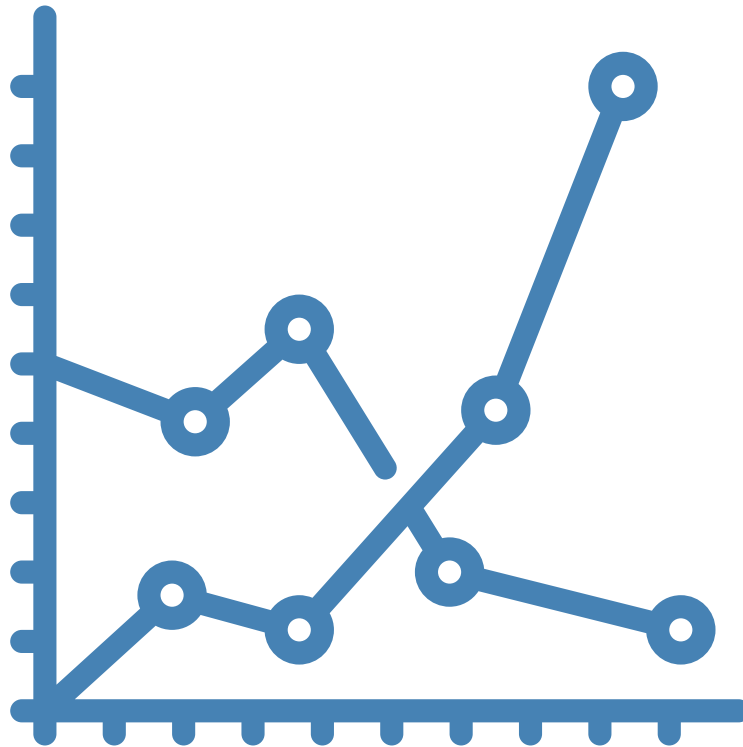# LEARN MORE 365

Annotations Lesson

Annotations

Greg Martin

Mapping text and labels

To include text or labels on your canvas, ggplot has two functions that are aptly named `geom_text()` and `geom_label()` respectively. Labels are simply text included in a box (as you will see in the examples below). In the example below, we've created a separate data object called `best_in_class` that has all of the same variables as the original `mpg` dataset but only contains rows for the cars that have the best highway fuel efficiency (the variable `hwy`) in their class. There are seven classes of car in the `mpg` dataset and so this new dataset has only seven rows. For each class of car, we want to create a label that identifies the `model` of car that represents the most fuel efficient car for its class. To do this we map the label aesthetic to the variable `model` and define the dataset as `best_in_class` as follows:

```r
best_in_class <- mpg %>%
  group_by(class) %>%
  filter(row_number(desc(hwy))==1)

ggplot(mpg, aes(displ, hwy))+
  geom_point(aes(color = class), shape = "square")+
  geom_text(aes(label = model),
                  data = best_in_class)+
  labs(x = "Engine size",
       y = "Highway fuel efficiency",
       color = "Class",
       title = "The most fuel efficient model in each class of car ")
```

