More Association and Correlation

Matt Lawrence

10/2/2019

Today we will continue 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 website which can be accessed `here`.

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

Recap

On Monday we began thinking about how two variables are associated by finding whether they are positively or negatively correlated and how weak or strong the correlation is. We looked at the variables measuring the proportion of workers in a commuting zone commuting 15 minutes or less, income, racial segregation, labor force participation, and a measure of upward mobility.

When we finished last class, we were looking at the correlation between `commute15min` and `mobility`. How would you describe the correlation between these two variables?

REPLACE THIS LINE WITH YOUR CODE

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

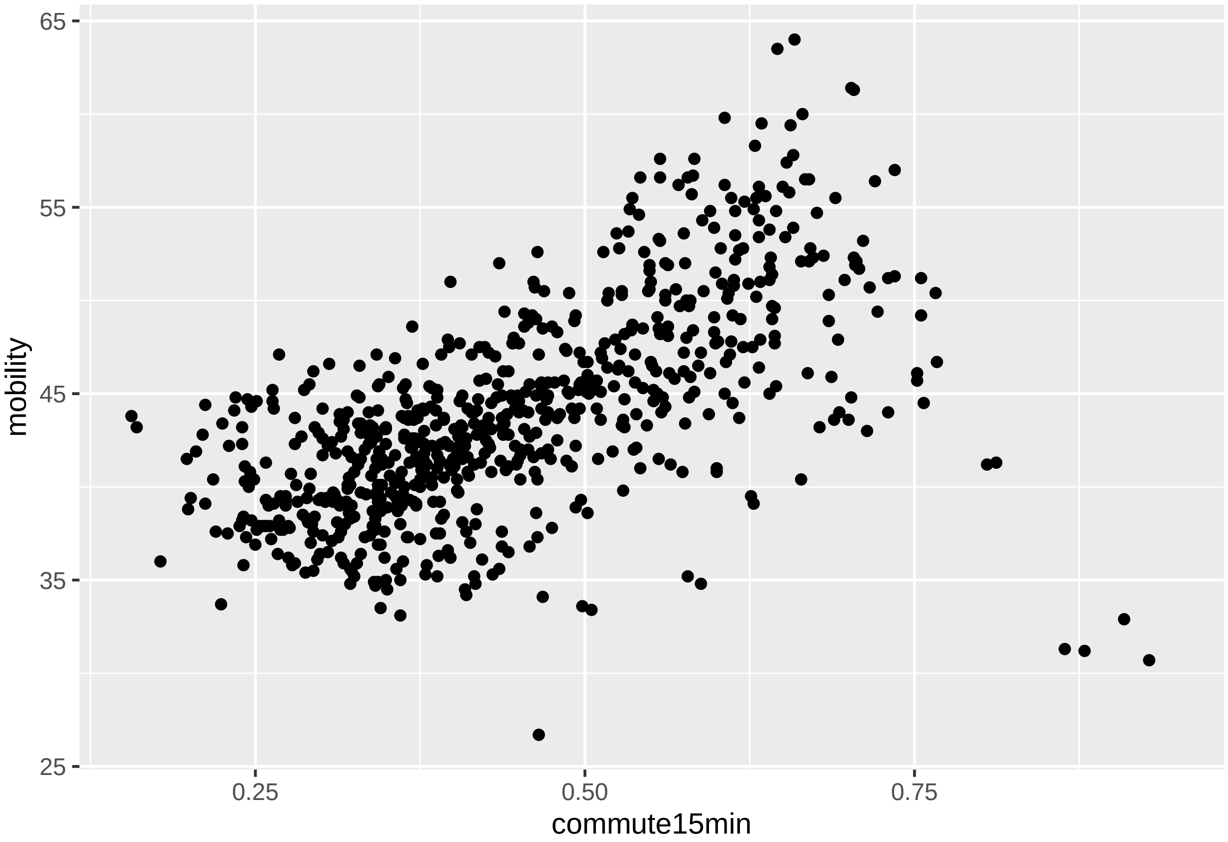
```
## [1] 0.6048691
```

And how can you make a scatterplot showing the correlation between these two variables?

