Data Visualisation

Bibek Sapkota

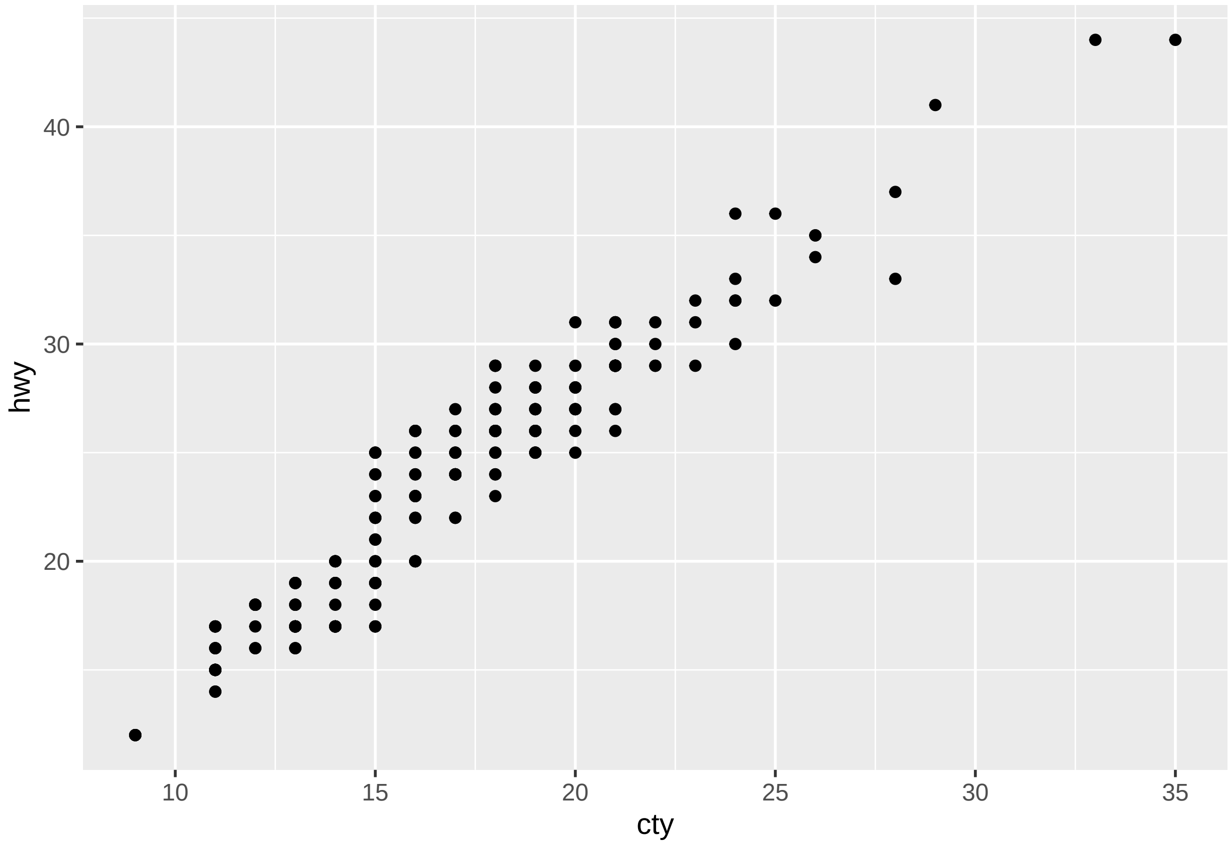
2024-08-11

#Importing library

```
library(ggplot2)
```

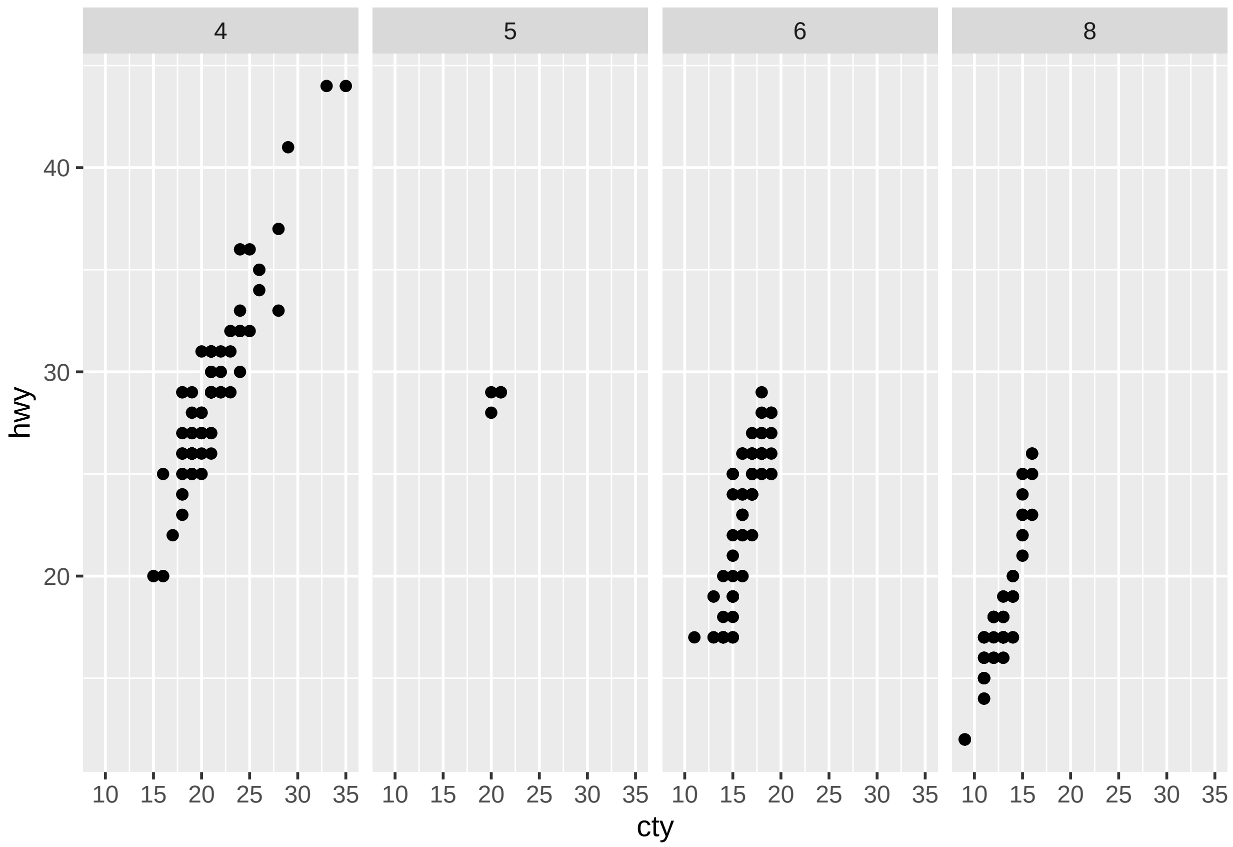
Facet grid

```
ggplot(data = mpg, aes(cty, hwy)) + geom_point() + facet_grid(. ~ .)
```



Interpretation: The above code shows a scatter plot of highway (hwy) versus city (cty) miles per gallon, with data points forming a positively correlated pattern. The facet_grid(. ~ .) command in the code is used, but since both sides are set to ., no actual faceting occurs, resulting in a single plot without any subgrouping.

```
ggplot(data = mpg, aes(cty, hwy)) + geom_point() + facet_grid(. ~ cyl)
```

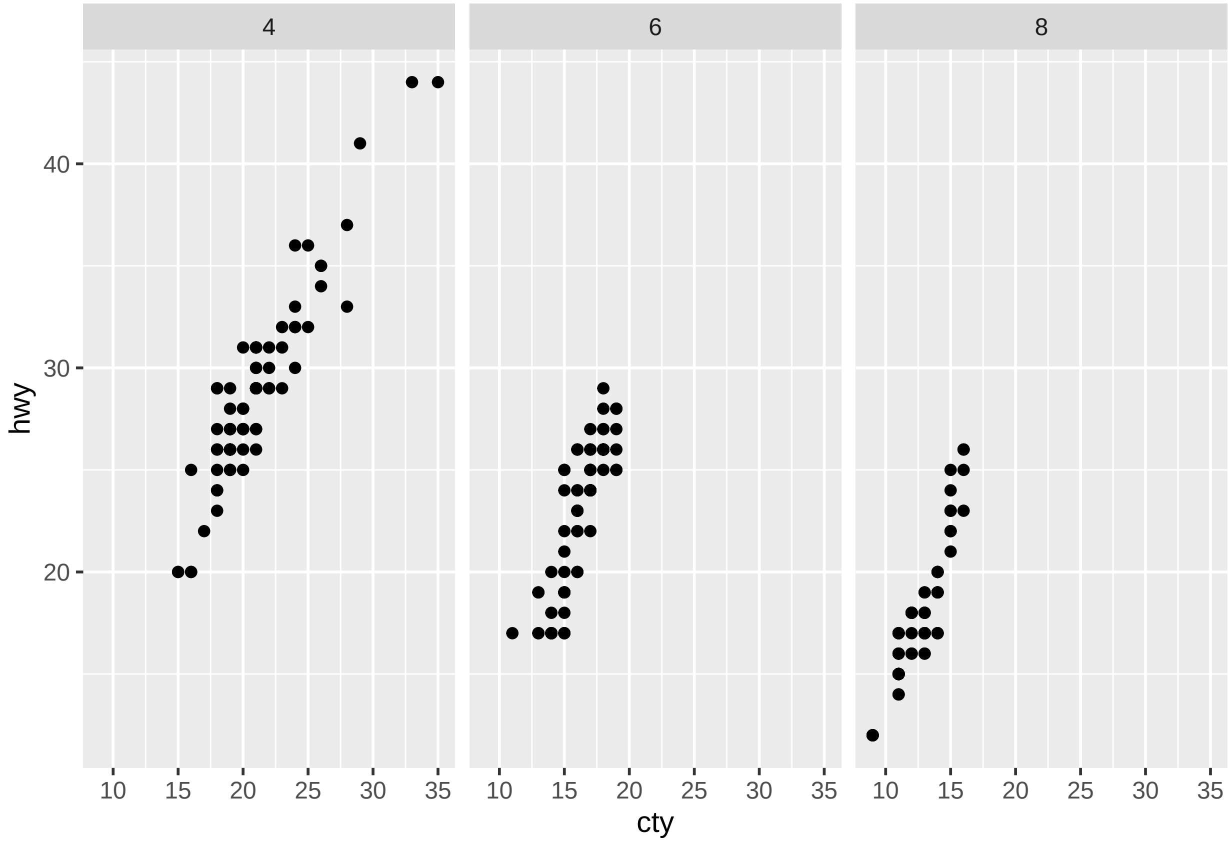


Interpretation: The above code shows a faceted scatter plot of highway (hwy) versus city (cty) miles per gallon, divided by the number of cylinders (cyl) in the vehicles. Each panel represents a different cylinder count (4, 5, 6, 8), revealing how fuel efficiency varies across vehicles with different engine sizes. The pattern shows that vehicles with fewer cylinders tend to have higher fuel efficiency.

