file on Canvas comes from the Opportunity Insights Project's website which can be accessed [here](#). (Note how to include links in Markdown: the link title should be in brackets followed by the link destination in parentheses. Only the link title will appear in the knitted file, but it will be clickable.)

Load the data as a data frame called `cz` and load tidyverse.

Finding Correlation Coefficients

Let's start with the correlation between income segregation and the proportion of workers who commute 15 minutes or less. What is a hypothesis for how these two variables could be related? Would you expect a positive or negative correlation?

The correlation coefficient is calculated as the covariance of x and y divided by the product of the standard deviations of x and y . In mathematical notation, we write:

$$cor_{x,y} = \frac{cov_{x,y}}{s_x s_y}$$

(If you hover over the equation above, you should see it converted to a more readable format. We won't learn how to write Tex equations in this class, but it could be good to know that R Markdown can handle them.)

We already know how to find the standard deviation using the `sd()` function. To find the covariance we use the `cov()` function and separate the two variables by a comma (just like a cross-tabulation). Let's plug all these values into the equation using `commute15min` as our X variable and `income_seg` as our Y variable.

```
cov(cz$commute15min, cz$income_seg) /  
  (sd(cz$commute15min) * sd(cz$income_seg))
```

```
## [1] -0.6083312
```

How would you interpret this correlation?

Fortunately, R can calculate the correlation for us using the `cor()` function. Just like `cov()` or cross-tabulations, we separate both variables with a comma.

```
cor(cz$commute15min, cz$income_seg)
```

```
## [1] -0.6083312
```

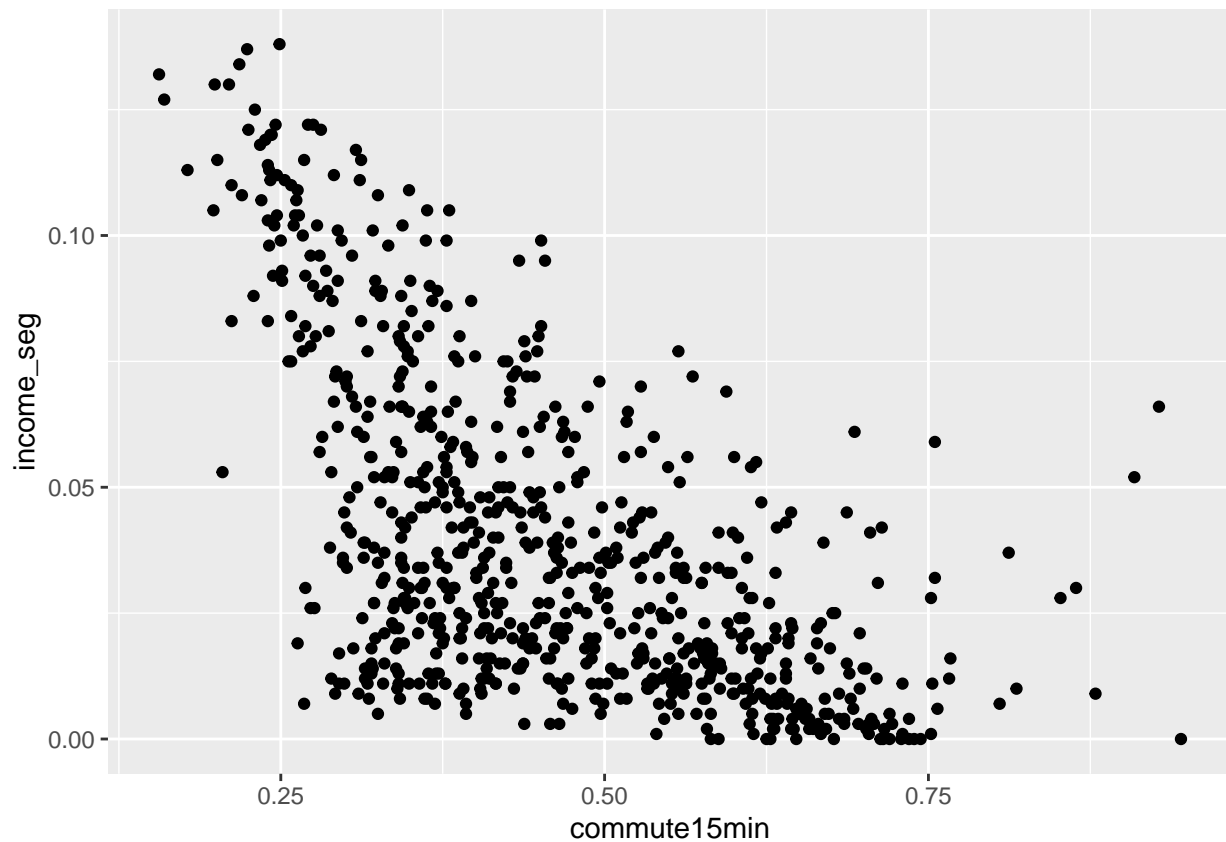
You should get the same value as we calculated earlier.

Visualizing Correlations With Scatterplots

We can also create a scatterplot showing how the distributions of both variables tend to move together. Set up everything in ggplot using the regular x and y aesthetics. For a scatterplot, the plot type is `geom_point()`.

```
commute_incomeseg_scatter <- ggplot(cz,
  aes(x = commute15min, y = income_seg))

commute_incomeseg_scatter + geom_point()
```



What would your hypothesis be for how commuting zones' median incomes and employment rates are associated? Find the correlation for median household income (`hh_income`) and `labor_force_participation_rate` using the `cor()` function, and create a scatterplot showing how the two variables are associated.

