

The Grammar of Graphics (of ggplot2)

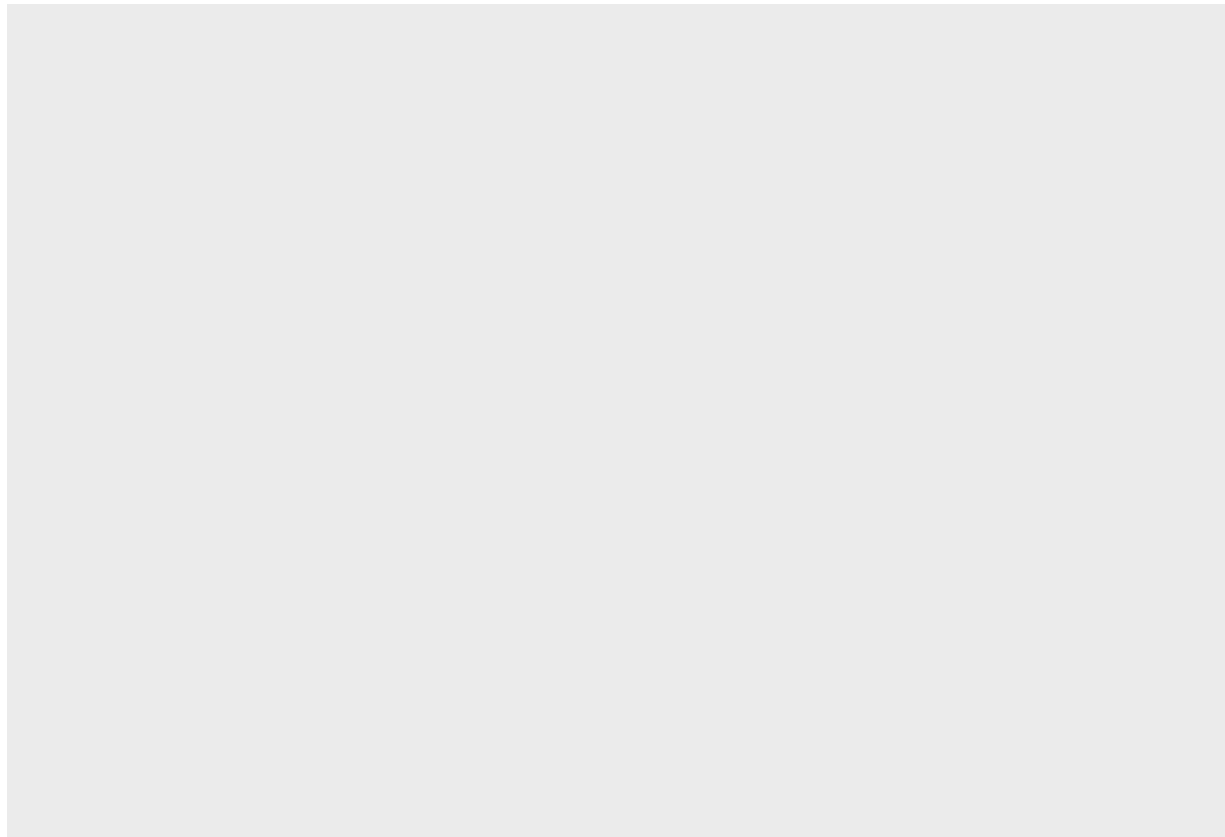
Lucas Mello Schnorr, Jean-Marc Vincent

February 14, 2017

This is a demonstration of how ggplot works.

An empty plot: understanding the layered nature of ggplot.

```
library(ggplot2);  
ggplot();
```



There is no data plot yet. Let's use the **pressure** data set:

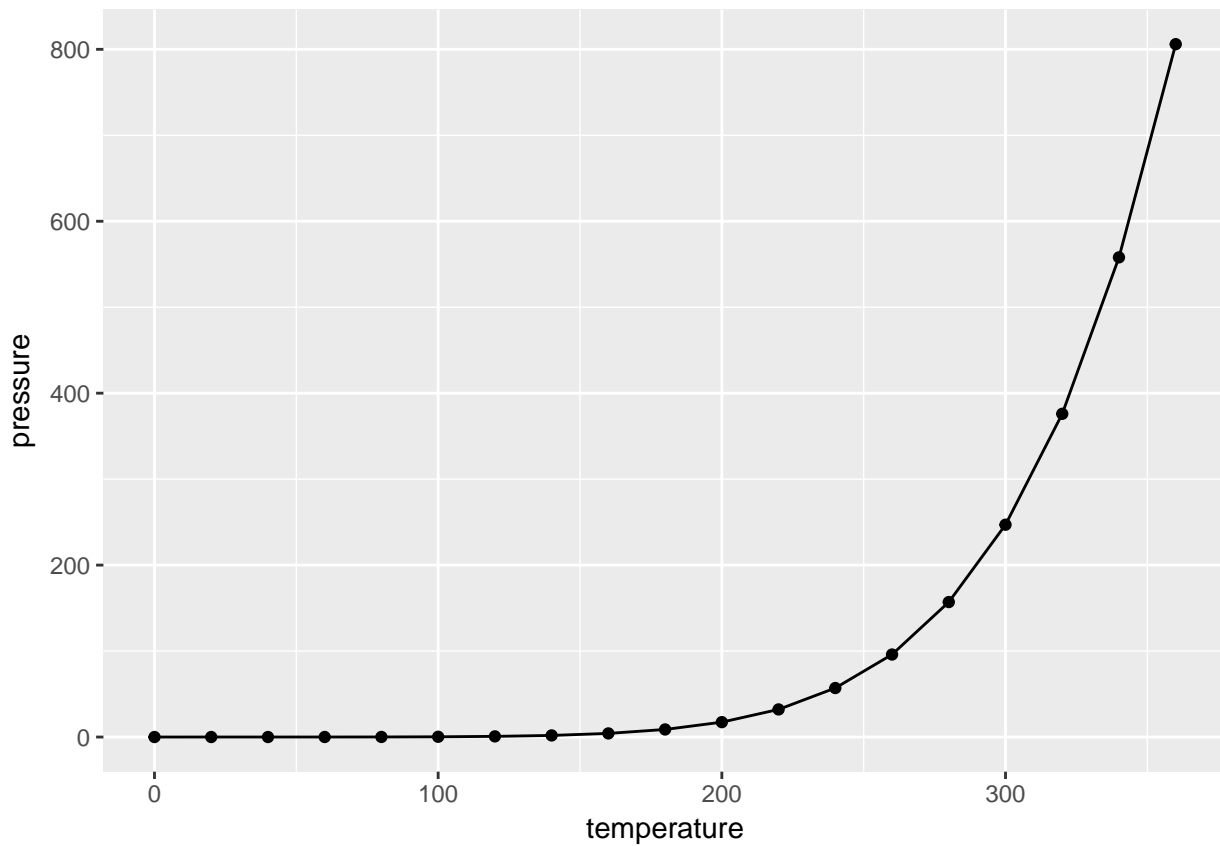
```
pressure;
```

```
##      temperature pressure  
## 1             0  0.0002  
## 2            20  0.0012  
## 3            40  0.0060  
## 4            60  0.0300  
## 5            80  0.0900  
## 6           100  0.2700  
## 7           120  0.7500  
## 8           140  1.8500  
## 9           160  4.2000  
## 10          180  8.8000
```

```
## 11      200  17.3000
## 12      220  32.1000
## 13      240  57.0000
## 14      260  96.0000
## 15      280 157.0000
## 16      300 247.0000
## 17      320 376.0000
## 18      340 558.0000
## 19      360 806.0000
```