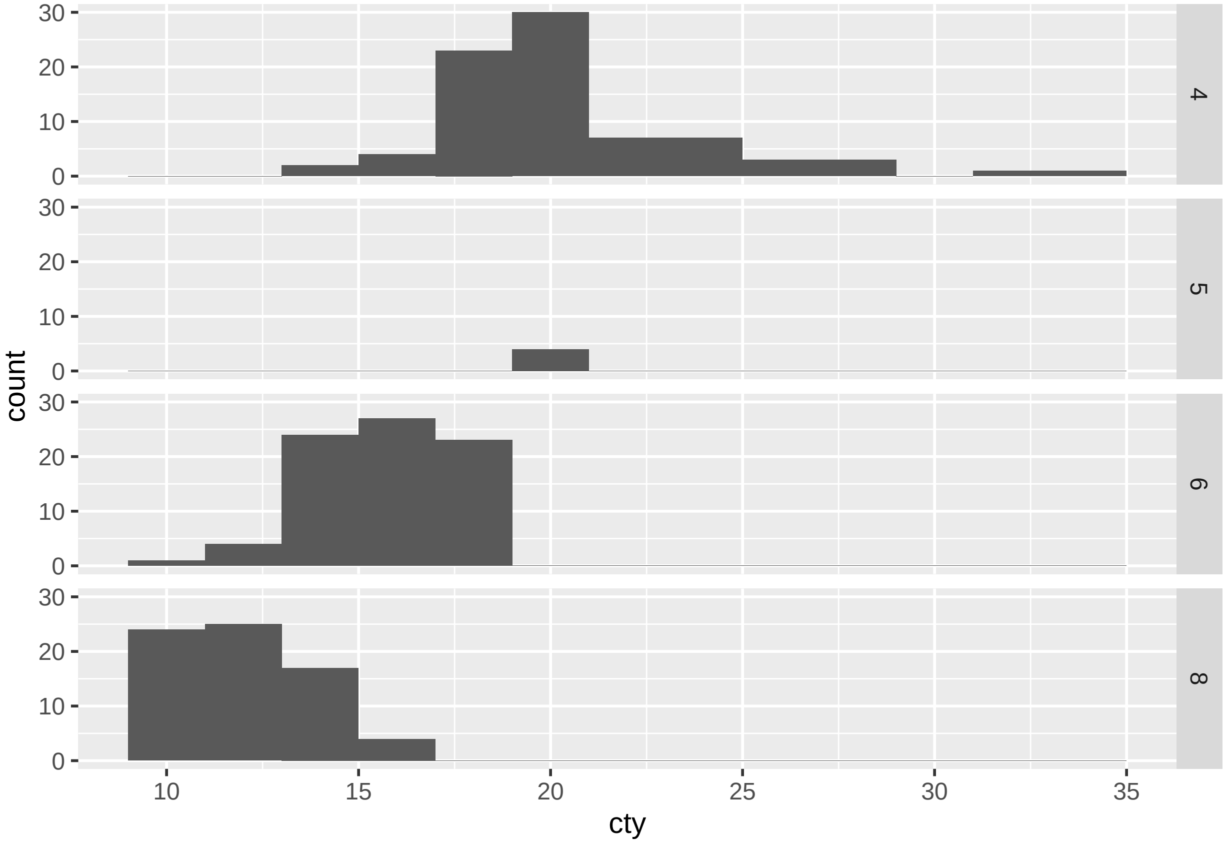
Task: The below code also shows the same output as above but it has plot all the cyl except 5cyl .

```
mpg2 <- subset(mpg, cyl != 5)

ggplot(data = mpg2, aes(cty, hwy)) + geom_point() + facet_grid(. ~ cyl)
```

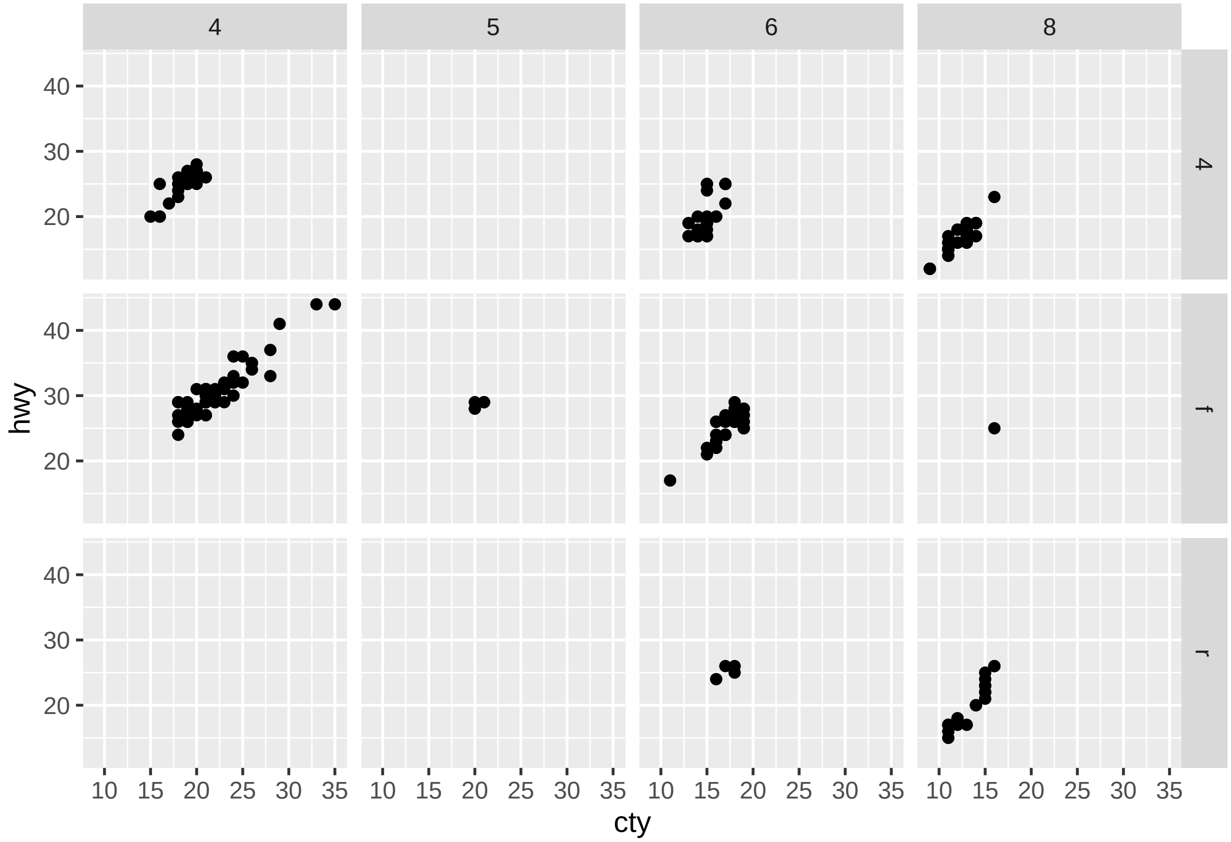


```
ggplot(data = mpg, aes(cty)) + geom_histogram(binwidth = 2) + facet_grid(cyl ~ .)
```



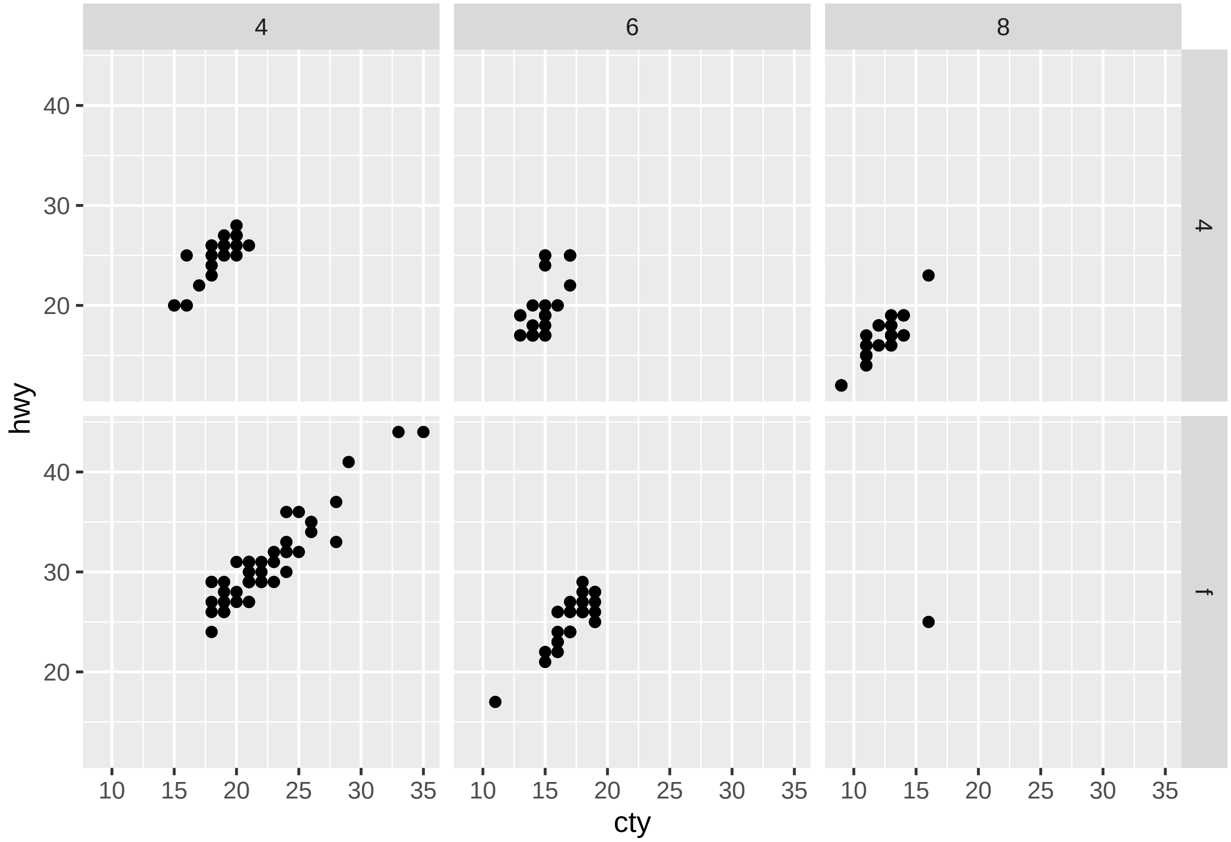
Interpretation: The above code shows histograms of city miles per gallon (cty), separated by the number of cylinders (cyl) in the vehicles using facet_grid(cyl ~ .). Each horizontal panel corresponds to a different cylinder count (4, 5, 6, 8). The histograms show how the distribution of city fuel efficiency varies across different engine sizes, with higher cylinder counts generally associated with lower fuel efficiency.

```
ggplot(data = mpg, aes(cty, hwy)) + geom_point() + facet_grid(drv ~ cyl)
```



Interpretation: The above graph filters the mpg dataset to exclude cars with 5 cylinders and to include only those with four-wheel or front-wheel drive. It then creates a scatter plot of city vs. highway miles per gallon (cty vs. hwy), with the plot divided into a grid based on drive type (drv) and cylinder count (cyl).

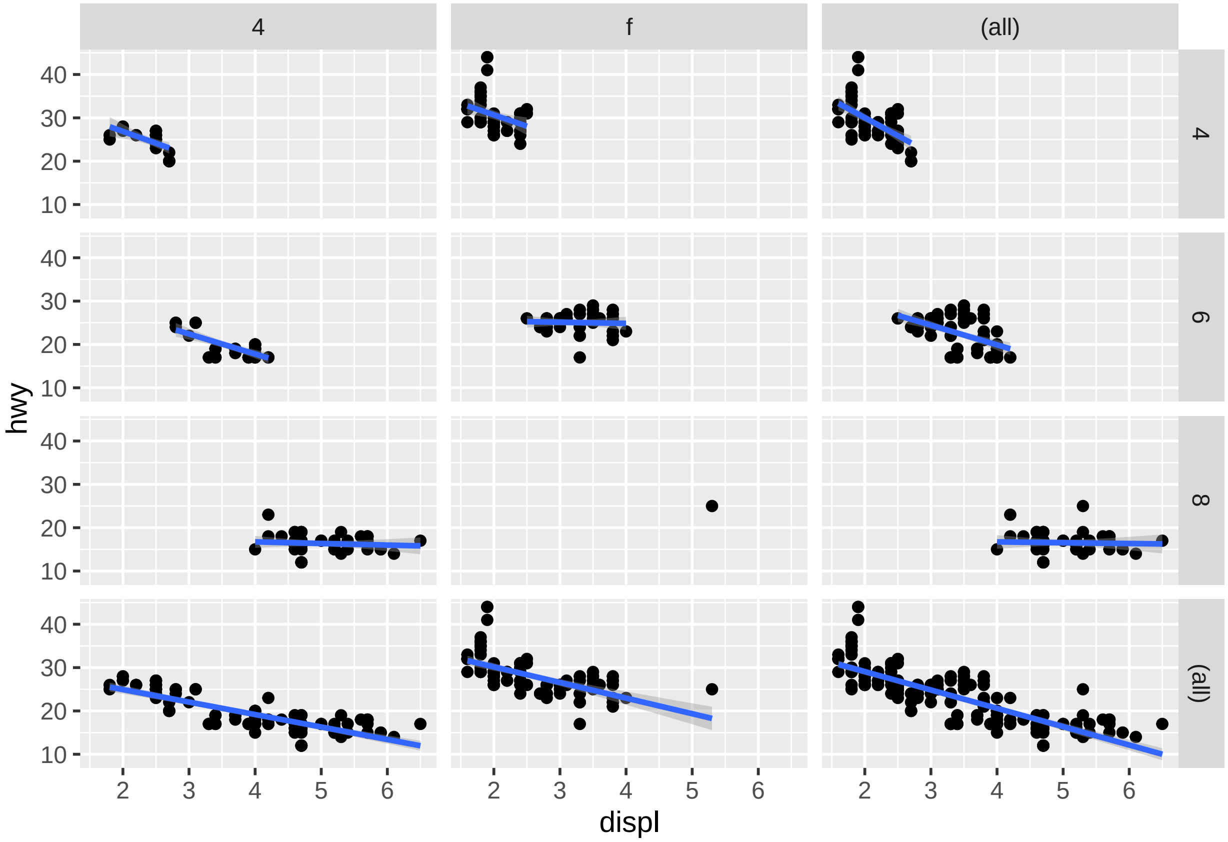
```
mpg2 <- subset(mpg, cyl != 5 & drv %in% c("4", "f"))
ggplot(data = mpg2, aes(cty, hwy)) + geom_point() + facet_grid(drv ~ cyl)
```