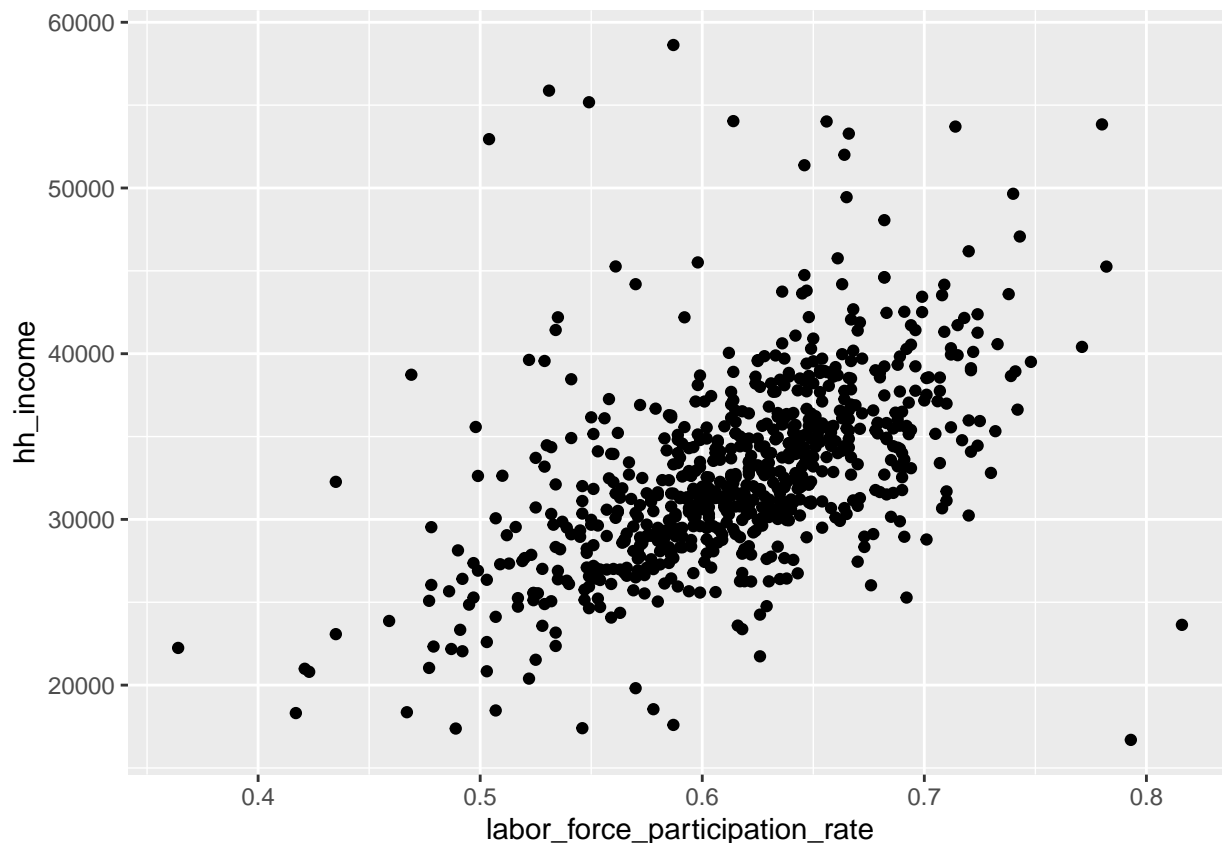
REPLACE THIS LINE WITH YOUR CODE

```
cor(cz$hh_income, cz$labor_force_participation_rate)
```

```
## [1] 0.5226587
```

```
income_laborforce_scatter <- ggplot(cz, aes(
  x = labor_force_participation_rate,
  y = hh_income))
```

```
income_laborforce_scatter + geom_point()
```



Let's look at one more association. What is your hypothesis for how `commute15min` and `mobility` would be related? Find the correlation coefficient for these two variables using the `cor()` function.

REPLACE THIS LINE WITH YOUR CODE

```
cor(cz$commute15min, cz$mobility)
```

```
## [1] NA
```

Uh oh. It looks like there is an error somewhere. To investigate, get a summary of all the variables in the dataframe using `summary()` and the data frame name:

```
summary(cz)
```

```
##      cz_id      cz_name      state      population_2000
##  Min.   : 100    Columbus   : 5    TX       : 64    Min.   : 1193
## 1st Qu.:12701    Jackson    : 4    KS       : 32    1st Qu.: 38384
## Median :26106    Springfield: 4    GA       : 28    Median : 103842
## Mean   :22444    Burlington : 3    MO       : 24    Mean   : 379787
## 3rd Qu.:31301    Center      : 3    SD       : 24    3rd Qu.: 289849
## Max.   :39400    Charleston : 3    MN       : 23    Max.   :16393360
##      (Other) :719    (Other):546
##      mobility      urban      frac_black      racial_seg
##  Min.   :26.70    Min.   :0.0000    Min.   :0.00000    Min.   :0.0000
## 1st Qu.:39.90    1st Qu.:0.0000    1st Qu.:0.00400    1st Qu.:0.0560
## Median :43.30    Median :0.0000    Median :0.02200    Median :0.1070
## Mean   :43.94    Mean   :0.4386    Mean   :0.07781    Mean   :0.1298
```

```

## 3rd Qu.:47.10 3rd Qu.:1.0000 3rd Qu.:0.08200 3rd Qu.:0.1810
## Max. :64.00 Max. :1.0000 Max. :0.65800 Max. :0.5540
## NA's :32
## income_seg poverty_seg affluence_seg commute15min
## Min. :0.00000 Min. :0.00000 Min. :0.00000 Min. :0.1560
## 1st Qu.:0.01400 1st Qu.:0.01300 1st Qu.:0.01300 1st Qu.:0.3450
## Median :0.03100 Median :0.02800 Median :0.03200 Median :0.4360
## Mean :0.03952 Mean :0.03626 Mean :0.04162 Mean :0.4572
## 3rd Qu.:0.05700 3rd Qu.:0.05400 3rd Qu.:0.06000 3rd Qu.:0.5630
## Max. :0.13800 Max. :0.12900 Max. :0.15400 Max. :0.9450
##
## hh_income gini top1pc_share local_tax_rate
## Min. :16696 Min. :0.2020 Min. : 2.673 Min. :0.00800
## 1st Qu.:29327 1st Qu.:0.3480 1st Qu.: 8.005 1st Qu.:0.01700
## Median :32372 Median :0.3980 Median :10.119 Median :0.02200
## Mean :32870 Mean :0.4055 Mean :10.842 Mean :0.02359
## 3rd Qu.:35816 3rd Qu.:0.4570 3rd Qu.:12.545 3rd Qu.:0.02700
## Max. :58628 Max. :0.8470 Max. :64.788 Max. :0.08200
## NA's :32 NA's :1
## local_govt_expenditures school_expenditures_per_student
## Min. : 952 Min. : 3.920
## 1st Qu.: 1722 1st Qu.: 5.168
## Median : 2112 Median : 5.897
## Mean : 2309 Mean : 6.037
## 3rd Qu.: 2638 3rd Qu.: 6.627
## Max. :13621 Max. :11.906
## NA's :2 NA's :10
## test_score_percentile_adj hs_dropout_rate_adj number_of_colleges
## Min. : -32.78500 Min. : -0.04300 Min. :0.00100
## 1st Qu.: -4.29300 1st Qu.: -0.01500 1st Qu.:0.01200
## Median : 0.74100 Median : -0.00400 Median :0.01700
## Mean : 0.00001 Mean : -0.00001 Mean :0.02311
## 3rd Qu.: 5.55400 3rd Qu.: 0.01100 3rd Qu.:0.02600
## Max. : 20.07100 Max. : 0.10900 Max. :0.24300
## NA's :36 NA's :148 NA's :157
## college_grad_rate_adj labor_force_participation_rate
## Min. : -0.35000 Min. :0.364
## 1st Qu.: -0.09700 1st Qu.:0.581
## Median : -0.01600 Median :0.619
## Mean : -0.00001 Mean :0.616
## 3rd Qu.: 0.08300 3rd Qu.:0.653
## Max. : 0.52800 Max. :0.816
## NA's :160
## manufacturing_employment_share migration_inflow migration_outflow
## Min. :0.0020 Min. :0.00000 Min. :0.00000
## 1st Qu.:0.0760 1st Qu.:0.01000 1st Qu.:0.01200
## Median :0.1330 Median :0.01400 Median :0.01600
## Mean :0.1404 Mean :0.01653 Mean :0.01683
## 3rd Qu.:0.1990 3rd Qu.:0.02100 3rd Qu.:0.02100
## Max. :0.4490 Max. :0.07700 Max. :0.05200
## NA's :17 NA's :17
## frac_foreign_born social_capital_index frac_religion
## Min. :0.00000 Min. : -3.1990 Min. :0.1100
## 1st Qu.:0.01200 1st Qu.: -0.7655 1st Qu.:0.4250