REPLACE THIS LINE WITH YOUR CODE

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

mobility_commute_scatter + geom_point()
```



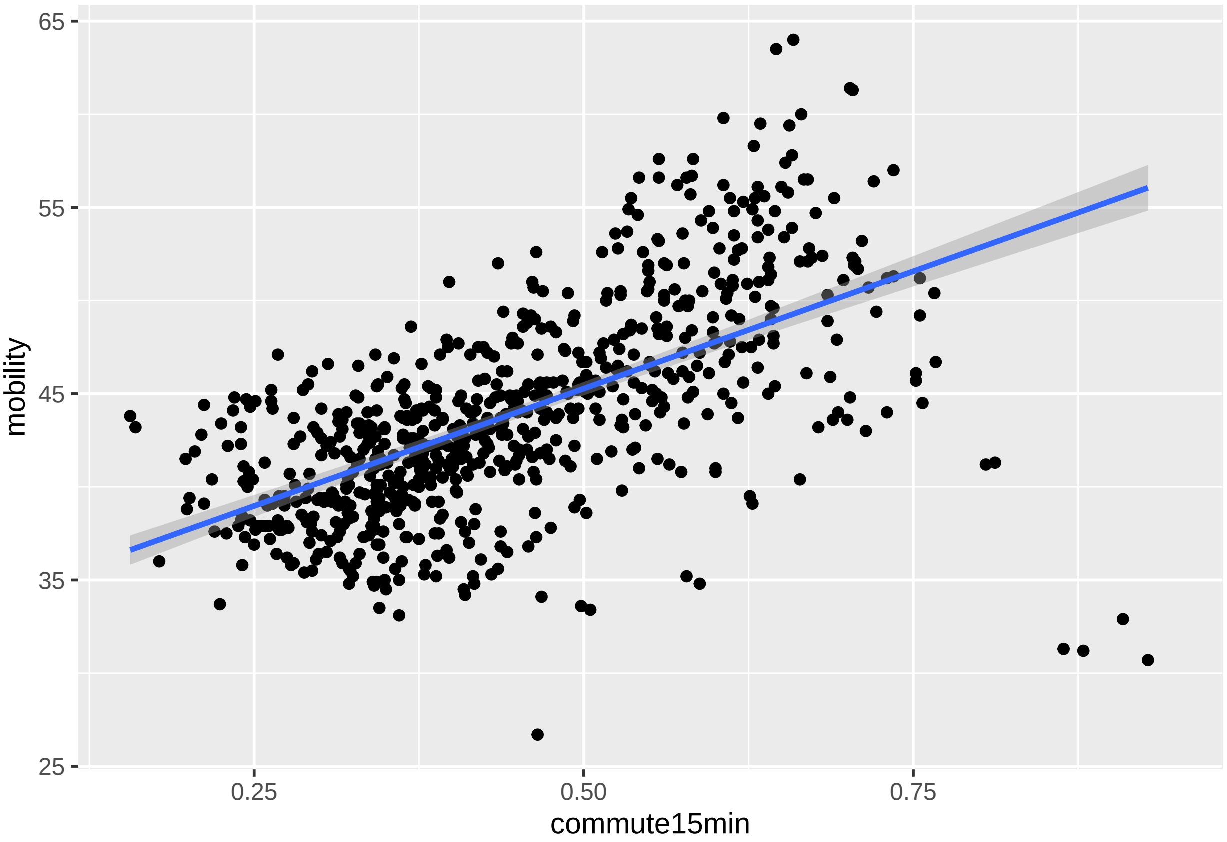
Introducing Lines of Best Fit

Scatterplots often include lines to help visualize the direction and strength of correlations. We will do much more with these lines at the end of the semester. But for now, it could be good to know how to include them. To do so, add another layer to the plot with the `geom_smooth(method = lm)` function. The smooth function finds a pattern across all the points, and the `method = lm` option says we want the pattern to be based on the "linear model" used in basic regression models.

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

mobility_commute_scatter + geom_point() + geom_smooth(method = lm)
```

```
## 'geom_smooth()' using formula 'y ~ x'
```



Exercise With Other Variables

Let's spend some time exploring a few of the other associations in our data using the following variables:

- `cz_name` = commuting zone name
- `state` = commuting zone state
- `mobility` = measure of absolute upward mobility
- `gini` = Gini coefficient of income inequality; higher gini values indicate more inequality
- `social_capital_index` = Social capital index
- `frac_foreign_born` = Proportion of residents born in other countries
- `frac_children_single_mothers` = Proportion of children living in single-parent households
- `school_expenditures_per_student` = Average expenditures per student in public schools
- `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
- `urban` = binary variable for urban (1) or rural (0) commuting zone
- `hh_income` = median household family income in commuting zone
- `racial_seg` = measure of racial segregation

Take a few minutes to explore how some of the variables in the `cz` data frame 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?