Margins

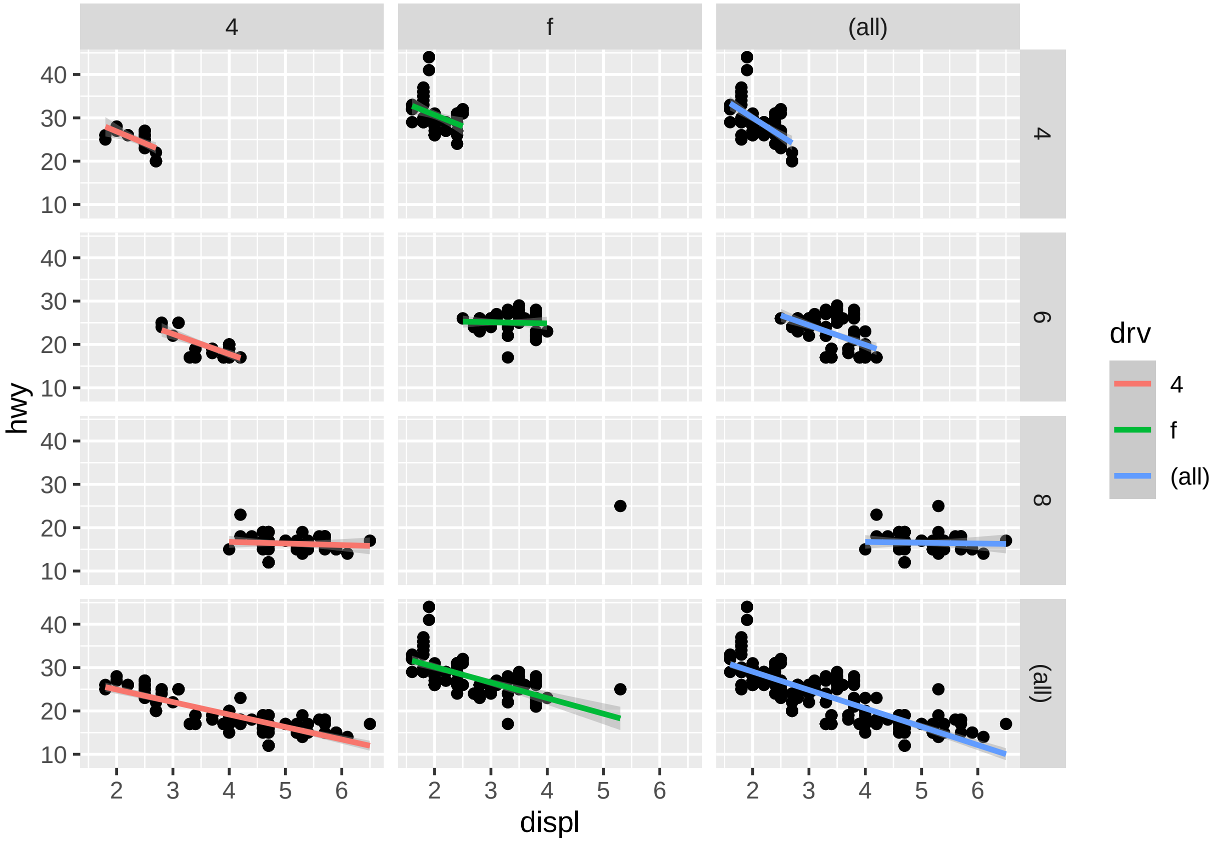
```
mpg2 <- subset(mpg, cyl != 5 & drv %in% c("4", "f"))
p <- ggplot(data = mpg2, aes(displ, hwy)) + geom_point() + geom_smooth(method = "lm")
p + facet_grid(cyl ~ drv, margins = T)
```

```
## ‘geom_smooth()‘ using formula = ’y ~ x’
```



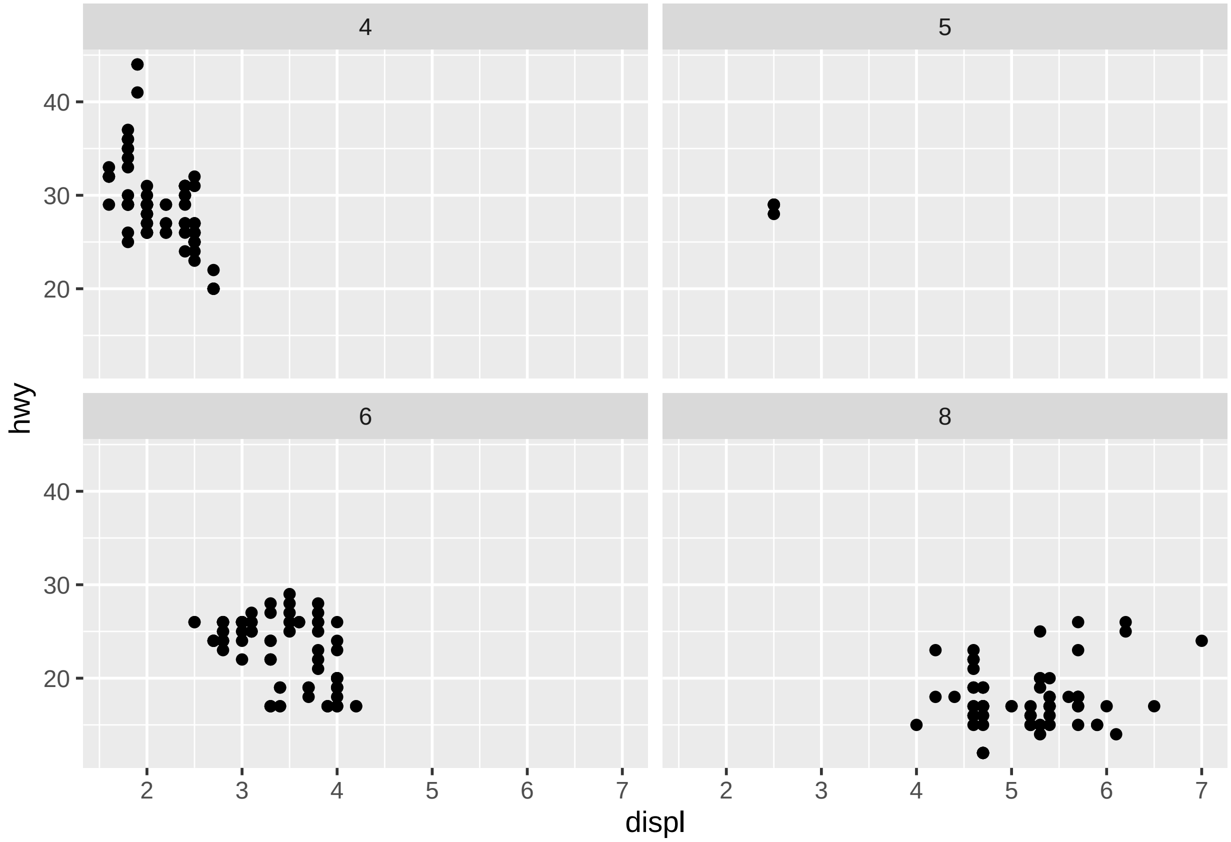
```r
mpg2 <- subset(mpg, cyl != 5 & drv %in% c("4", "f"))
p <- ggplot(data = mpg2, aes(displ, hwy)) + geom_point() + geom_smooth(aes(colour = drv), method
= "lm")
p + facet_grid(cyl ~ drv, margins = T)
```