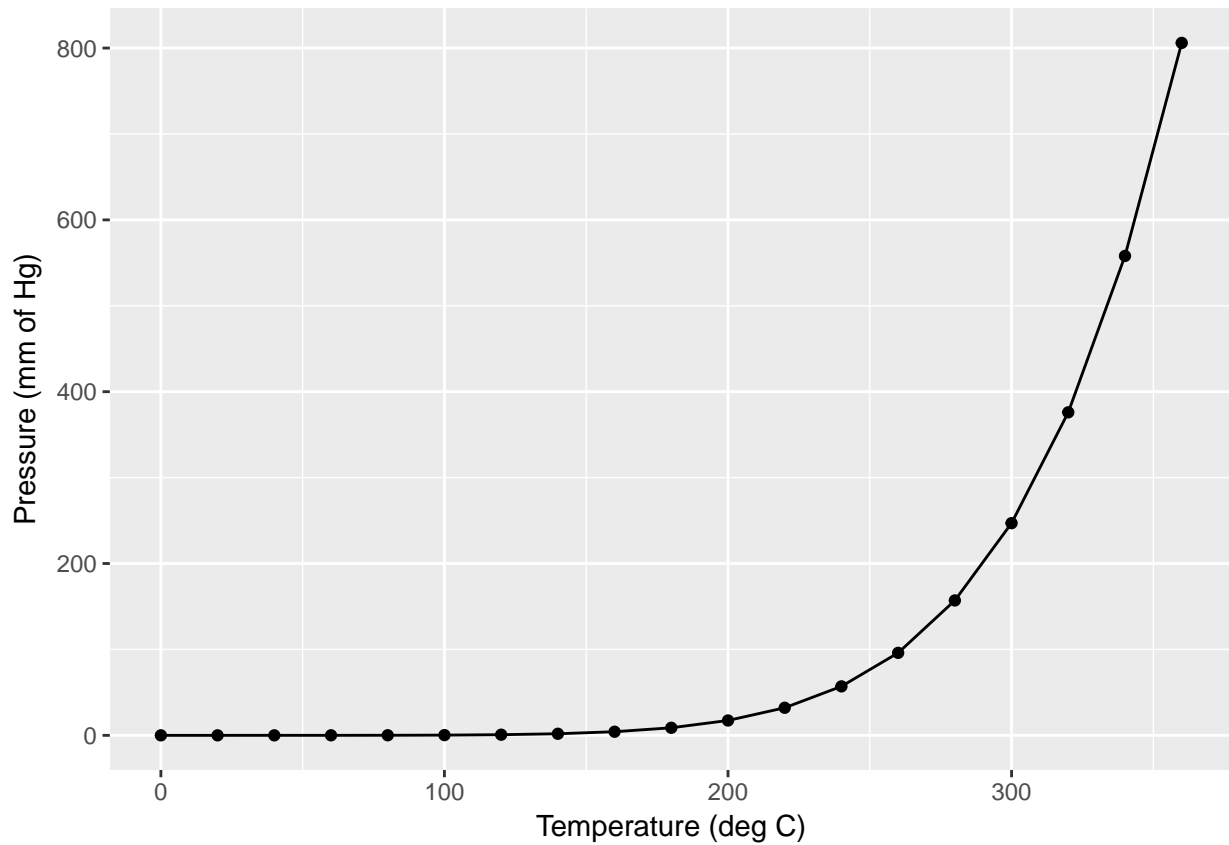
Let's plot the simplest case, just points and a line interpolating them.

```
ggplot(data=pressure, aes(x=temperature, y=pressure)) + geom_point() + geom_line();
```