REPLACE THIS LINE WITH YOUR CODE

Positive = Social Capital (0.641)

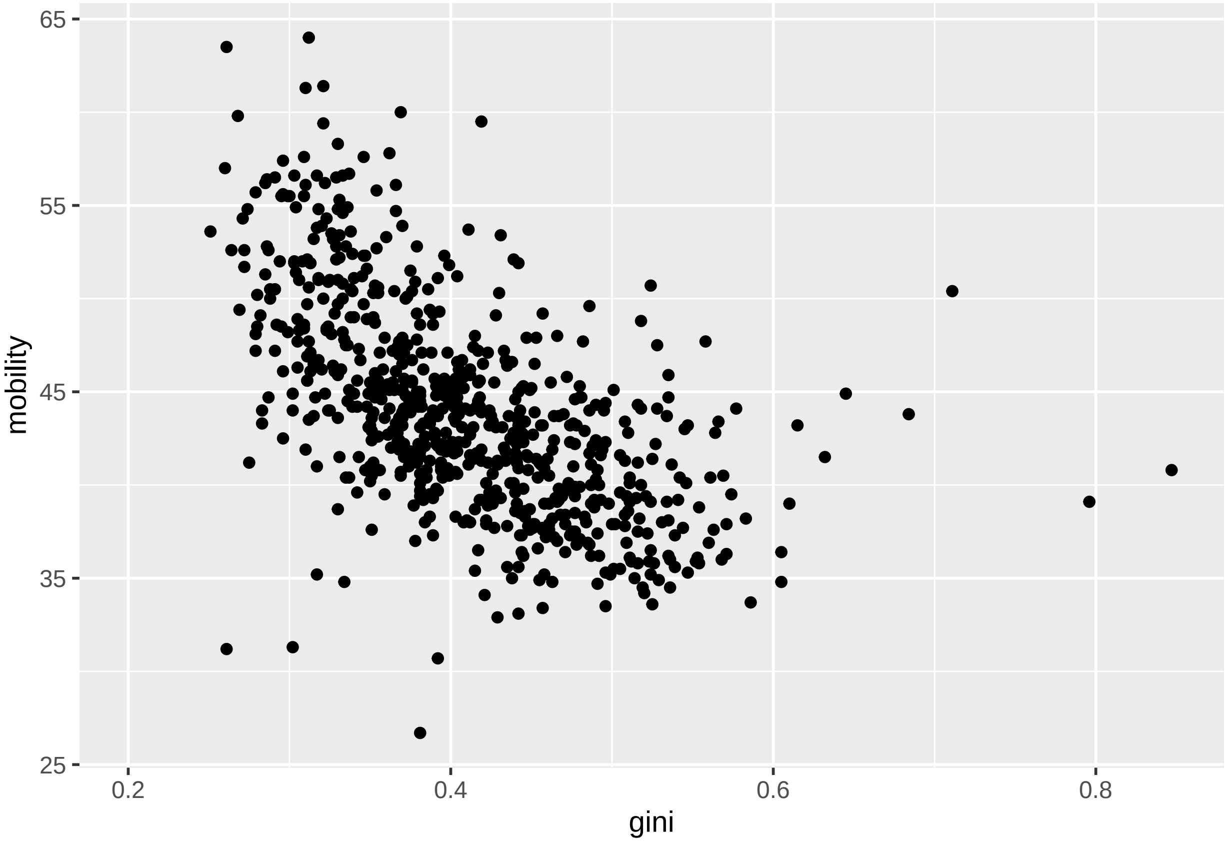
```
cor(cz$social_capital_index, cz$mobility,
    use = "complete")
```

```
## [1] 0.6405067
```

Negative = Gini (-0.578)

```
ggplot(cz, aes(x = gini, y = mobility)) + geom_point()
```

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



None = Foreign Born (-0.027)

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

```
## [1] -0.02713187
```

Correlations With Indexing

```
florida <- cz %>%
    filter(state == "FL")

cor(florida$social_capital_index, florida$mobility,
    use = "complete")
```

```
## [1] -0.2975421
```

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.