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

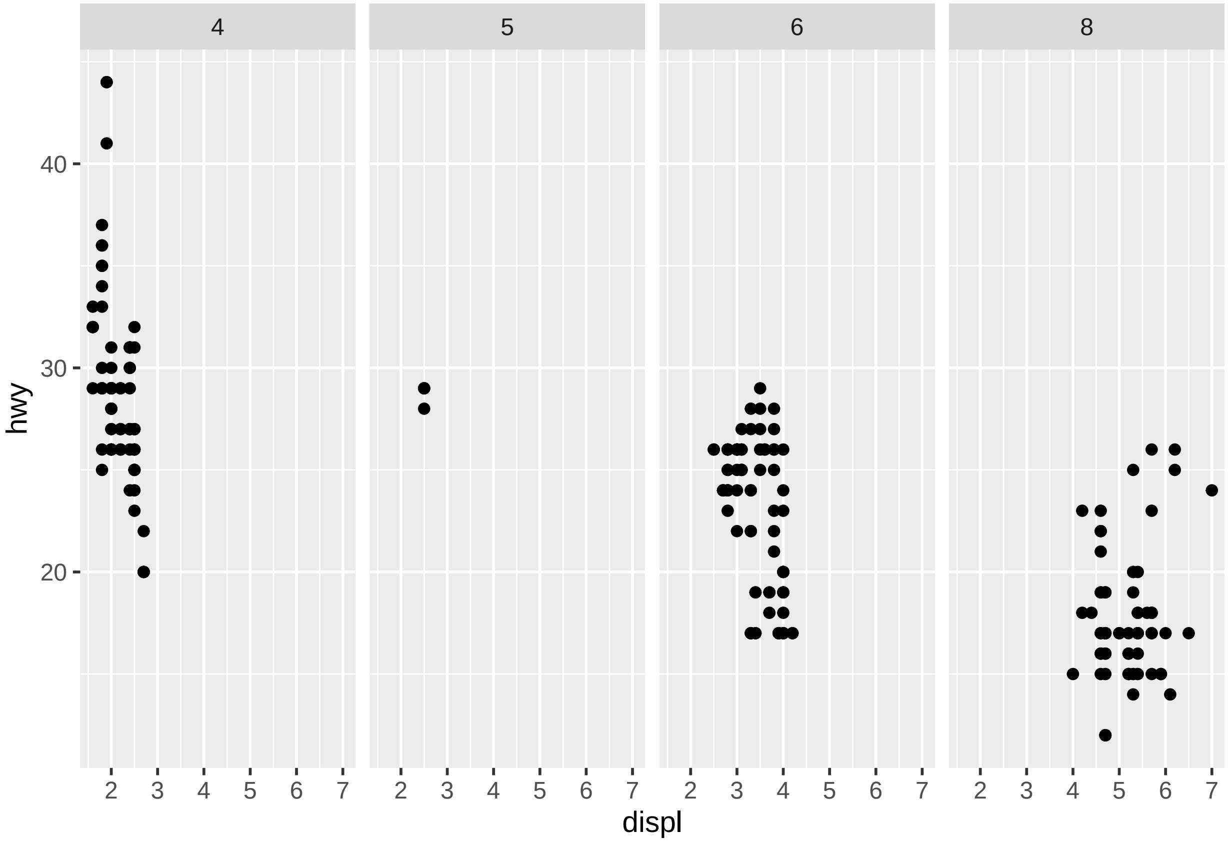


#Facet wrap

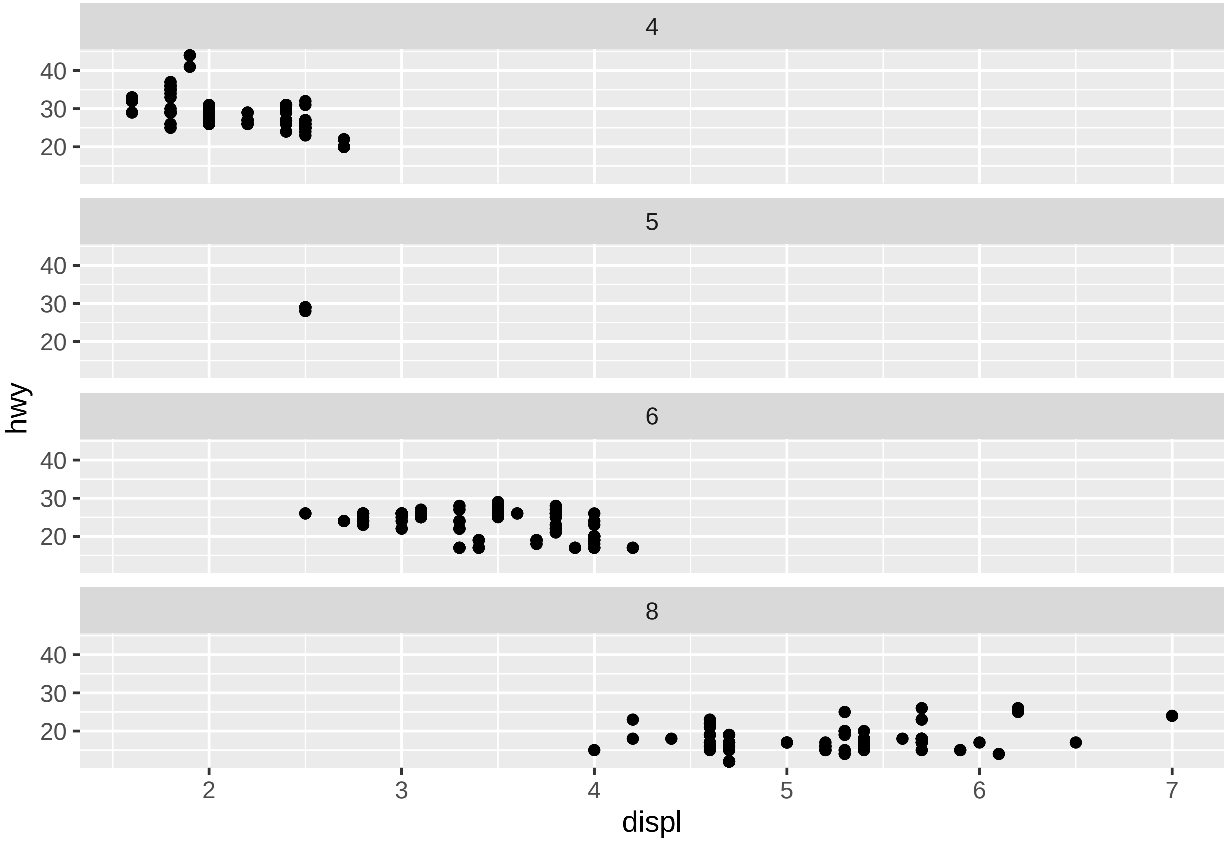
```
p <- ggplot(data = mpg, aes(x = displ, y = hwy)) + geom_point()
p + facet_wrap(~cyl)
```



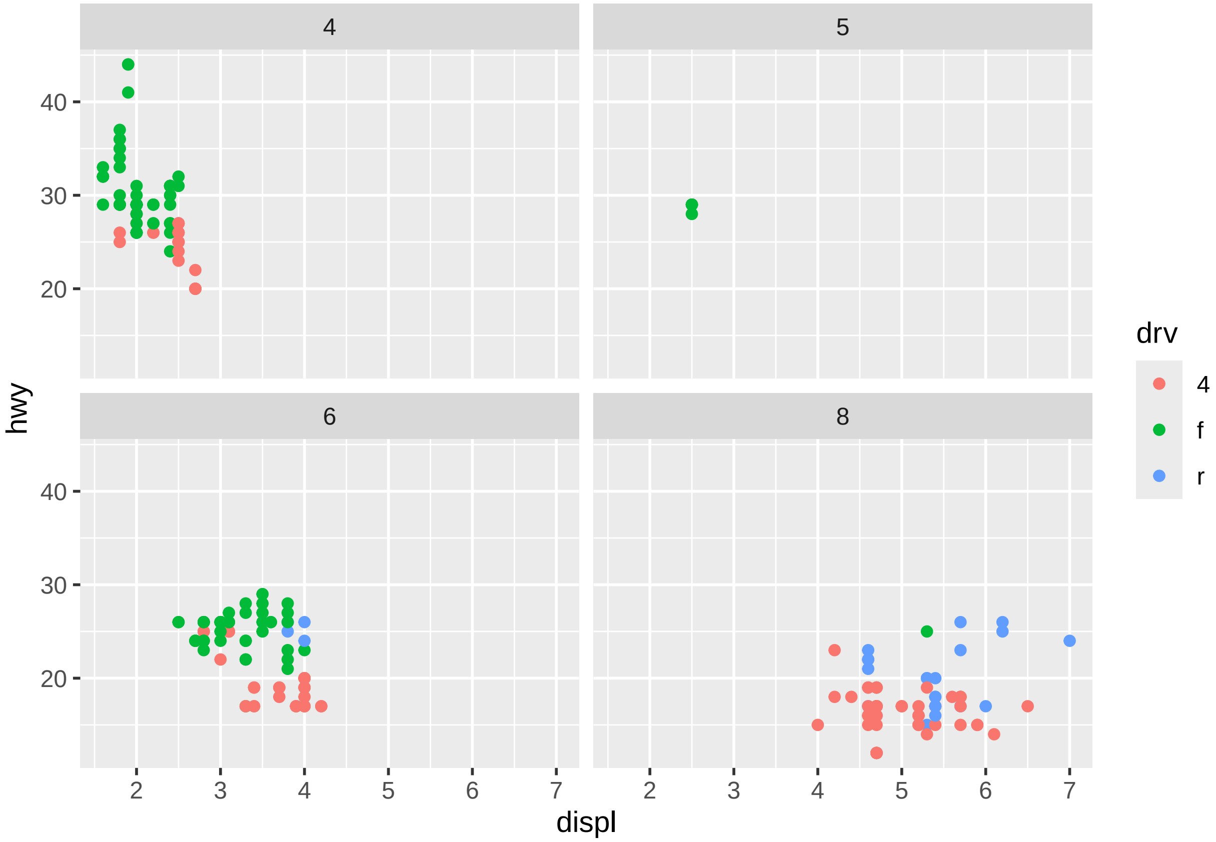
```
p + facet_wrap(~cyl, nrow = 1)
```



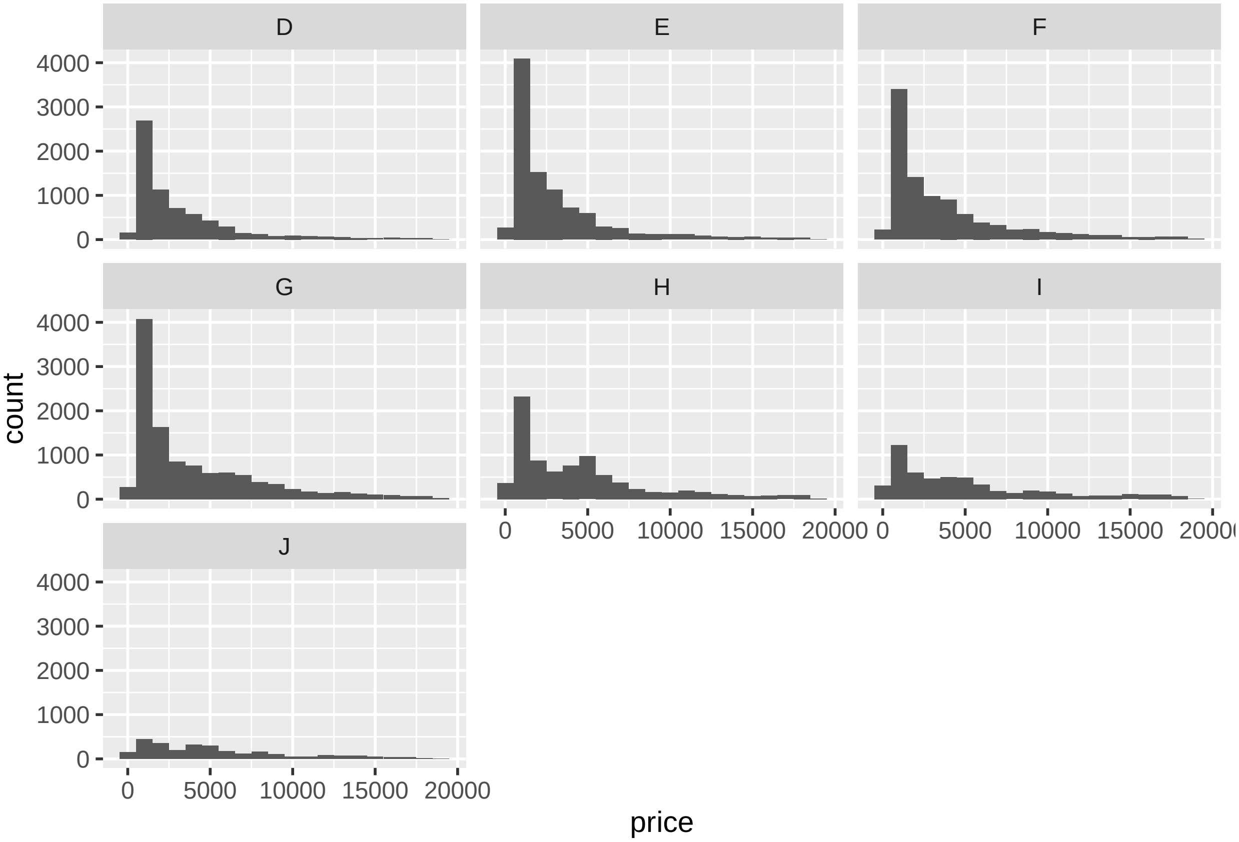
```r
p + facet_wrap(~cyl, ncol = 1)
```



```
p <- ggplot(data = mpg, aes(x = displ, y = hwy, color = drv)) + geom_point()
p + facet_wrap(~cyl)
```