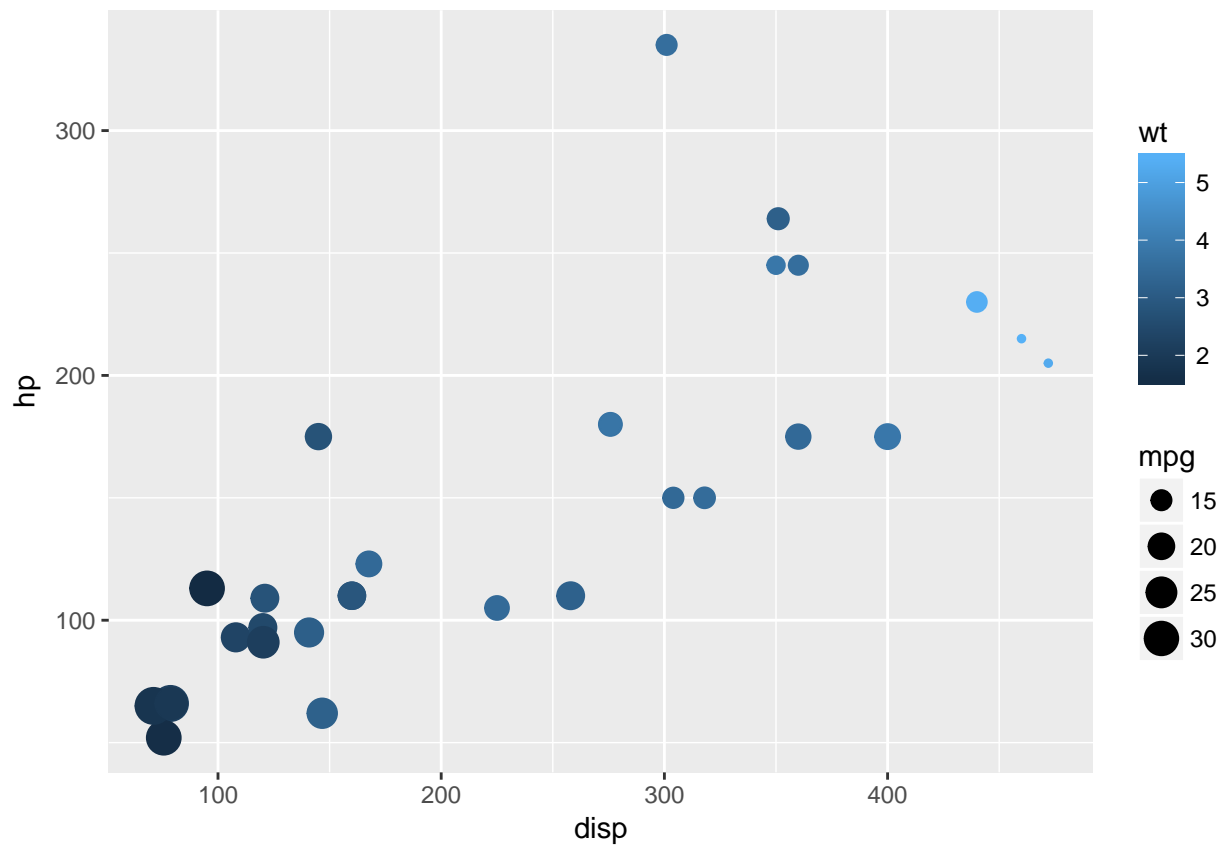
How to customize labels:

```
ggplot(data=pressure, aes(x=temperature, y=pressure)) +
  geom_point() +
  geom_line() +
  xlab("Temperature (deg C)") +
  ylab("Pressure (mm of Hg)");
```



We can associate the size and color of points to other variables.

```
ggplot(data=mtcars, aes(x=disp, y=hp, size=mpg, color=wt)) +  
  geom_point();
```



Let's see the list of all aesthetics accepted by `geom_point()`: http://docs.ggplot2.org/current/geom_point.html