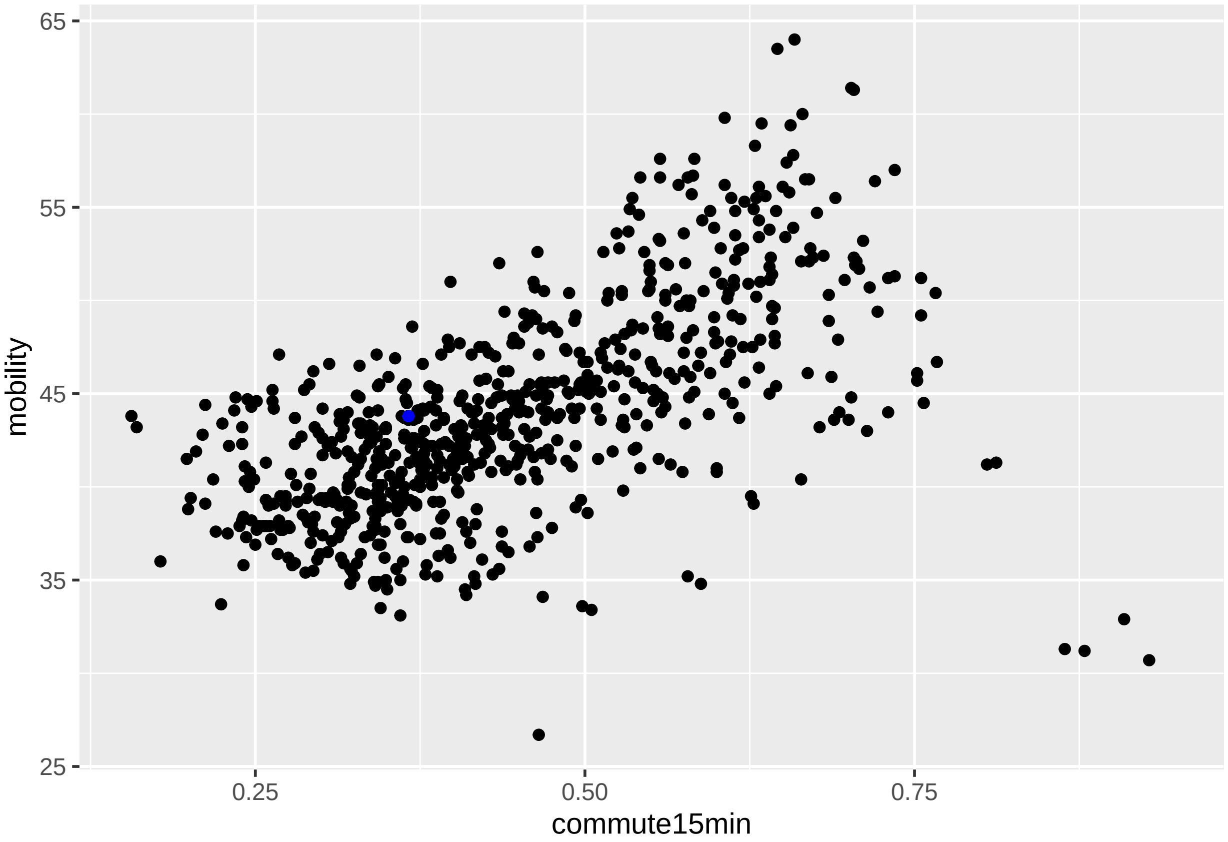
One way is to remember that ggplot is just a collection of layers. The idea with this approach is that on top of our existing plot we will add another layer that only has the point we want to identify. To do so, we need to create another data frame that only has that observation. We know how to do that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