```

```
## Median :0.02400 Median : 0.0640 Median :0.5250
## Mean :0.04117 Mean : 0.1717 Mean :0.5456
## 3rd Qu.:0.04600 3rd Qu.: 0.9653 3rd Qu.:0.6430
## Max. :0.39700 Max. : 7.3050 Max. :1.3080
## NA's :19
## violent_crime_rate frac_children_single_mothers frac_adults_divorced
## Min. :0.000000 Min. :0.0820 Min. :0.04000
## 1st Qu.:0.001000 1st Qu.:0.1710 1st Qu.:0.08500
## Median :0.001000 Median :0.1960 Median :0.09800
## Mean :0.001594 Mean :0.2017 Mean :0.09666
## 3rd Qu.:0.002000 3rd Qu.:0.2260 3rd Qu.:0.10900
## Max. :0.028000 Max. :0.4340 Max. :0.19000
## NA's :27
## frac_adults_married income_growth_06_10 drop_this_column
## Min. :0.3730 Min. : -0.118000 Mode:logical
## 1st Qu.:0.5450 1st Qu.: -0.008000 NA's:741
## Median :0.5800 Median : -0.002000
## Mean :0.5745 Mean : -0.001669
## 3rd Qu.:0.6070 3rd Qu.: 0.004000
## Max. :0.6950 Max. : 0.046000
##
```

Dealing With Missing Values

There are 32 observations where the value for mobility is “NA”. That is R’s way of telling us the values are missing, or “not available”. Most datasets will have some missing values, so we need ways to deal with them. For correlations, the way to tell R we only want to use cases without any missing values is to add the `use=complete` option:

```
cor(cz$commute15min, cz$mobility, use="complete")
```

```
## [1] 0.6048691
```

For many other functions, use `na.rm = TRUE` to remove all values with an NA. For example:

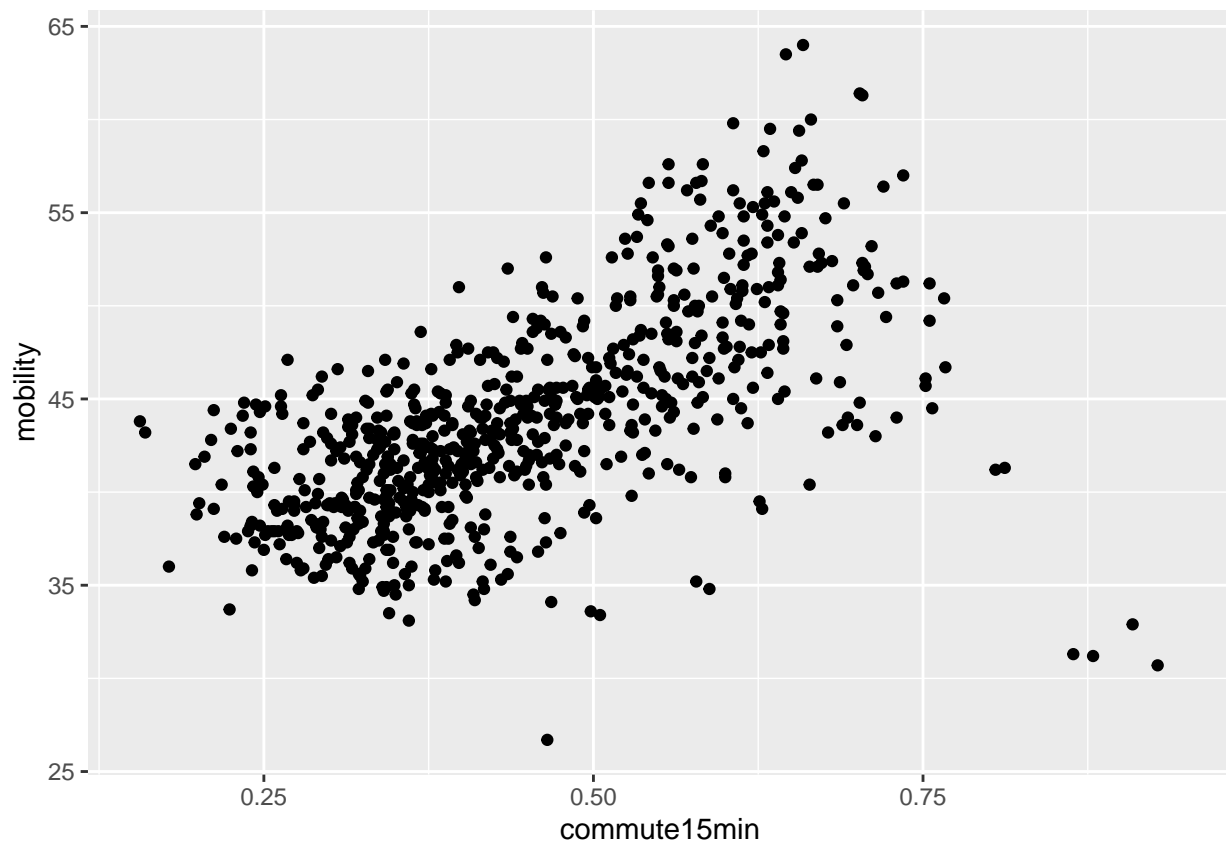
```
mean(cz$mobility, na.rm = TRUE)
```

```
## [1] 43.94344
```