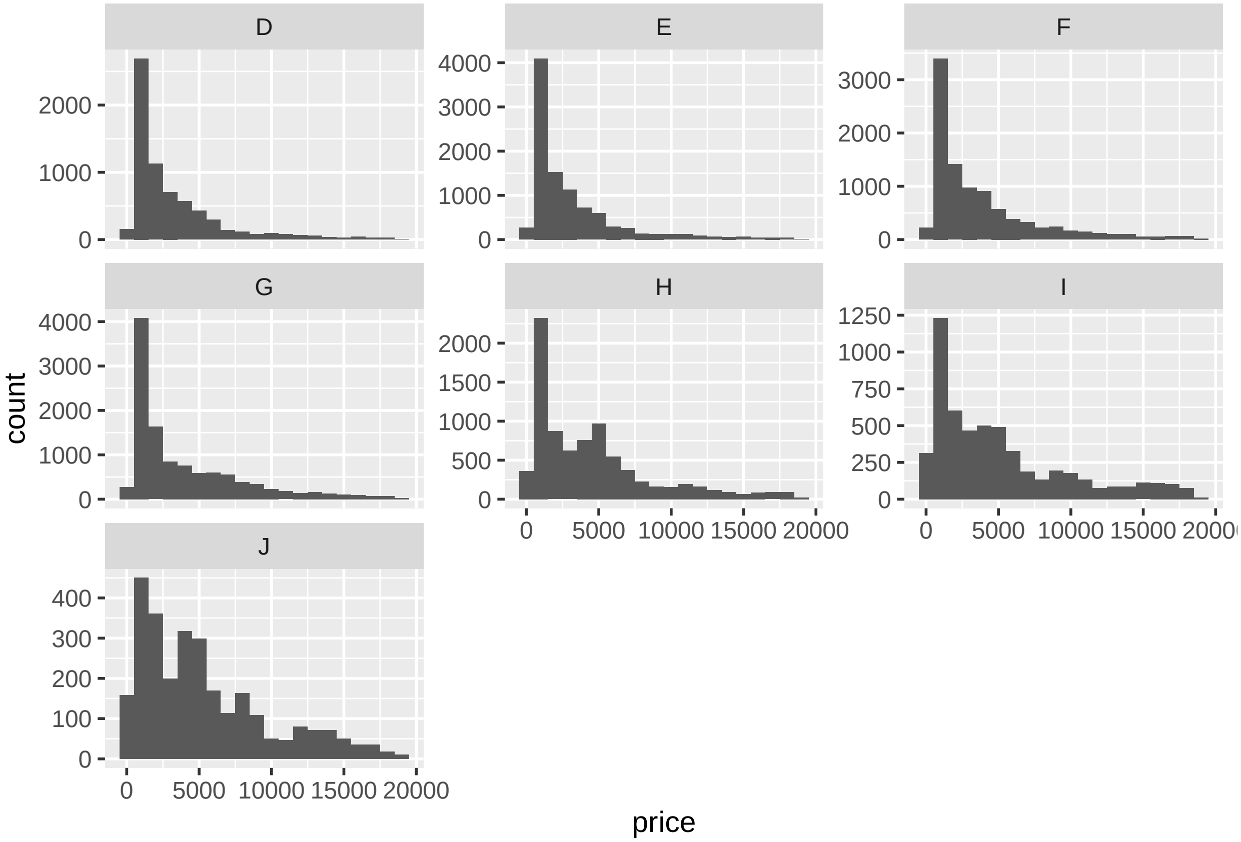


```r
p <- ggplot(data = diamonds, aes(x = price)) + geom_histogram(binwidth = 1000)
p + facet_wrap(~color)
```

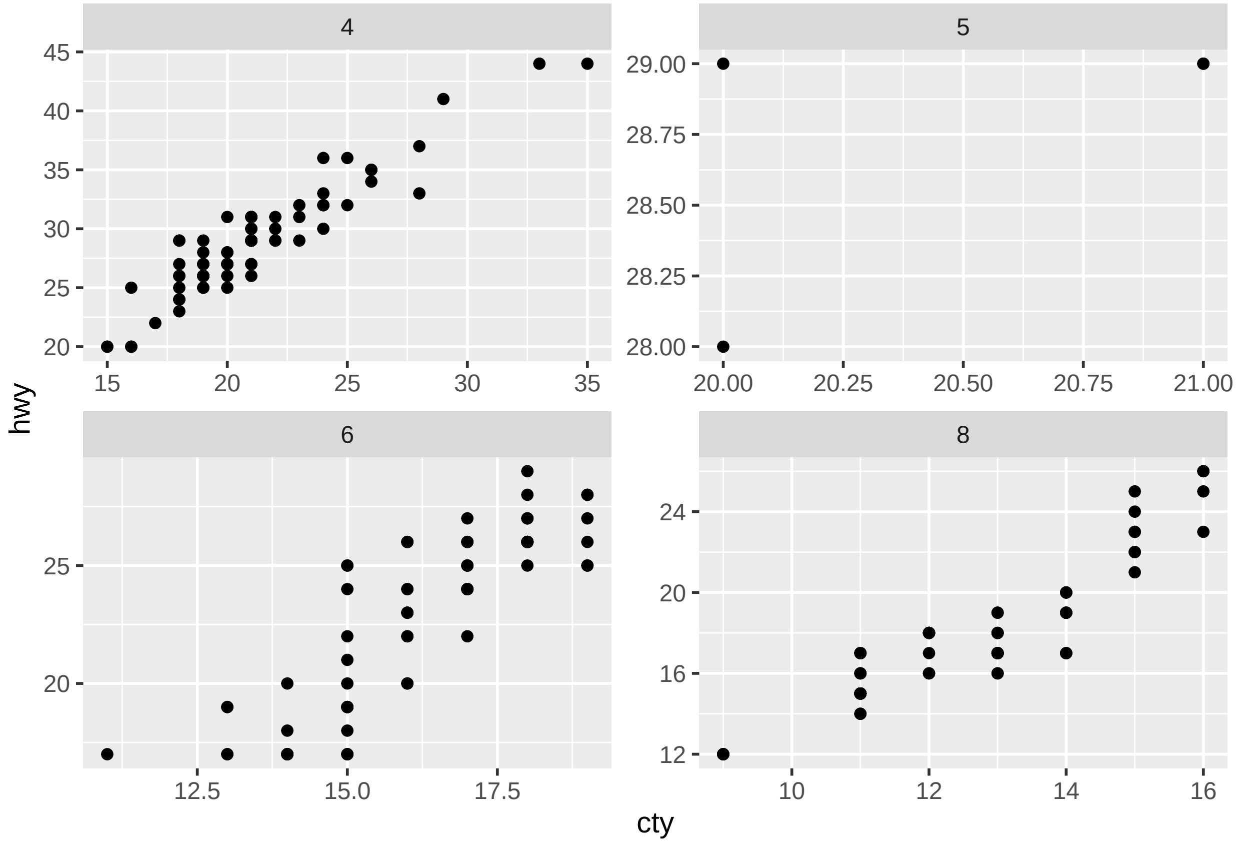


Position Sclaes

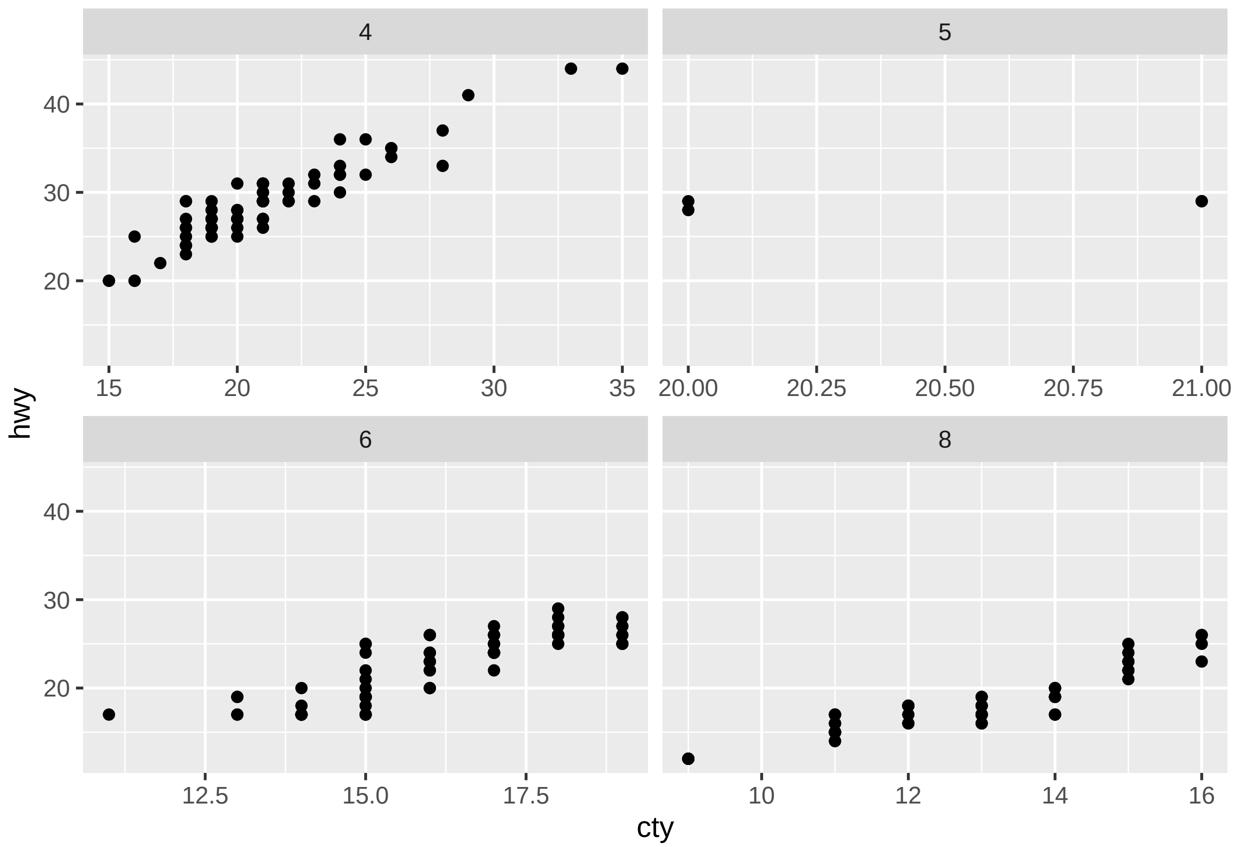
```r
p + facet_wrap(~color, scales = "free_y")
```



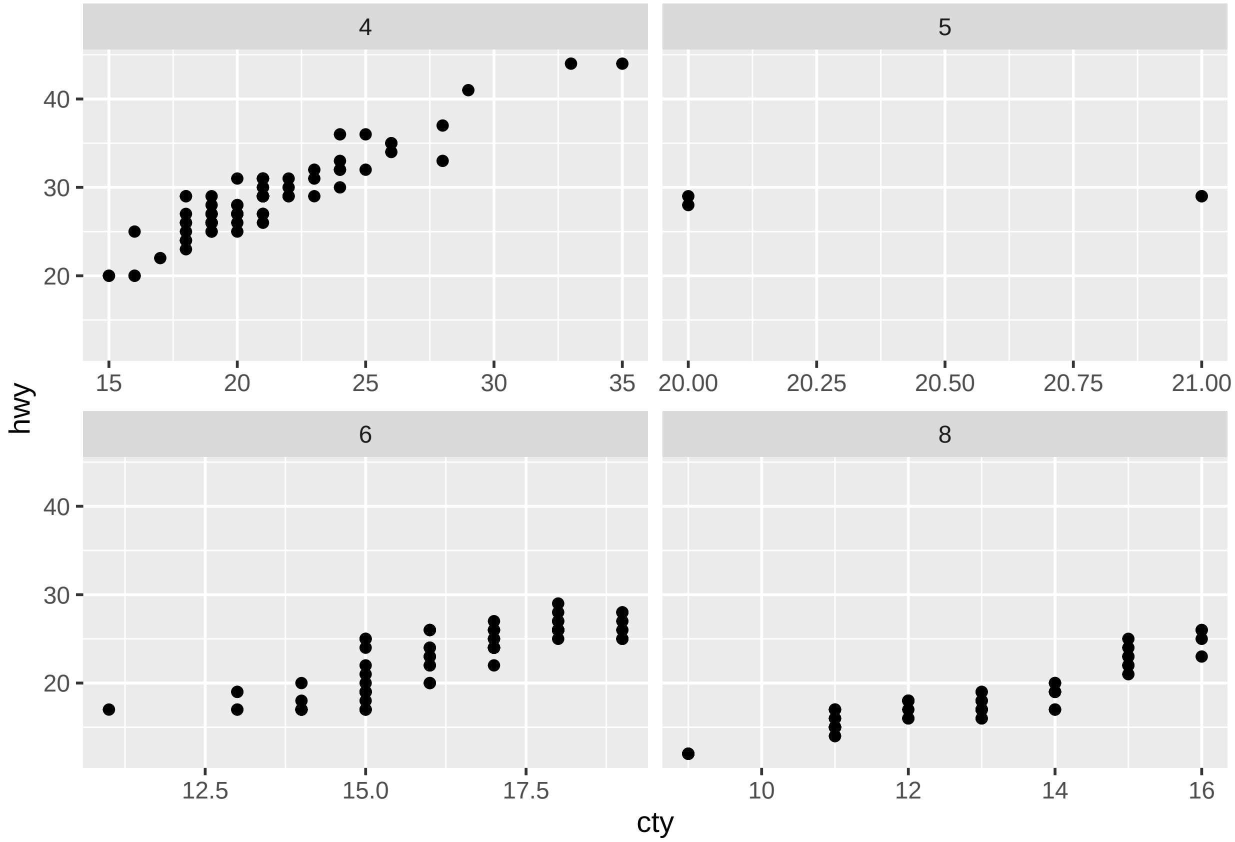
```
ggplot(data = mpg, aes(cty, hwy)) + geom_point() + facet_wrap(~ cyl, scales = "free")
```



```
ggplot(data = mpg, aes(cty, hwy)) + geom_point() + facet_wrap(~ cyl, scales = "free_x")
```