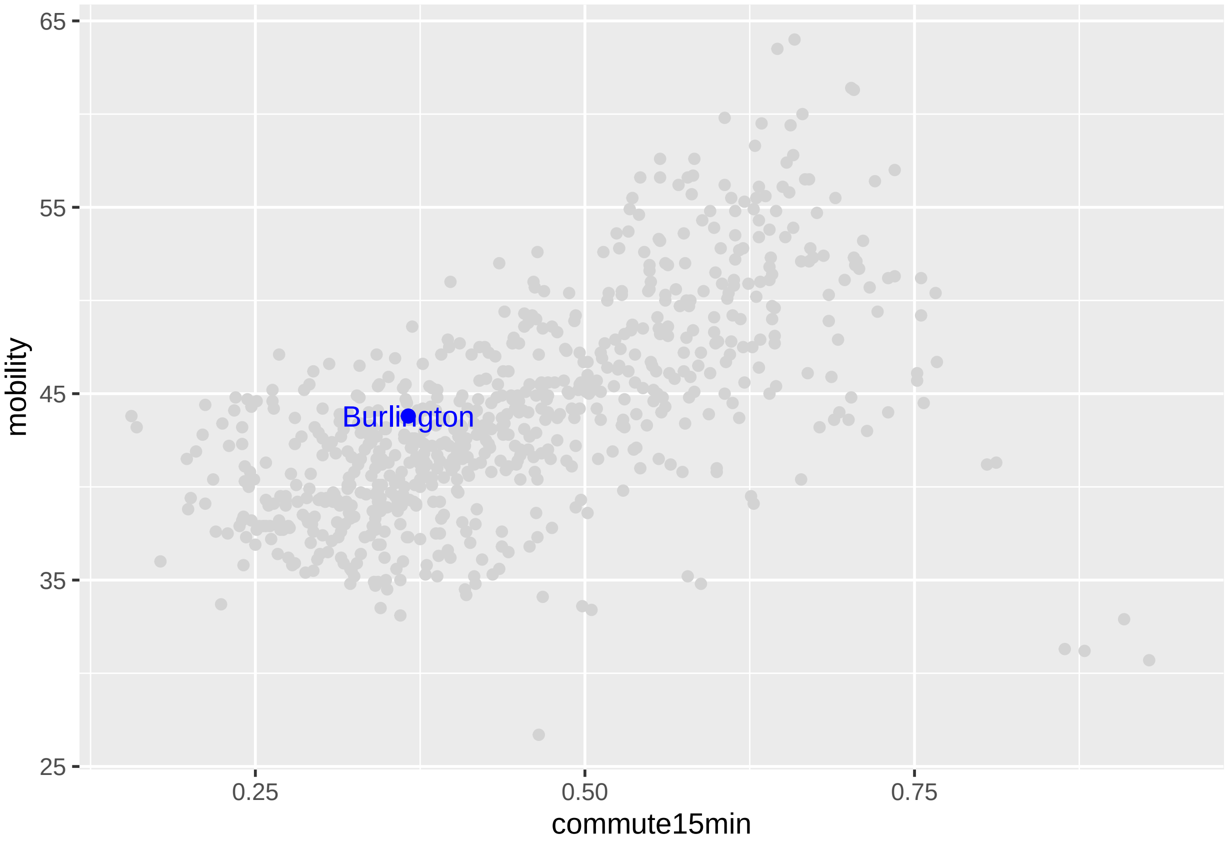
Now add another `geom_point()` layer for this new data frame to our existing plot. We'll have to give the name of the data frame and the aesthetic map again:

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



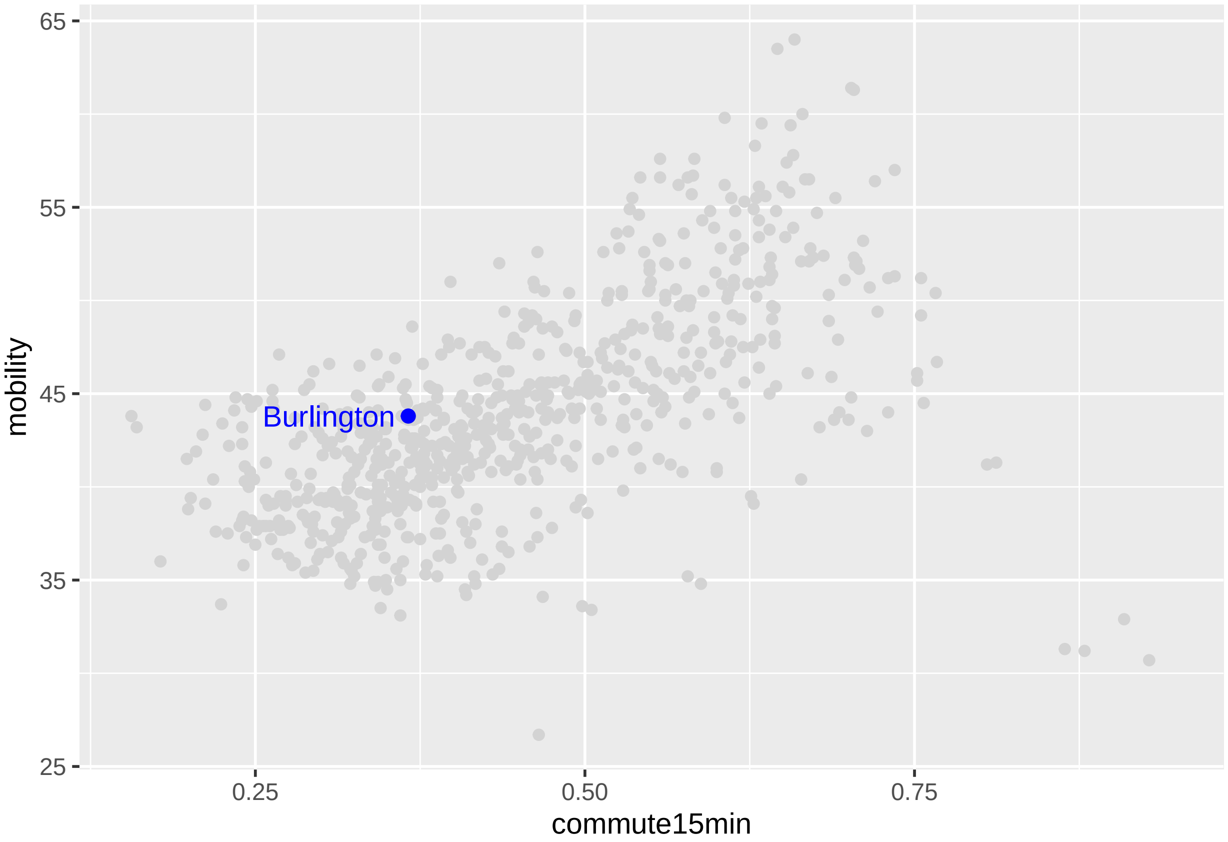
The blue point for Burlington is there, but it is hard to see. We can improve the plot by changing the color of all the points in the original plot to light gray, doubling the size of the Burlington point, and adding a blue label to identify it:

```
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)
```



```
cz$cz_name[cz$state=="NY"]
```

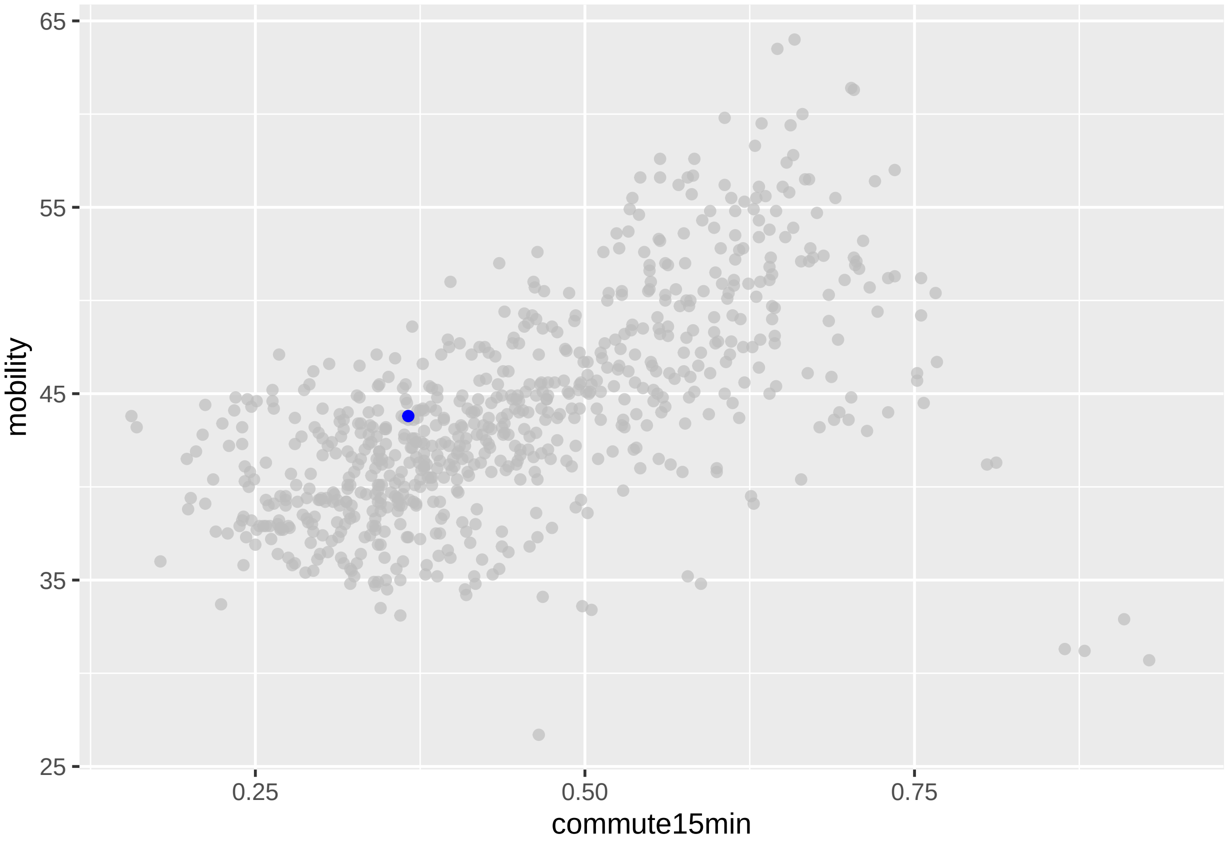
```
##  [1] "Watertown"    "Plattsburgh"  "Olean"        "Amsterdam"    "Elmira"
##  [6] "Oneonta"      "Syracuse"     "Union"        "Buffalo"      "Albany"
## [11] "Poughkeepsie" "New York"
```