Fortunately, ggplot2 knows to only use complete cases so we do not have to adjust our code to get a scatterplot:

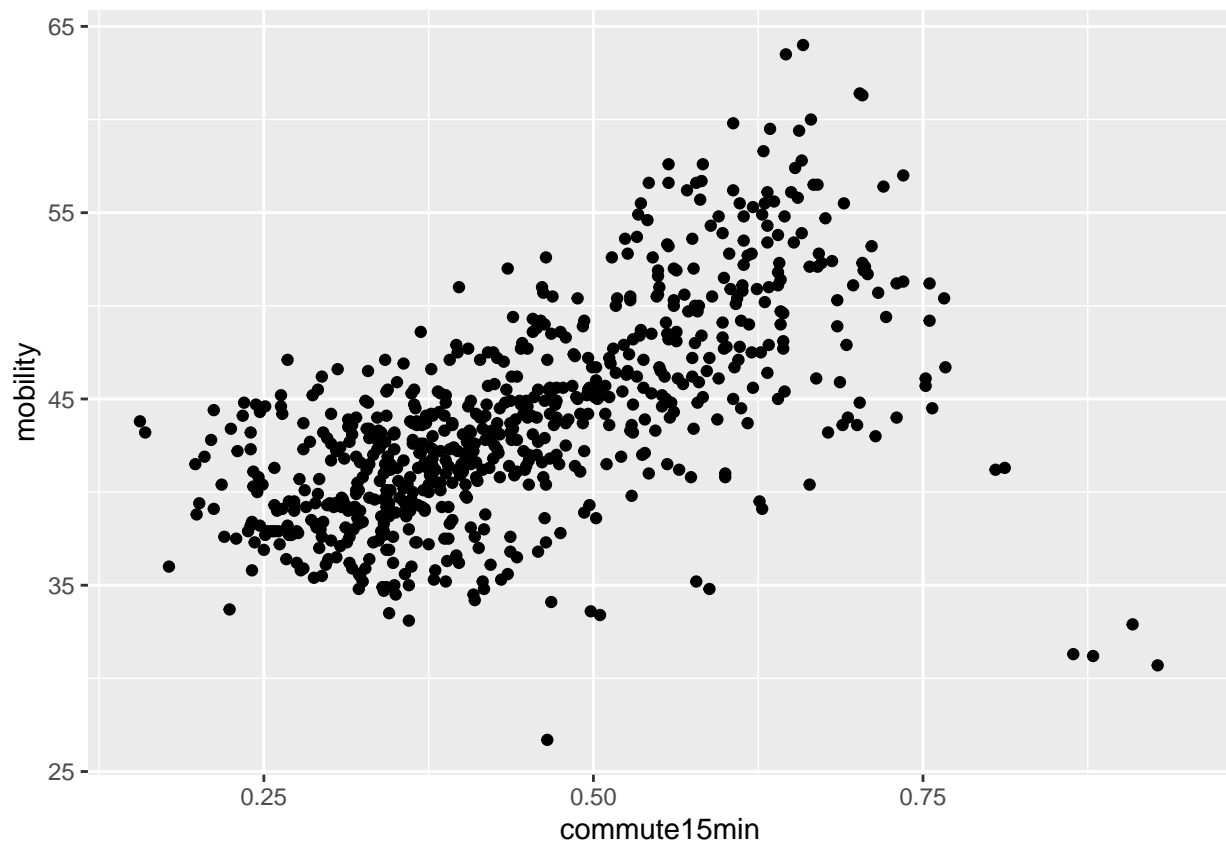
```
mobility_commute_scatter <- ggplot(cz,
                                   aes(x = commute15min, y = mobility))
mobility_commute_scatter + geom_point()
```

```
## Warning: Removed 32 rows containing missing values (geom_point).
```



You can see that there is a warning message alerting us to the fact that there are 32 missing values. To get rid of that, add `warning=FALSE` to the code chunk header:

```
mobility_commute_scatter <- ggplot(cz,  
                                   aes(x = commute15min, y = mobility))  
  
mobility_commute_scatter + geom_point()
```



Exercise With Other Variables

Take a few minutes to explore how other variables in this dataset are associated with mobility. What is a relationship where you would expect a negative association? What is a relationship where you would expect a positive association? What is a relationship where you would expect no association?

Here are the other four of the “big five” variables correlated with mobility:

- `gini` = Gini coefficient of income inequality; higher gini values indicate more inequality
- `social_capital` = Social capital index
- `frac_children_single_mothers` = Proportion of children living in single-parent households
- `hs_dropout_rate_adj` = High school dropout rate adjusted for family income; positive values indicate that the hs dropout rate is larger than expected given a commuting zone’s median family income, and negative values indicate that the hs dropout rate is smaller than expected given a commuting zone’s median family income

REPLACE THIS LINE WITH YOUR CODE

Plots With Labeled Points

The plots we have been making so far show points for every commuting zone (for which data are available). It is often helpful to identify specific points that are important for the analysis. For example, you might want to isolate the point for a specific commuting zone. There are several ways to do this.