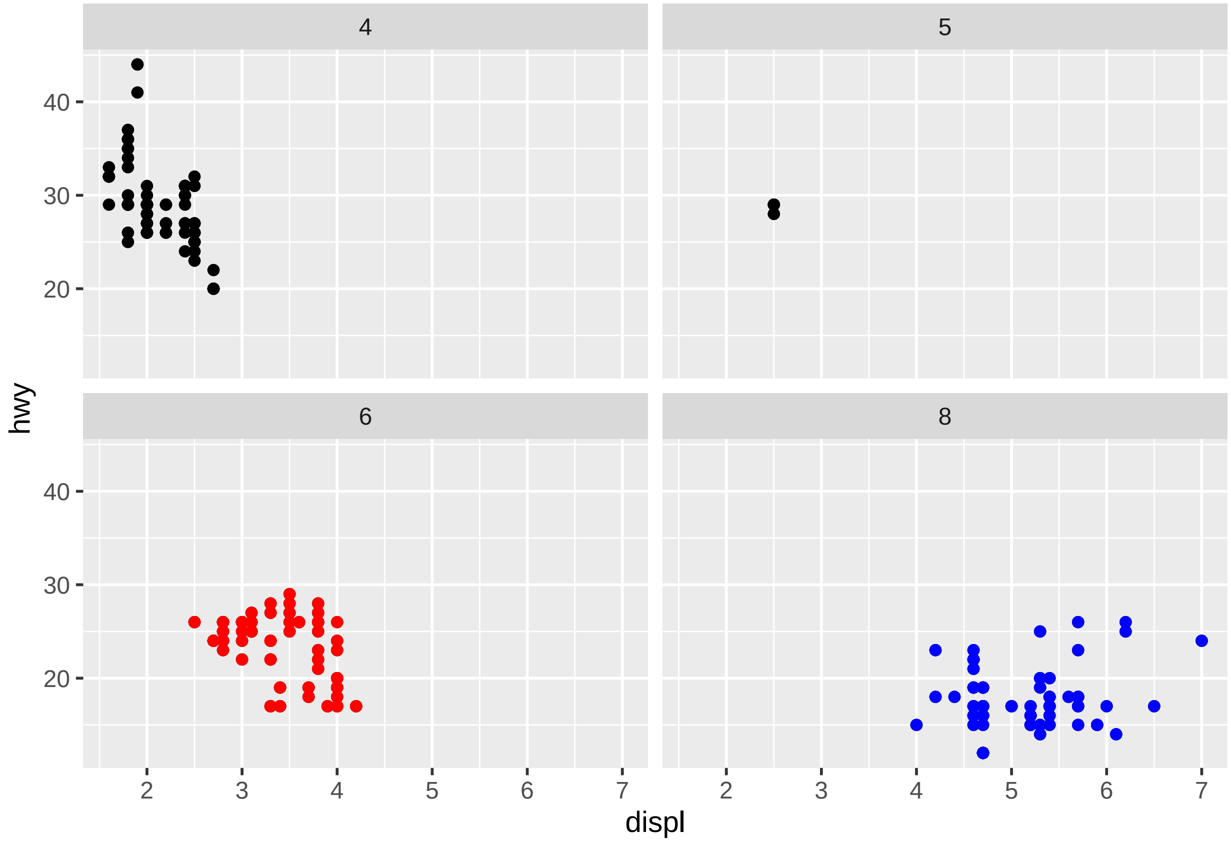


```
ggplot(data = mpg, aes(cty, hwy)) + geom_point() + facet_wrap(~ cyl, scales = "free_x")
```



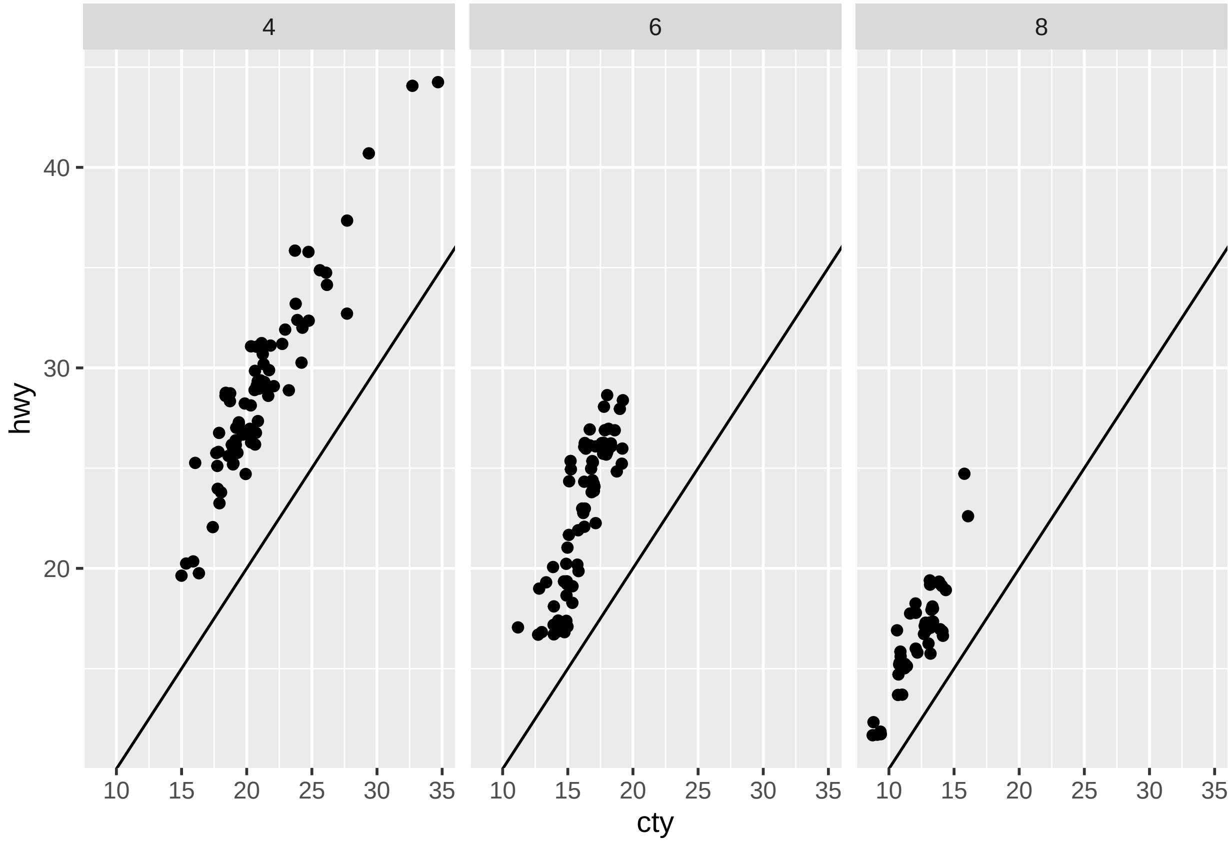
#Decorating factes

```
p <- ggplot(data = mpg, aes(x = displ, y = hwy)) + geom_point()
q <- p + facet_wrap(~cyl)
cycl6 <- subset(mpg, cyl == 6)
#Extra
cycl8 <- subset(mpg, cyl == 8)
q + geom_point(data = cycl6, color = "red")+
    geom_point(data = cycl8, color = "blue")
```