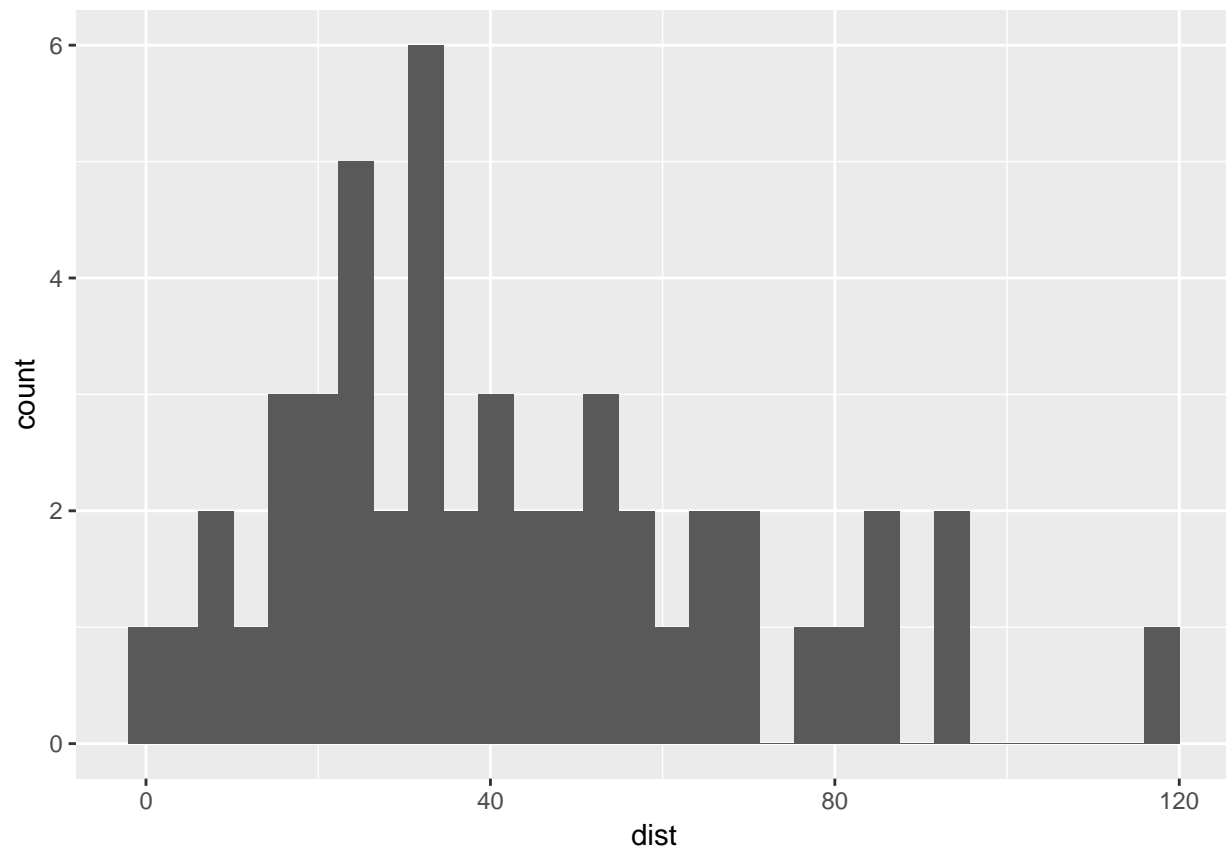
Let's plot a histogram:

```
str(cars);
```

```
## 'data.frame':  50 obs. of  2 variables:
## $ speed: num  4 4 7 7 8 9 10 10 10 11 ...
## $ dist : num  2 10 4 22 16 10 18 26 34 17 ...
```