The most fuel efficient model in each class of car
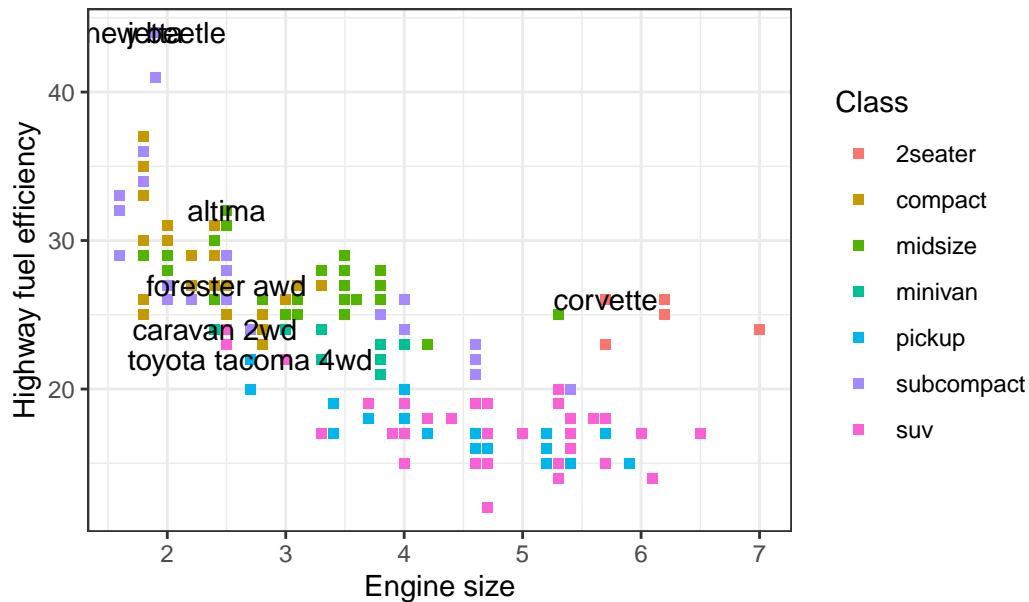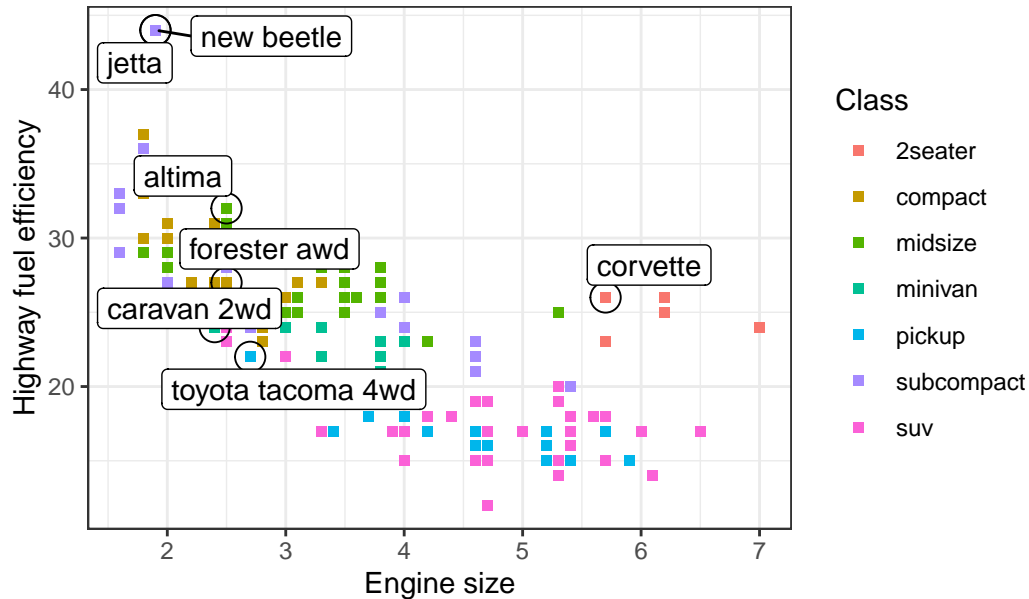

Preventing overlap of labels

Notice in the example above that the two text objects at the top left ("jetta" and "new beetle") overlap each other and can't be read. To overcome this problem we use a package called `ggrepel` to repel labels away from each other and prevent overlap.

In the code below, we've plotted the `mpg` data once again. In this plot we've created a circle around the data points where they represent the best model of car in their class (with respect to fuel efficiency when driving on the highway). Again we've used the stand-alone data object called `best_in_class` created above. When using the `geom_label_repel()` function we need to specify the data. Interestingly, the new data frame also contains `displ` and `hwy` variables and so the aesthetic mapping for the `x` and `y` axis' defined in the first line of ggplot code still apply.

```
ggplot(mpg, aes(displ, hwy))+
  geom_point(aes(color = class), shape = "square")+
  geom_point(size = 5,
             shape = 1,
             data = best_in_class)+
  geom_label_repel(aes(label = model),
                   data = best_in_class)+
  labs(x = "Engine size",
       y = "Highway fuel efficiency",
```