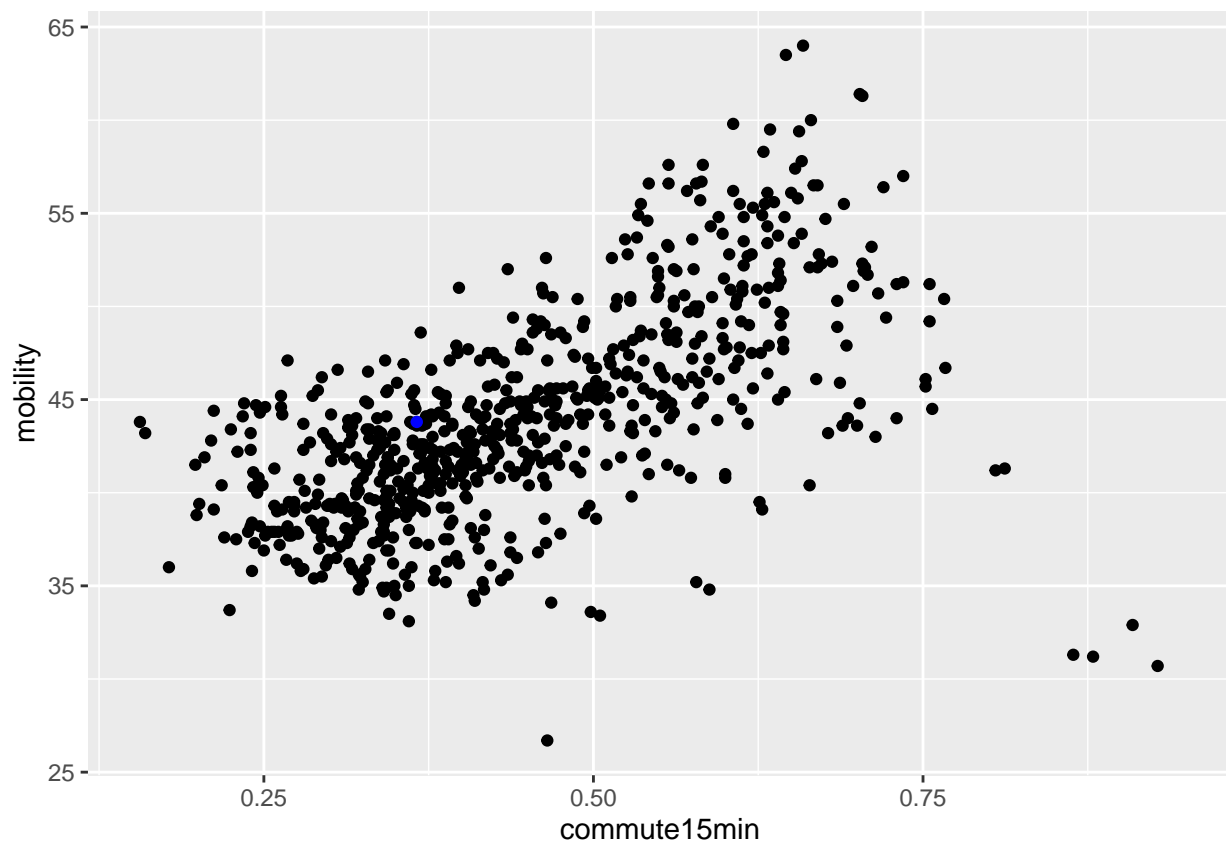
One way is to remember that ggplot is just a collection of layers. The idea with this approach is that on top of our existing plot we will add another layer that only has the point we want to identify. To do so, we need to create another data frame that only has that observation. We know how to do that using the `filter()` function in dplyr.

As an example, let's highlight the point for Burlington, VT's commuting zone in blue. First, create the Burlington data frame.

```
burlington <- filter(cz, cz_name == "Burlington" & state == "VT")  
# There are three commuting zones named Burlington, so we should add the state.
```

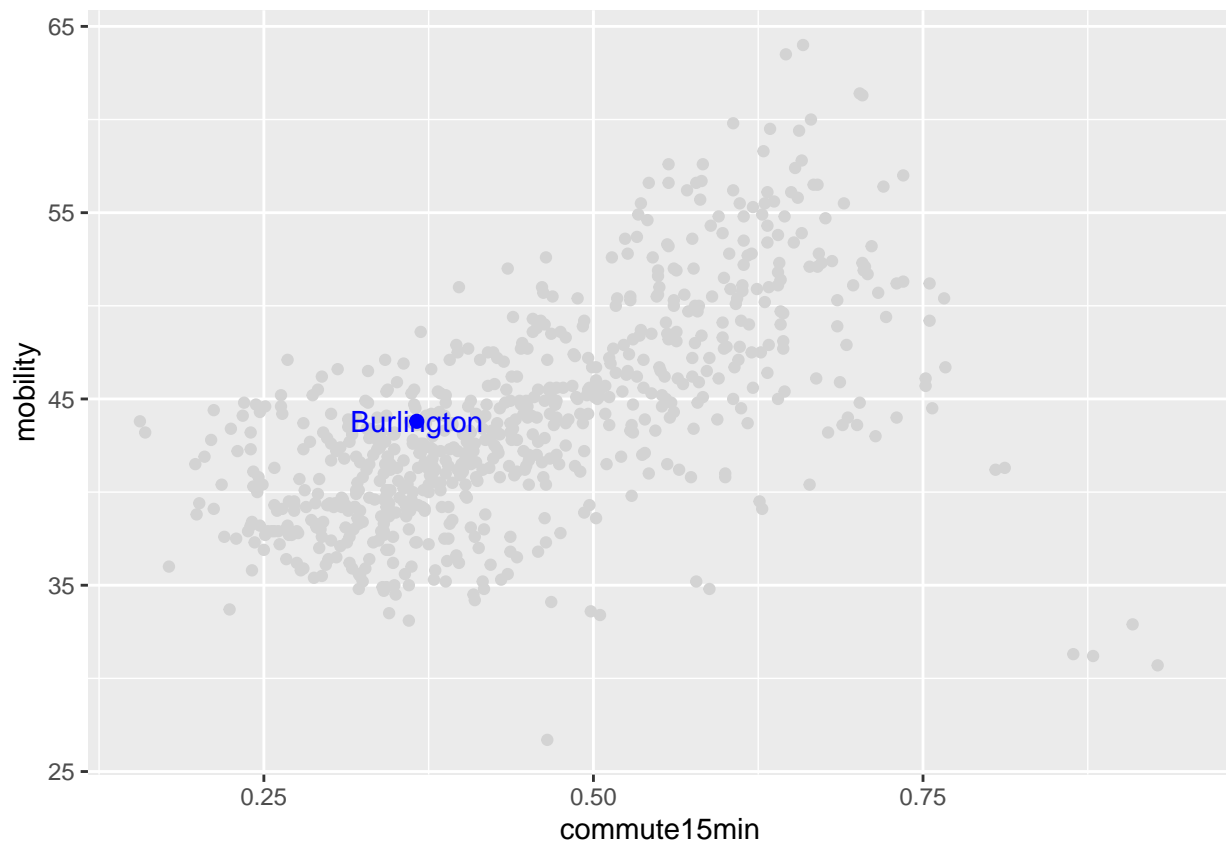
Now add another `geom_point()` layer for this new data frame to our existing plot. We'll have to give the name of the data frame and the aesthetic map again:

```
mobility_commute_scatter + geom_point() +  
  geom_point(data = burlington, aes(x = commute15min, y = mobility),  
            color = "Blue")
```



