

(Part B) Enhance the bike-sharing demand prediction app with city details plots

Estimated time needed: 90 minutes

Note: This lab is a continuation of the previous lab “Hands-on Lab: Build a bike-sharing demand prediction app with R Shiny and Leaflet”. Open the same project and perform the tasks as per the instructions in this lab to enhance plots with more details.

In the previous lab, you have built a R Shiny app with leaflet to show the max bike-sharing demand predictions for each city. In the app, users can also select a specific city to show its detailed weather label.

Next, you are asked to use ggplot to render some more detailed plots such as bike-sharing prediction trend, temperature trend, humidity and bike-sharing demand prediction correlation, when users zoom-in to a city.

This lab includes the following tasks:

- **TASK 1:** Add a static temperature trend line
- **TASK 2:** Add an interactive bike-sharing demand prediction trend line
- **TASK 3:** Add a static humidity and bike-sharing demand prediction correlation plot

Let's start by opening RStudio project you created in the previous lab.

Task 1: Add a static temperature trend line

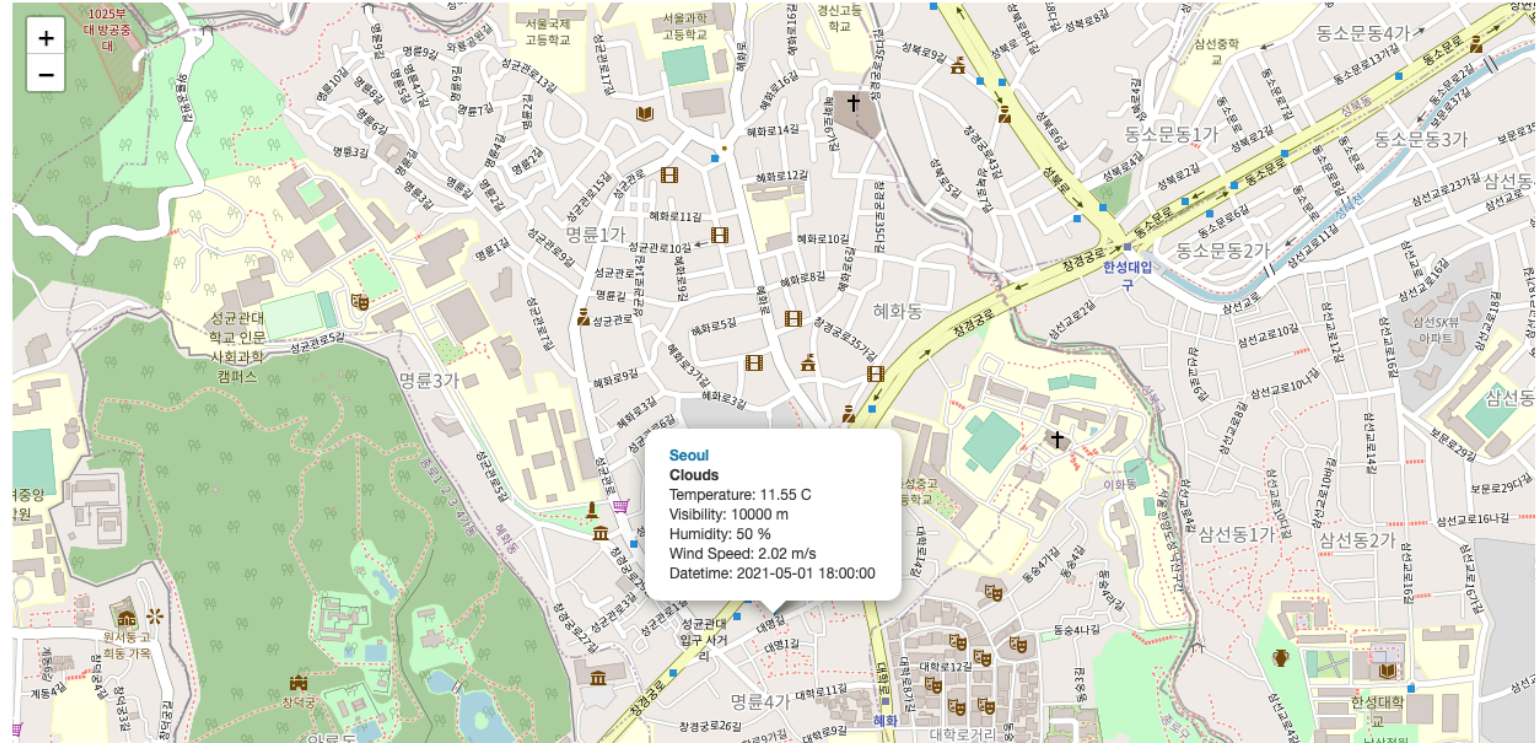
When users select a city to zoom-in, one important information they may want to know is the temperature trend in the next 5-days. Such trend information can be visualized using line chart.