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

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

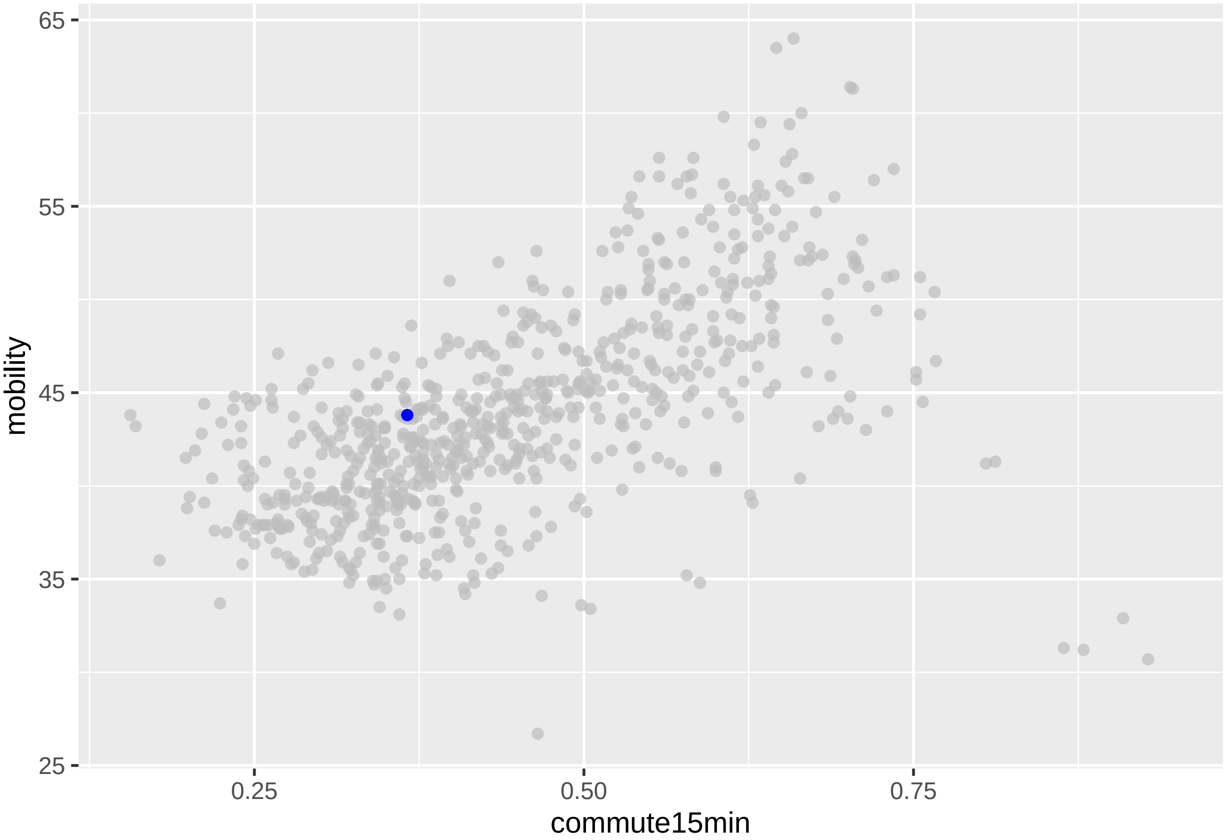
Now we simply add the `gghighlight()` function and include the same conditions by which we filtered in the previous example. Note that we will add the color option to the geom_point() function; all non-highlighted points will be turned gray:

```
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:

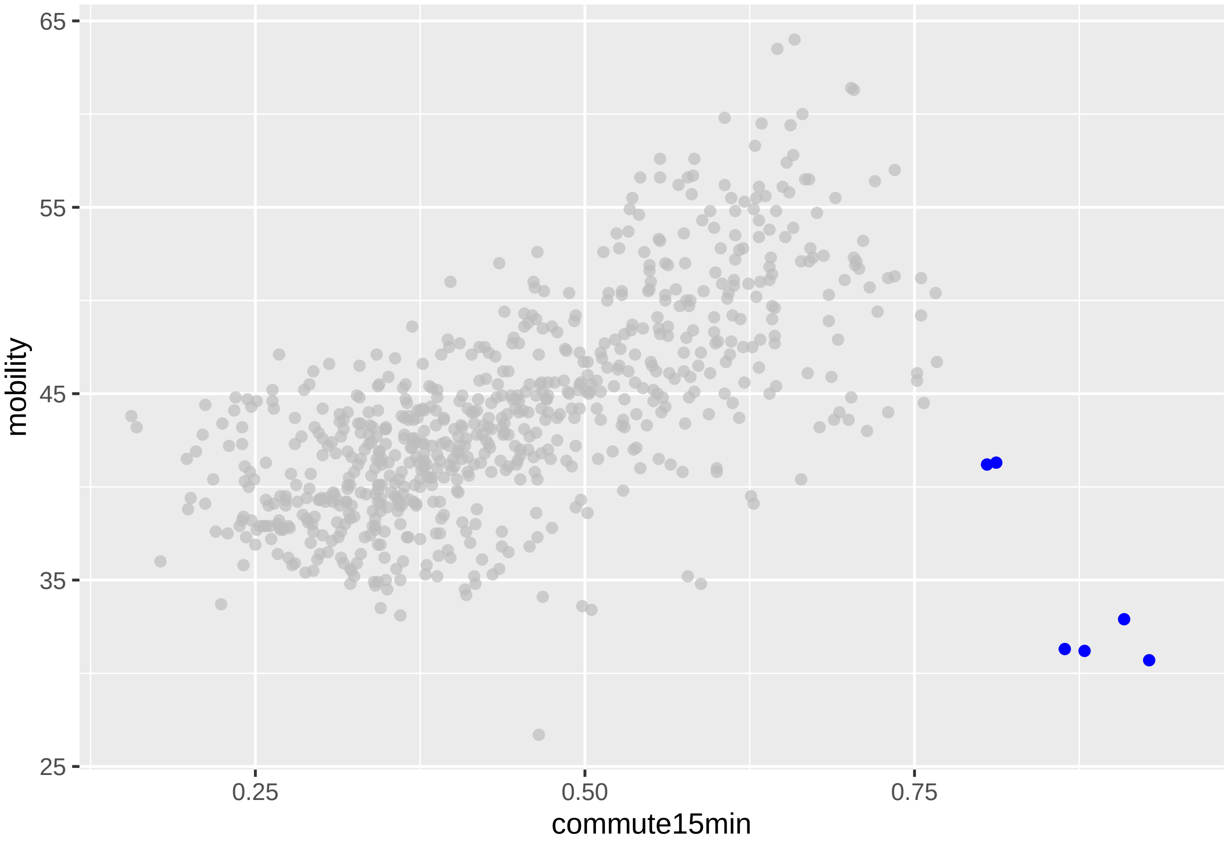
```
mobility_commute_scatter + geom_point(color = "Blue") +
     gghighlight(cz_name=="Burlington" & state=="VT",
                 label_key = cz_state)
```



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 = cz_state)
```


65
55
45
35
25
mobility
0.25
0.50
0.75
commute15min