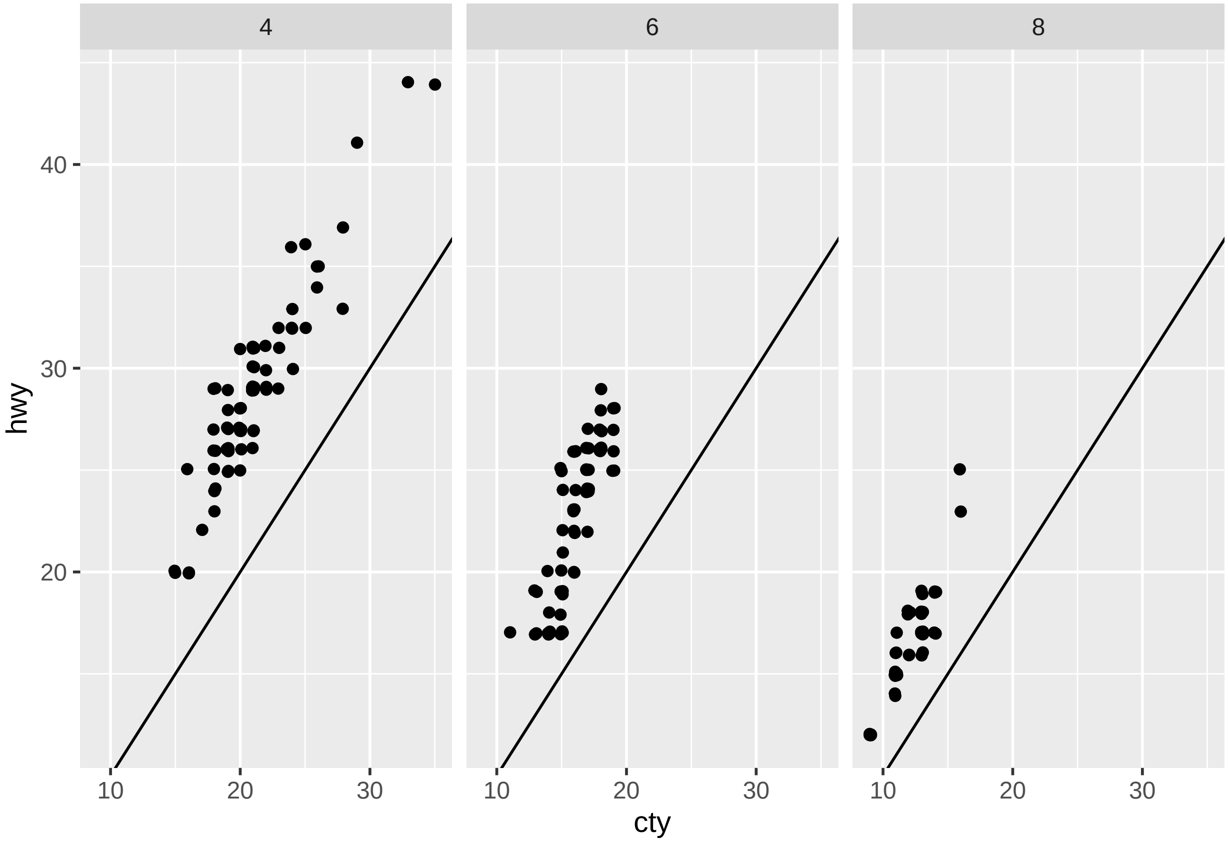


Faceting with abline and jitter

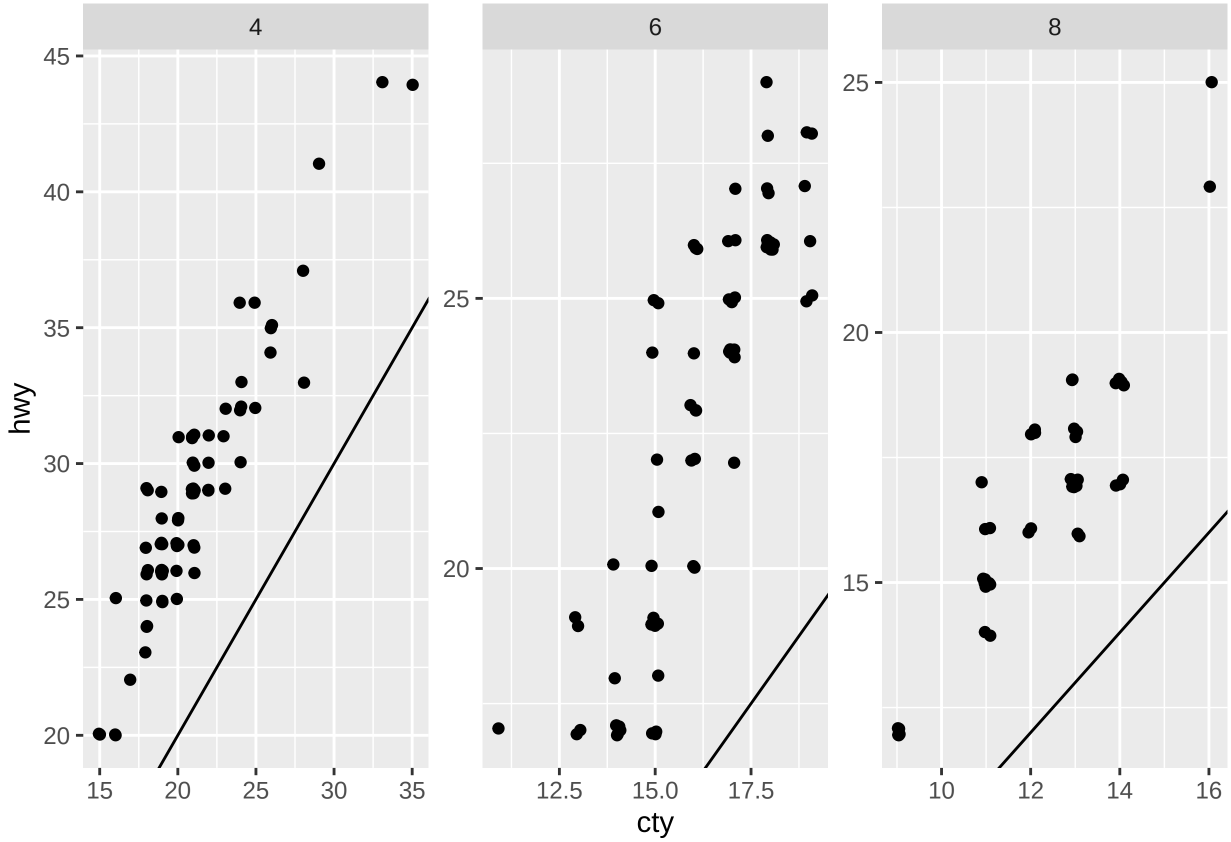
```
p <- ggplot(mpg2, aes(cty, hwy)) +
 geom_abline() +
 geom_jitter()
p + facet_wrap(~cyl)
```



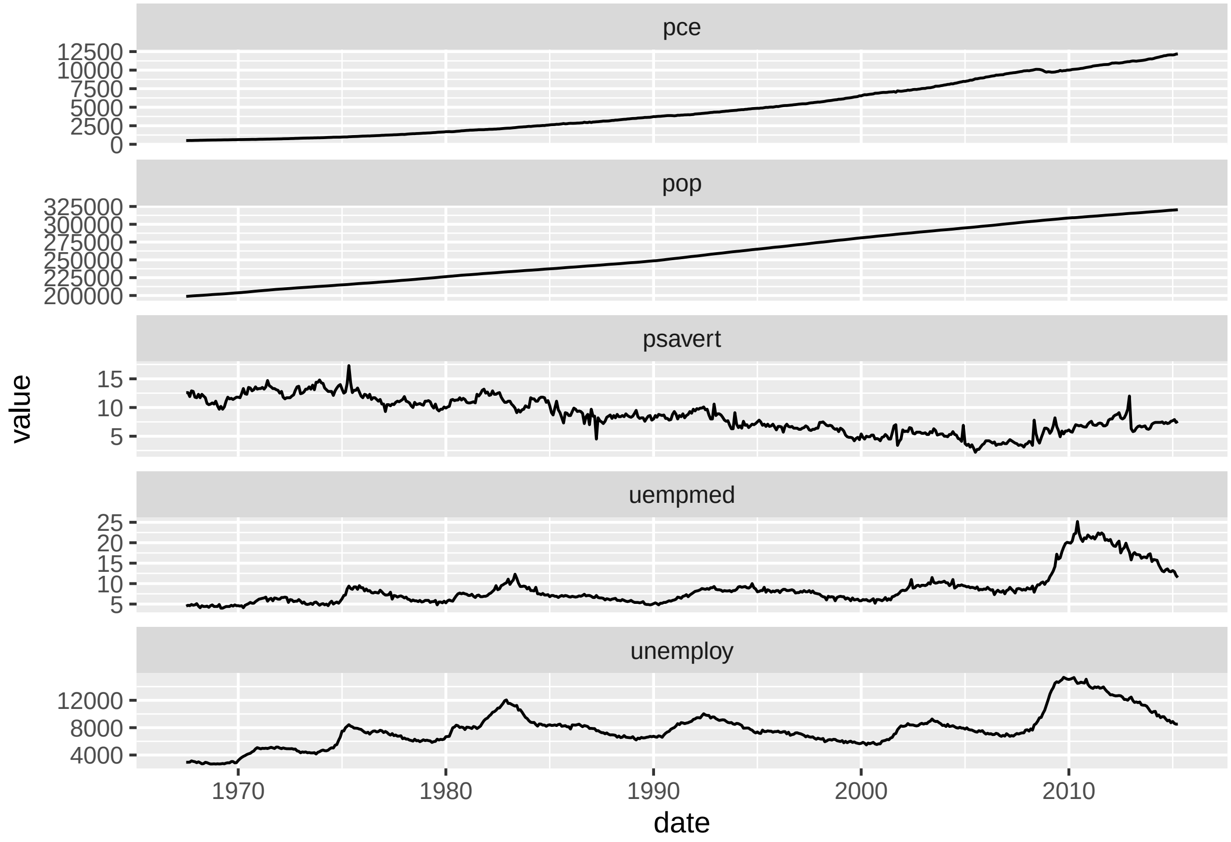
```
p <- ggplot(mpg2, aes(cty, hwy)) +
 geom_abline() +
 geom_jitter(width = 0.1, height = 0.1)
p + facet_wrap(~cyl)
```



```r
p + facet_wrap(~cyl, scales = "free")
```



```
ggplot(economics_long, aes(date, value)) +
 geom_line() +
 facet_wrap(~variable, scales = "free_y", ncol = 1)
```

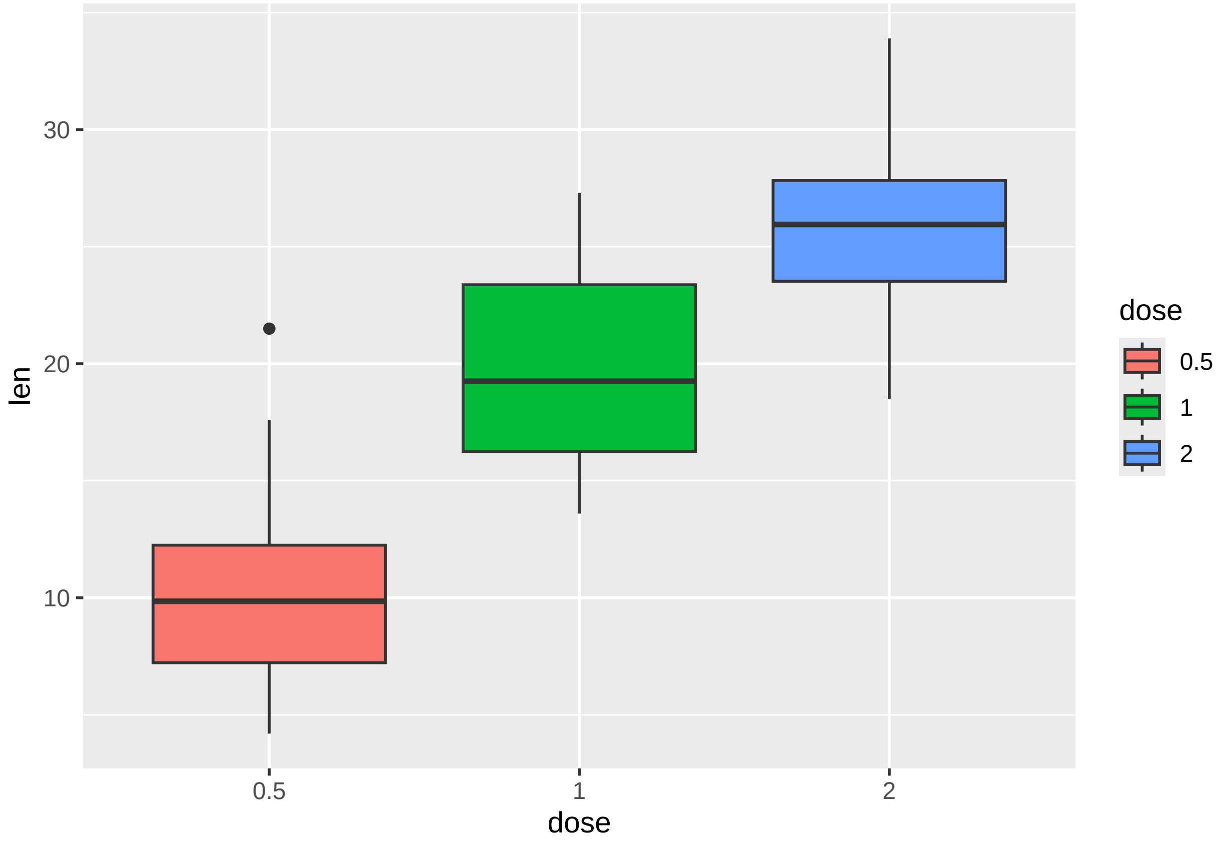


Facets with boxplot

```
ToothGrowth$dose <- as.factor(ToothGrowth$dose)
df <- ToothGrowth
head(df)
```