- Open ui.R, add a plot output with in sidebarPanel() with id temp_line and any other options you want to add such as height and width.
- Open server.R, locate to the if-else code branch checking if All cities or a specific city is selected in the observeEvent() function, then add a temperature trend plot using renderPlot(...) function with following configurations:
 - Add a geom_line to show the trend line
 - Add a geom_point to show temperature point
 - Add a geom_text to show the value on the plot

Note that the data for the trend line should come directly from city_weather_bike_df, instead of the cities_max_bike data frame used for leaflet.

Your temperature trend line should look similar to the following example

Bike-sharing demand prediction app



From this plot, users can easily check the temperature trend for the selected city.

Task 2: Add an interactive bike-sharing demand prediction trend line

Another important trend line for a selected city is its bike-sharing demand prediction trend. It will also be a line chart but let's try to add some interaction to the chart so that users can click the point and see its bike-sharing demand and datetime.

- Open ui.R, add a plot output with in sidebarPanel() with id bike_line and a click event click = "plot_click". You may add other options you want such as height and width.
- In ui.R, also add a verbatimTextOutput("bike_date_output") text output with id bike_date_output. So that the point you click on the plot can be shown in here.
- Open server.R, locate to the if-else code branch checking if all cities or a specific city is selected in the observeEvent() function, then add a bike-sharing demand prediction trend plot using renderPlot(...) function with following configurations:
 - Add a geom_line to show bike-sharing prediction trend line with the x axis represents the FORECASTDATETIME and the y axis represents BIKE_PREDICTION
 - Add a geom_point to show bike-sharing
 - Add a geom_text to show the value on the plot

Users then should able to click on the trend line to show the bike-sharing prediction and datetime values so let's add a text output rendering here.