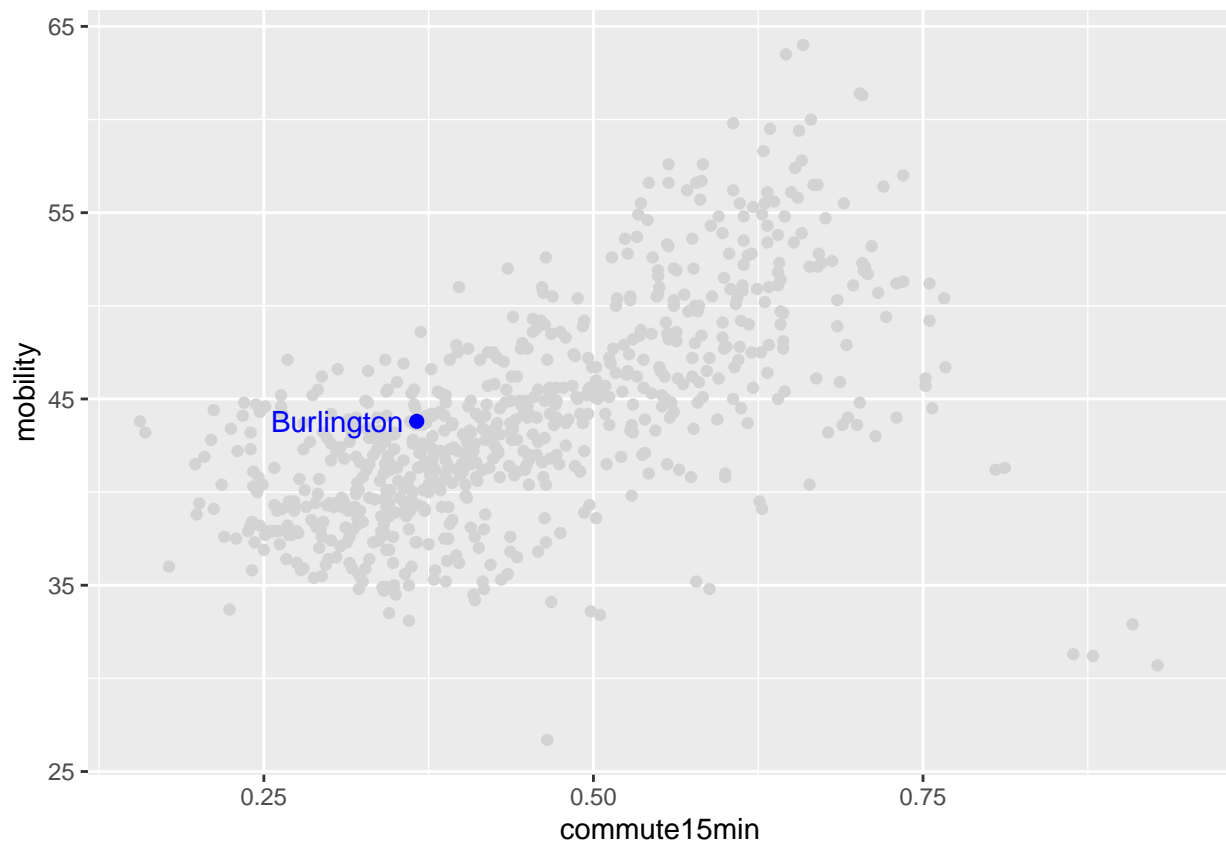
The blue point for Burlington is there, but it is hard to see. We can improve the plot by changing the color of all the points in the original plot to light gray, doubling the size of the Burlington point, and adding a blue label to identify it:

```
mobility_commute_scatter + geom_point(color = "Light Gray") +  
  geom_point(data = burlington, aes(x = commute15min, y = mobility),  
            color = "Blue", size = 2) +  
  geom_text(data = burlington, aes(label = cz_name),  
            color = "Blue")
```

We can use `vjust` and `hjust` to nudge the label. Let's nudge it slightly to the left.

```
mobility_commute_scatter + geom_point(color = "Light Gray") +  
  geom_point(data = burlington, aes(x = commute15min, y = mobility),  
    color = "Blue", size = 2) +  
  geom_text(data = burlington, aes(label = cz_name) ,  
    color = "Blue", hjust = 1.1)
```