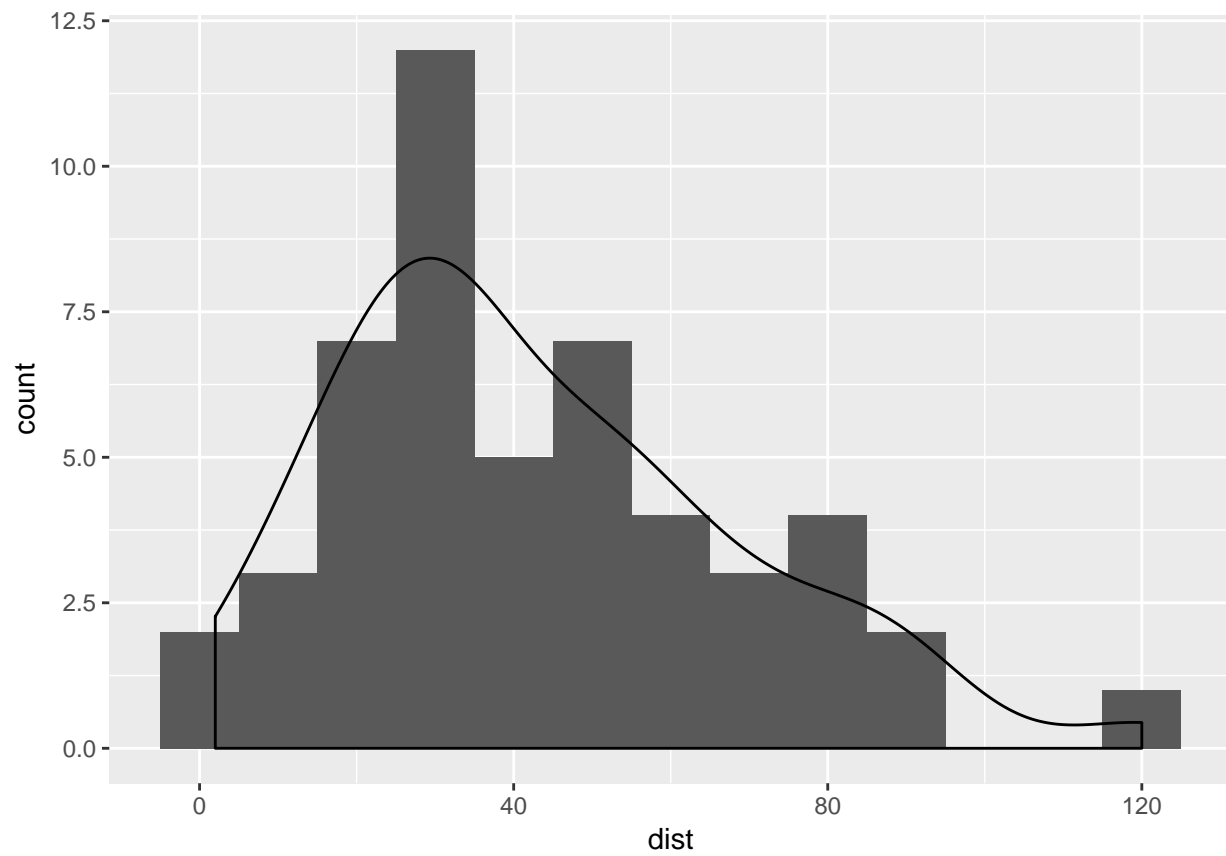
```
ggplot(data=cars) +
  geom_histogram(aes(x=dist));
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



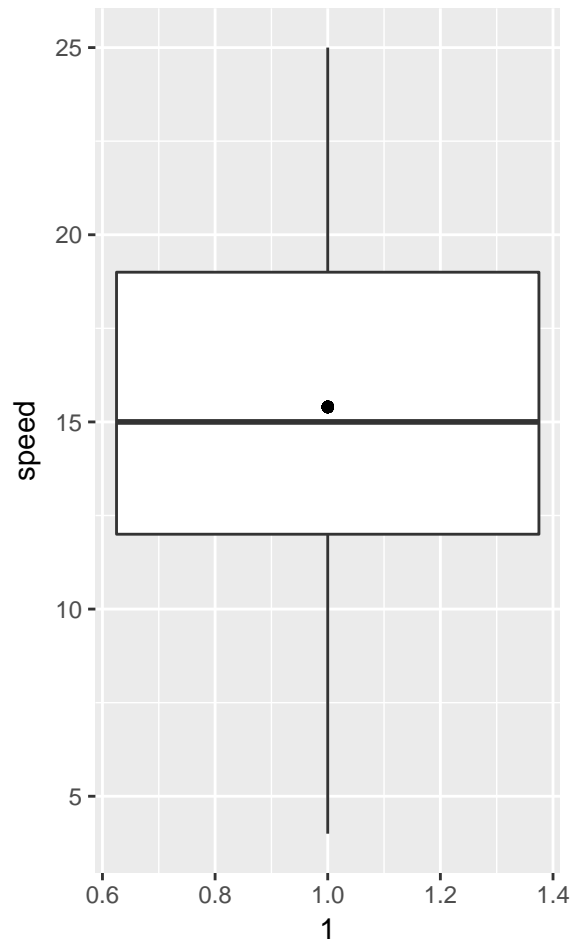
Merge a histogram and the corresponding Probability Density Function (PDF).

```
ggplot(cars, aes(x=dist)) +  
  geom_histogram(binwidth = 10)+  
  geom_density(aes(y=10 * ..count..));
```



Boxplot of speed, using a point to represent the mean value.

```
ggplot(cars, aes(x=1, y=speed)) + geom_boxplot() + geom_point(y=mean(cars$speed));
```



Using facets to separate data.

```
ggplot(mtcars, aes(x=hp, y=mpg)) + facet_wrap(~gear) + geom_point();
```