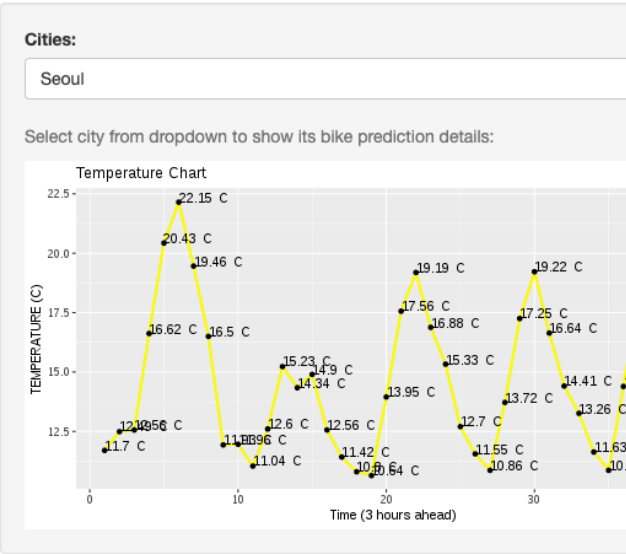
- In server.R file, add a renderText() function to create a formatted text output showing the clicked x and y value.

```
1. 1
2. 2
3. 3
4. 4

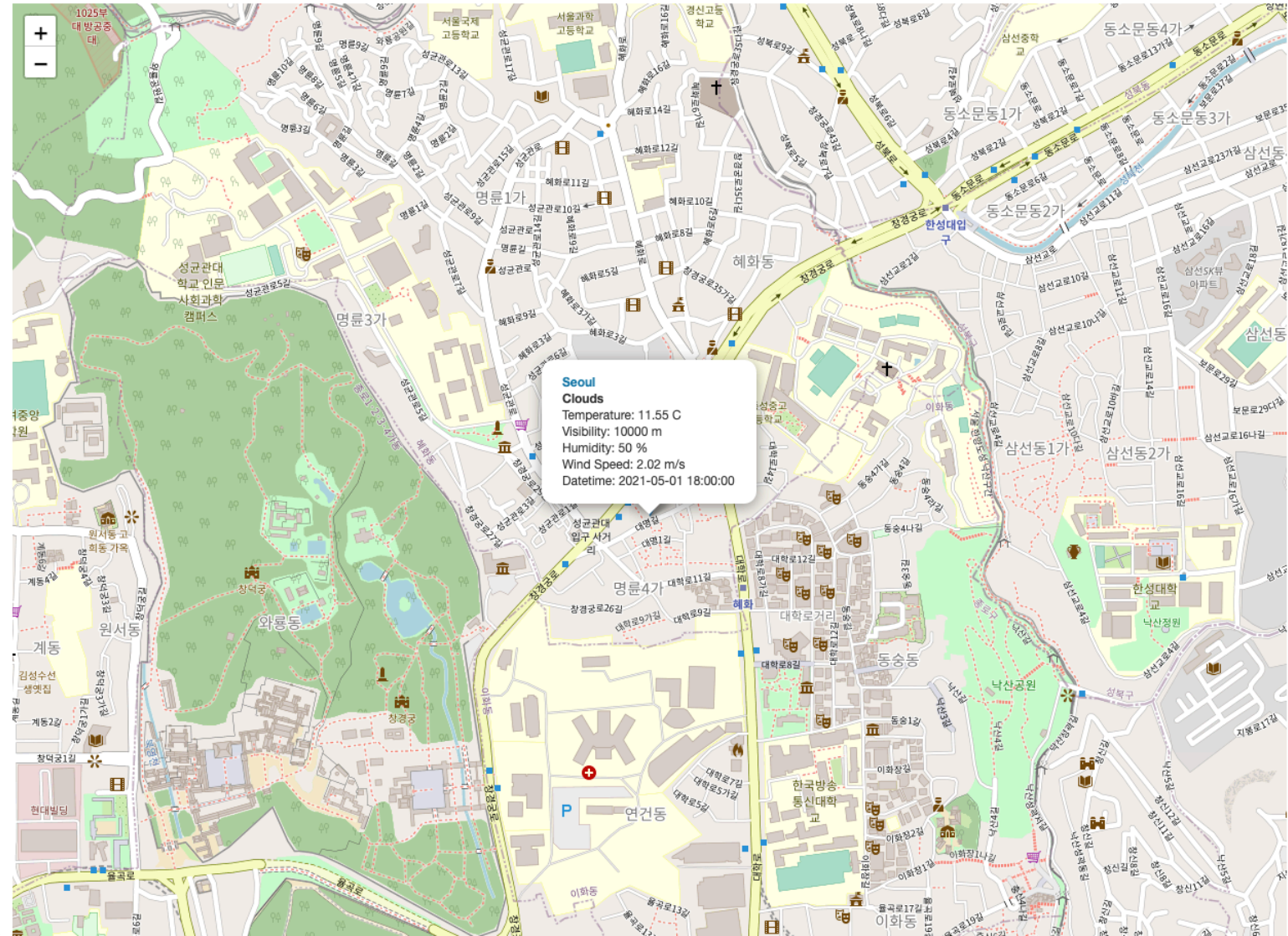
1. output$bike_date_output <- renderText({
2.   ...
3. })
4.
```

Copied!

Your bike-sharing demand prediction trend plot should look similar to the following example:



Bike-sharing demand prediction app



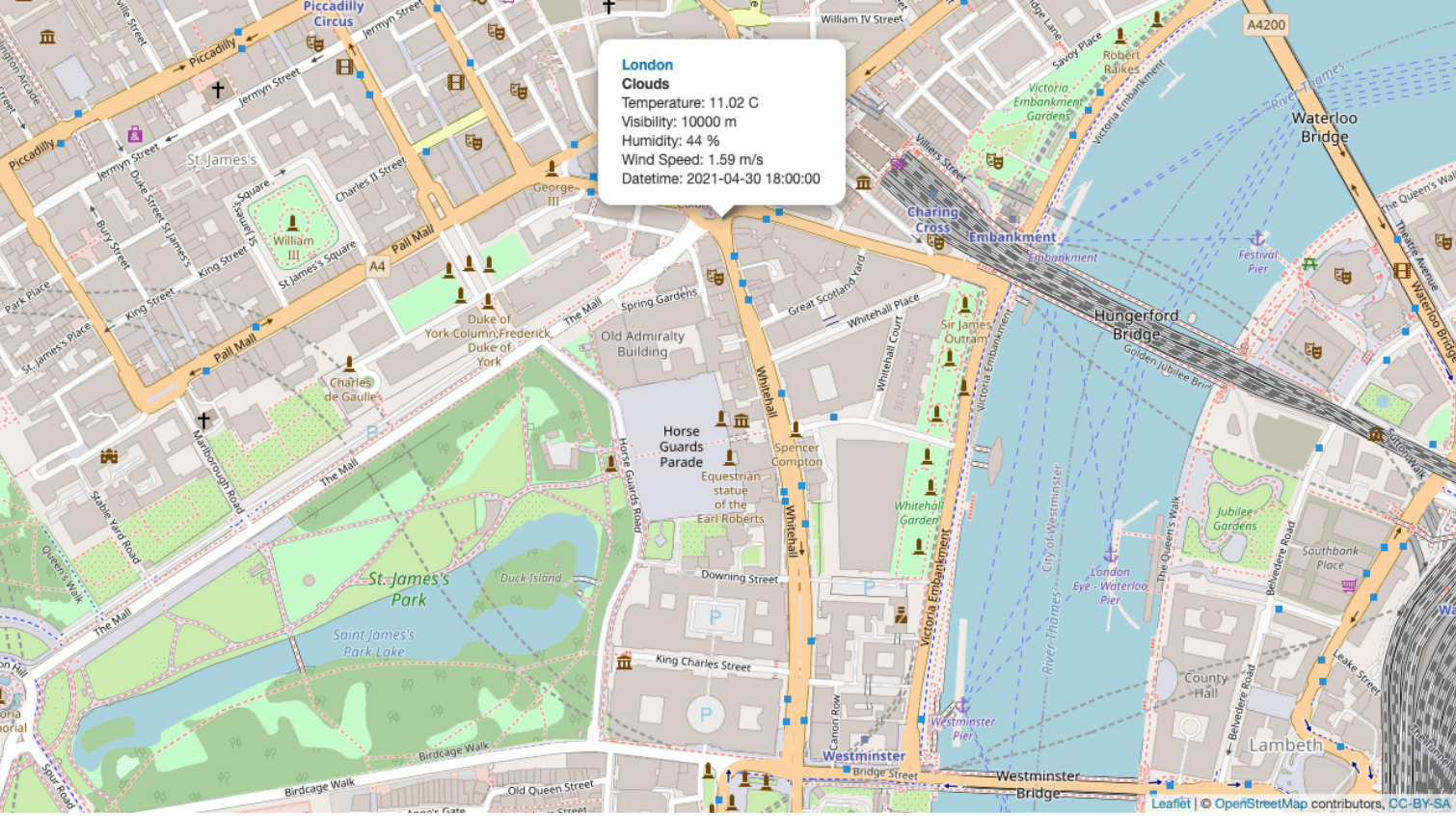
From this trend line, users can easily see the bike-demand prediction trend in the next 5-days. Users can also click the points on the plot to show its detailed prediction and datetime value.

Task 3: Add a static humidity and bike-sharing demand prediction correlation plot

From the previous analysis, we know the bike-sharing demand value highly depends on the humidity value. So it would be great to show such correlation scatter point plot when a city is selected.

- Open `ui.R`, add a plot output with `in sidebarPanel()` with `id humidity_pred_chart`. You may add other options you want such as height and width.
- Open `server.R`, locate to the `if-else` code branch checking if all cities or a specific city is selected in the `observeEvent()` function, then add a trend plot using `renderPlot(...)` function with following configurations:
 - Add a `geom_point()` with `x` axis represents HUMIDITY and `y` axis represents BIKE_PREDICTION
 - Add a `geom_smooth()` with method `lm` and formula `y ~ poly(x, 4)`. You may choose any high order polynomial terms here

Your correlation plot should look similar to the following example:



Next steps

Congratulations! You are almost finished the entire capstone project. Next, you will be preparing how to present your work and data-driven insights using PowerPoint.

Author(s)

[Yan Luo](#)

Other Contributor(s)

Jeff Grossman