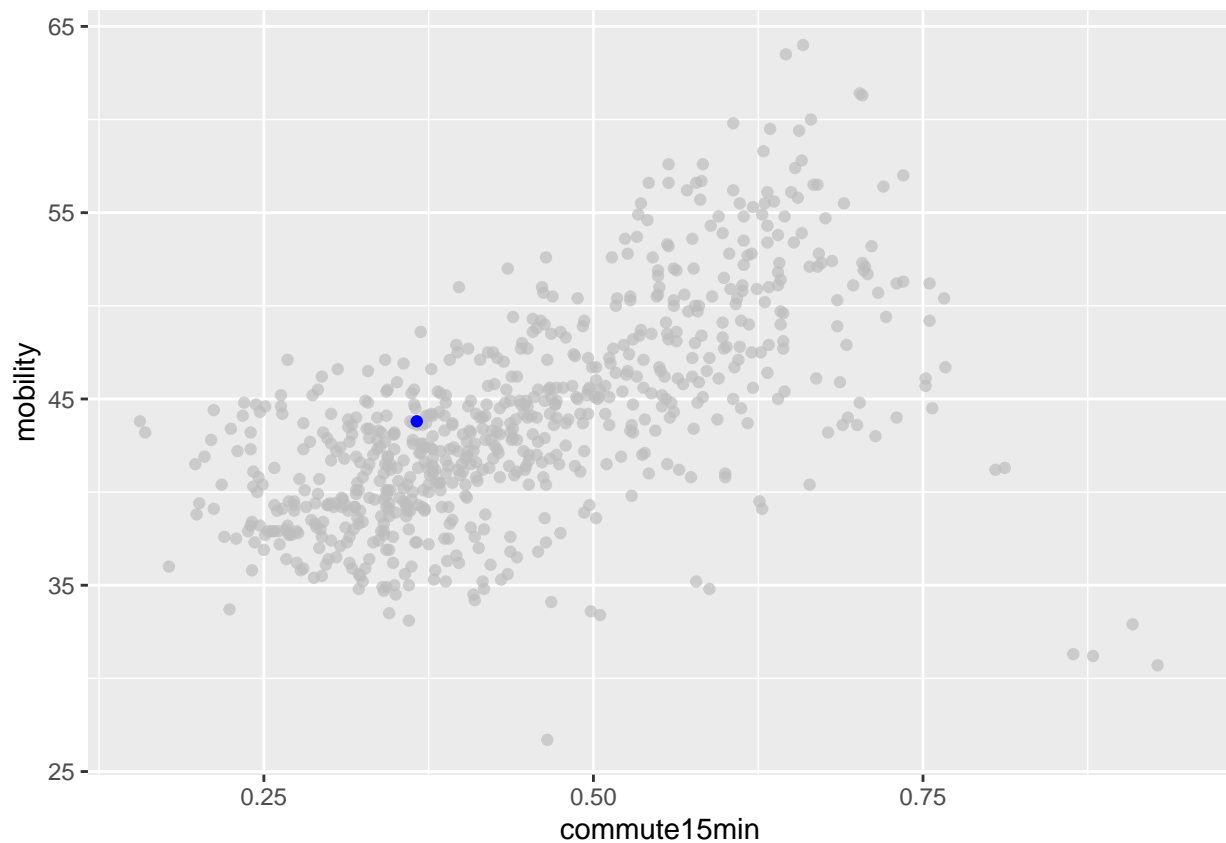


A second (much easier) option uses the `gghighlight` package. Install and load the package.

```
#install.packages("gghighlight")  
library(gghighlight)
```

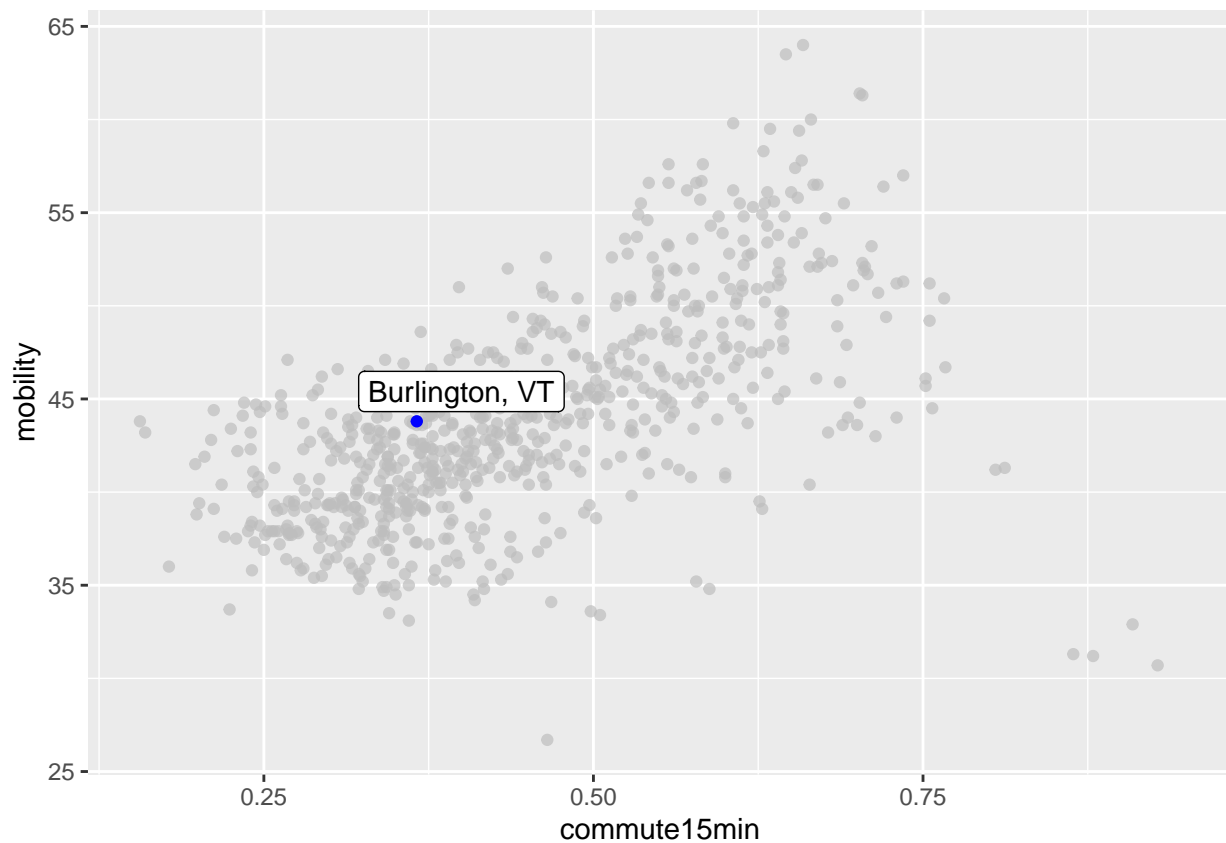
Now we simply add the `gghighlight()` function and include the same conditions by which we filtered in the previous example. Note that we will add the `color` option to the `geom_point()` function; all non-highlighted points will be turned gray:

```
mobility_commute_scatter + geom_point(color = "Blue") +  
  gghighlight(cz_name=="Burlington" & state=="VT")
```