2

```
        color = "Class",
        title = "The most fuel efficient model in each class of car ")
```

The most fuel efficient model in each class of car

Using colored labels instead of legends

The same principle of creating a new data object that is reference inside a geometry can be applied to create a layer of labels over the data instead of having a legend. In the example below, we've removed the legend and instead mapped the labels to the class of car. The fact that the color aesthetic is mapped to the class of car in the first line of ggplot code means that it will be applied to each of the geometries that follow, including the label.

```
class_ave <- mpg %>%
  group_by(class) %>%
  summarise(
    displ = median(displ),
    hwy = median(hwy)
  )

mpg %>%
  ggplot(aes(displ, hwy, color = class))+
    geom_label_repel(aes(label = class),
```

```
                     data = class_ave,
                     size = 6,
                     label.size = 0,
                     segment.color = NA)+
    geom_jitter(size = 2,
                alpha = 0.3,
                shape = "square"
                )+
    theme(legend.position = "none")+
    labs(x = "Engine size",
        y = "Highway fuel efficiency",
        title = "Median fuel efficiency for each class of car")
```


Median fuel efficiency for each class of car
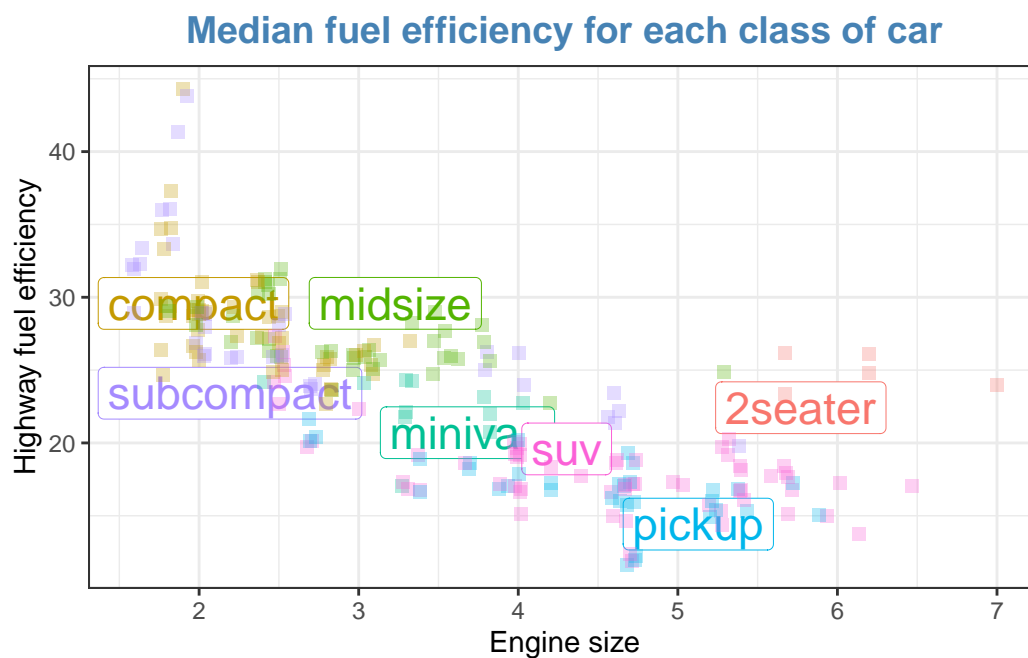
Learn More 365

Share this cheat sheet with others

Unlock a wealth of FREE resources TODAY!
Click on the links below to enrich your knowledge in these topics:

📚 Statistics & Research Methods Resource Library
📚 R Programming & Data Visualization Resource Library
📚 Discover the Public Health & Epidemiology Resource Library

Explore video lessons on **Statistics & Research Methods,
R Programming & Data Visualization
Public Health & Epidemiology,** and more
at Learn More 365!

Click **HERE** or scan the QR code below to start learning!