We can add a label with the `label` key = option. You can list a variable name (`label_key = cz_name`) or a specific string as in this example:

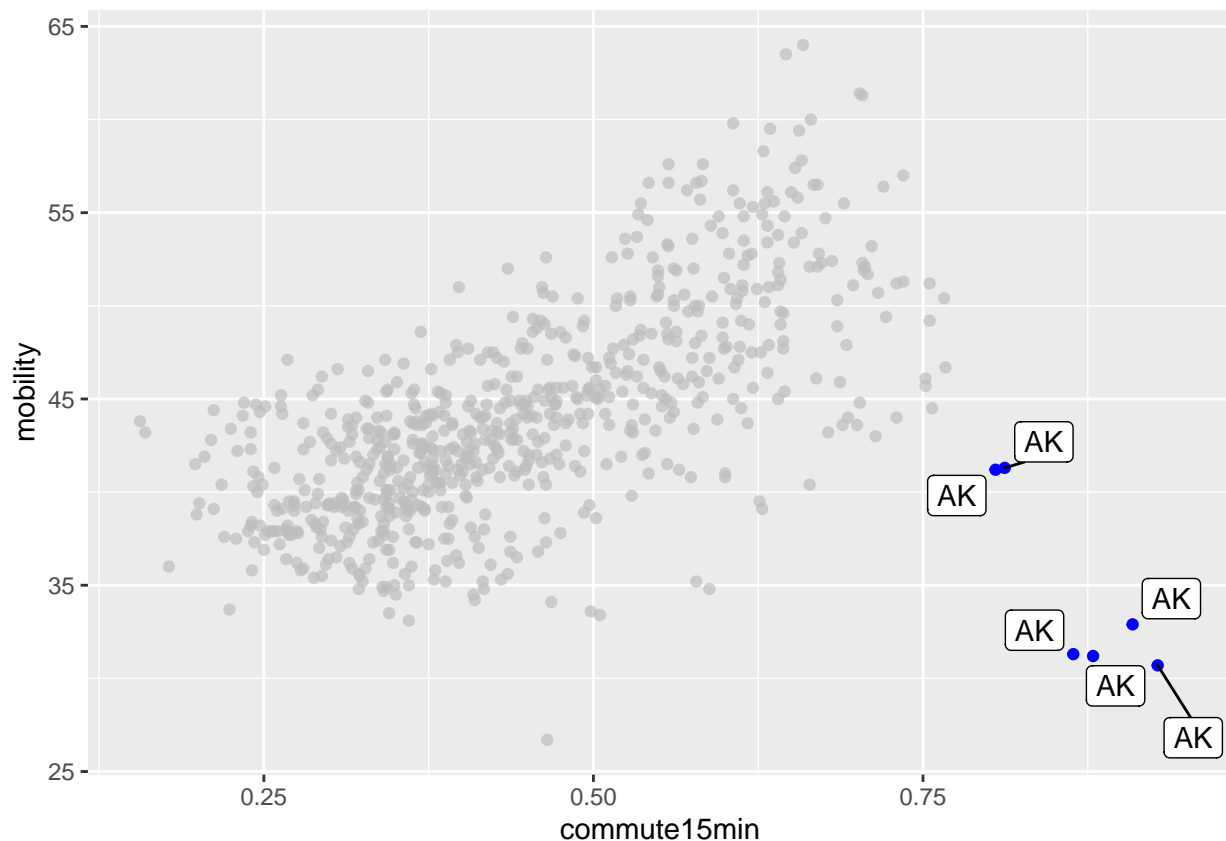
```
mobility_commute_scatter + geom_point(color = "Blue") +
  gghighlight(cz_name=="Burlington" & state=="VT",
    label_key = c("Burlington, VT"))
```



You can highlight points that fall within ranges of values as well. For example, try highlighting the commuting zones where more than 80% of residents commute 15 minutes or less, and label them by state.

REPLACE THIS LINE WITH YOUR CODE

```
mobility_commute_scatter + geom_point(color = "Blue") +  
  gghighlight(commute15min > .80,  
    label_key = state)
```



Introducing Multivariate Relationships

Identifying individual points is one way to dig deeper into how two variables are associated. Another analytical tool is to examine whether the association differs at specific values of another variable. For example, what would you expect the relationship to be between racial segregation and income?

REPLACE THIS LINE WITH YOUR CODE

```
cor(cz$racial_seg, cz$hh_income, use = "complete")
```

```
## [1] 0.1445054
```

Would this association be the same in urban and rural commuting zones?

REPLACE THIS LINE WITH YOUR CODE

```
cor(cz$racial_seg[cz$urban==0], cz$hh_income[cz$urban==0], use = "complete")
```

```
## [1] -0.2621044
```

```
cor(cz$racial_seg[cz$urban==1], cz$hh_income[cz$urban==1], use = "complete")
```

```
## [1] 0.3231209
```

```
correlations <- cz %>% group_by(urban) %>%
  summarize(cor = cor(racial_seg, hh_income, use = "complete"))
```

```
correlations
```

```
## # A tibble: 2 x 2
##   urban    cor
##   <int> <dbl>
## 1     0 -0.262
## 2     1  0.323
```

Remember we can use `facet_grid()` to display two plots in one figure. Before we get there, let's make sure the `urban` variable is recognized as a factor variable, and change the labels from "0" and "1" to "Rural" and "Urban".

```
cz$urban <- factor(cz$urban,
  levels=c("0", "1"),
  labels=c("Rural", "Urban"))
```

```
library(ggplot2)
```

```
scatter <- ggplot(cz, aes(x = hh_income, y = racial_seg))
scatter + geom_point(aes(color = urban)) +
  facet_grid(.~urban) +
  scale_color_manual(values = c("Red", "Blue")) +
  guides(color = FALSE) +
  labs(x = "Median Income", y = "Racial Segregation",
  title = "Rural-Urban Differences in Association\nBetween Income and Racial Segregation",
  subtitle = "Data from Opportunity Insights")
```

Rural–Urban Differences in Association
Between Income and Racial Segregation

Data from Opportunity Insights