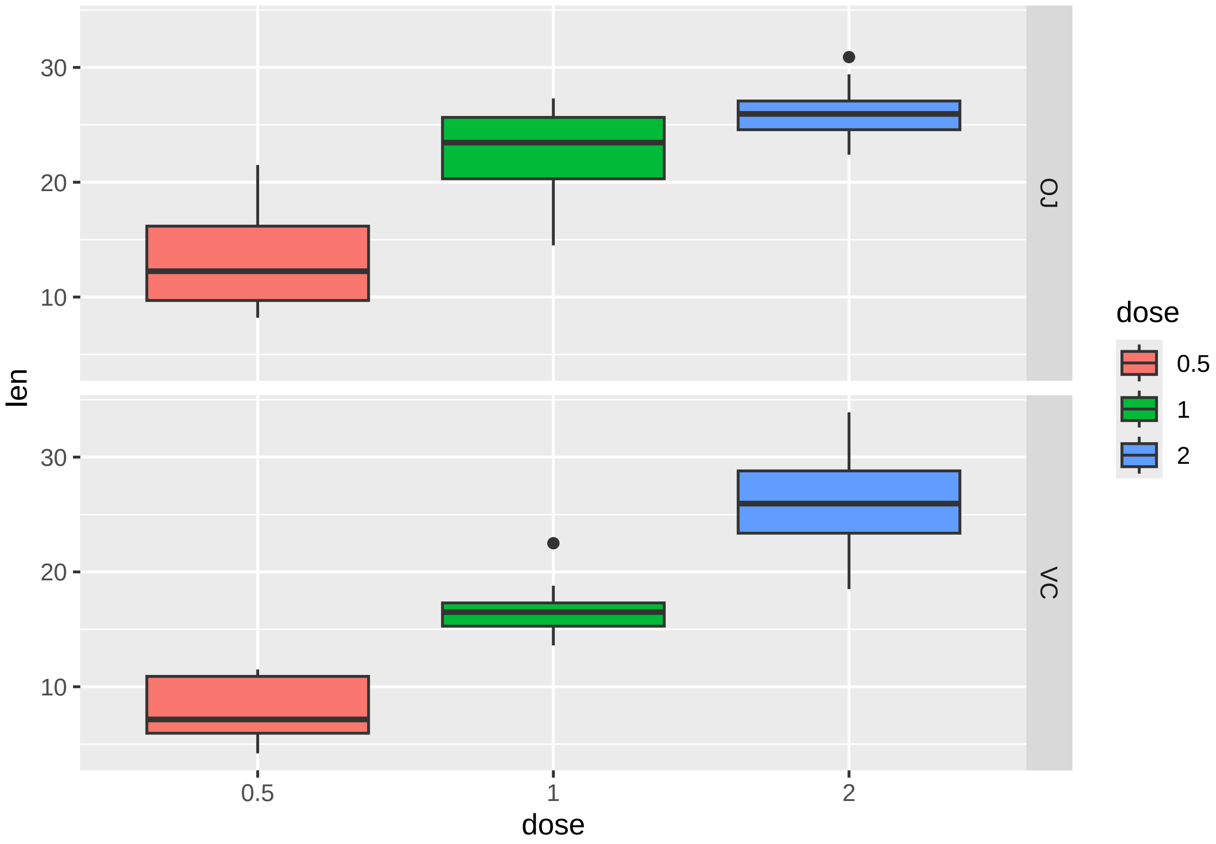
```
##    len supp dose
## 1  4.2   VC  0.5
## 2 11.5   VC  0.5
## 3  7.3   VC  0.5
## 4  5.8   VC  0.5
## 5  6.4   VC  0.5
## 6 10.0   VC  0.5
```

```
bp <- ggplot(df, aes(x=dose, y=len, group=dose)) + geom_boxplot(aes(fill=dose))
print(bp)
```



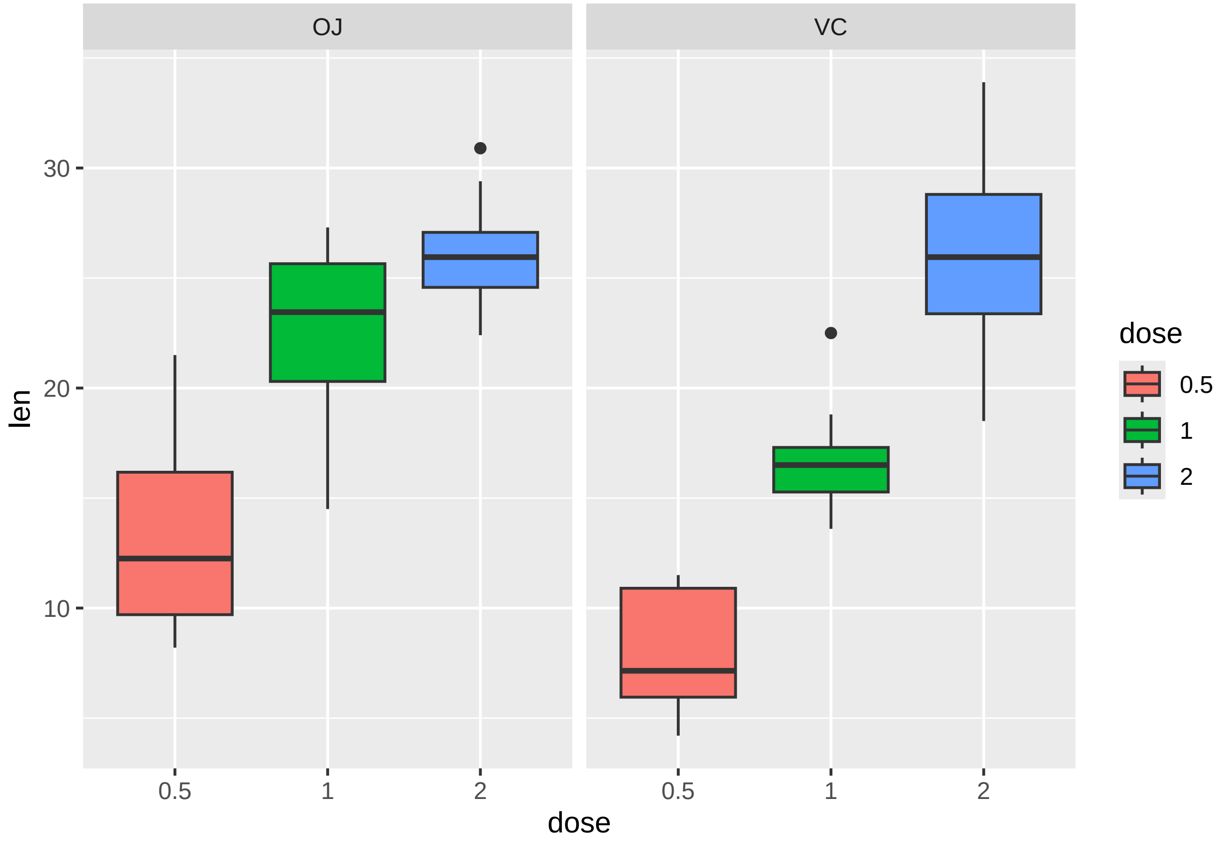
To split in vertical direction

```
bp + facet_grid(supp ~ .)
```

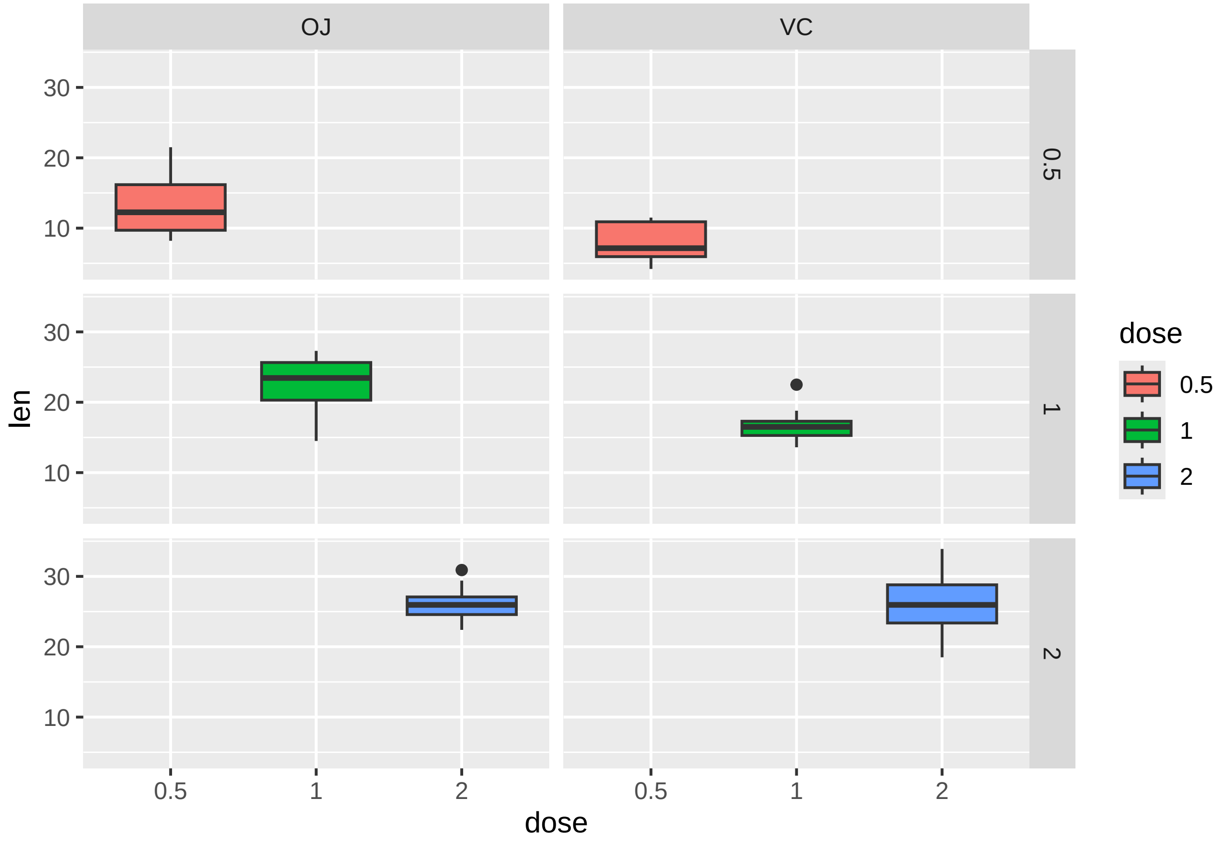


To split in horizontal direction

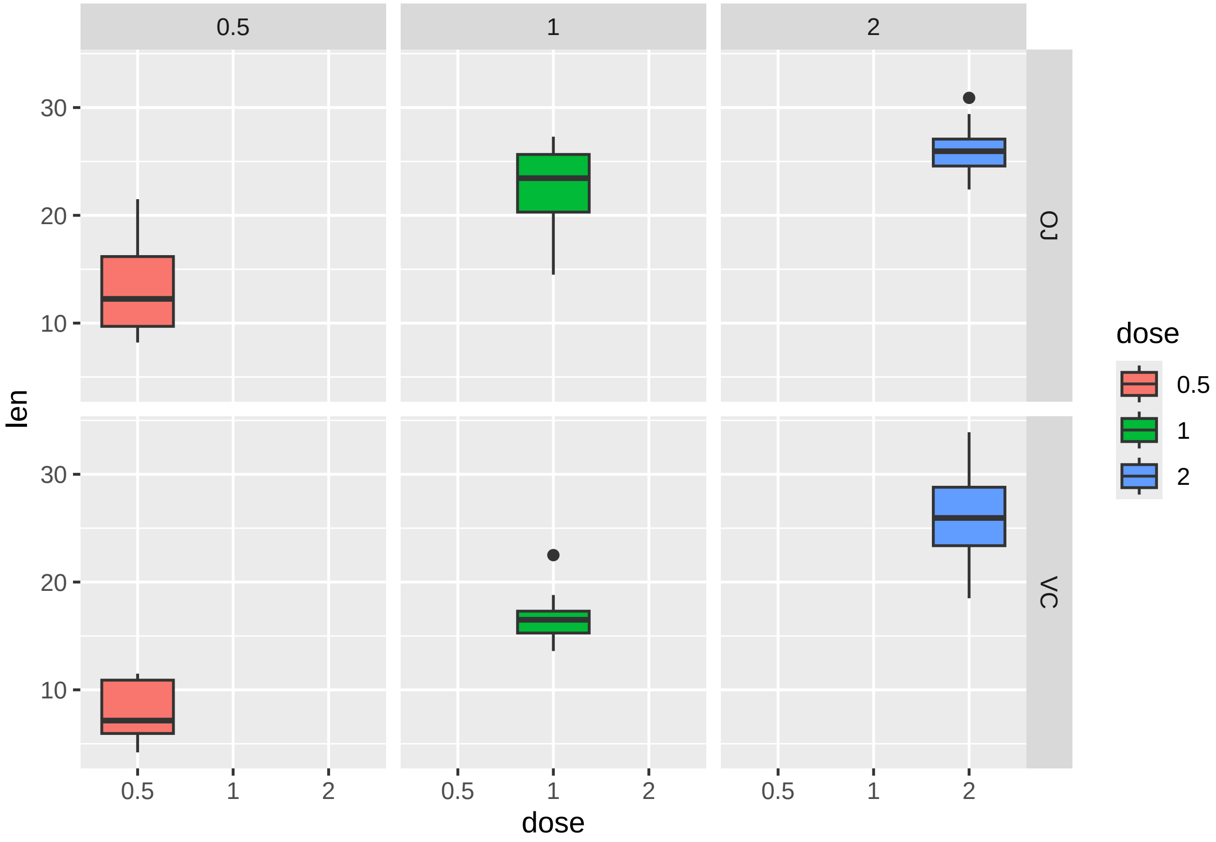
```
bp + facet_grid(. ~ supp)
```



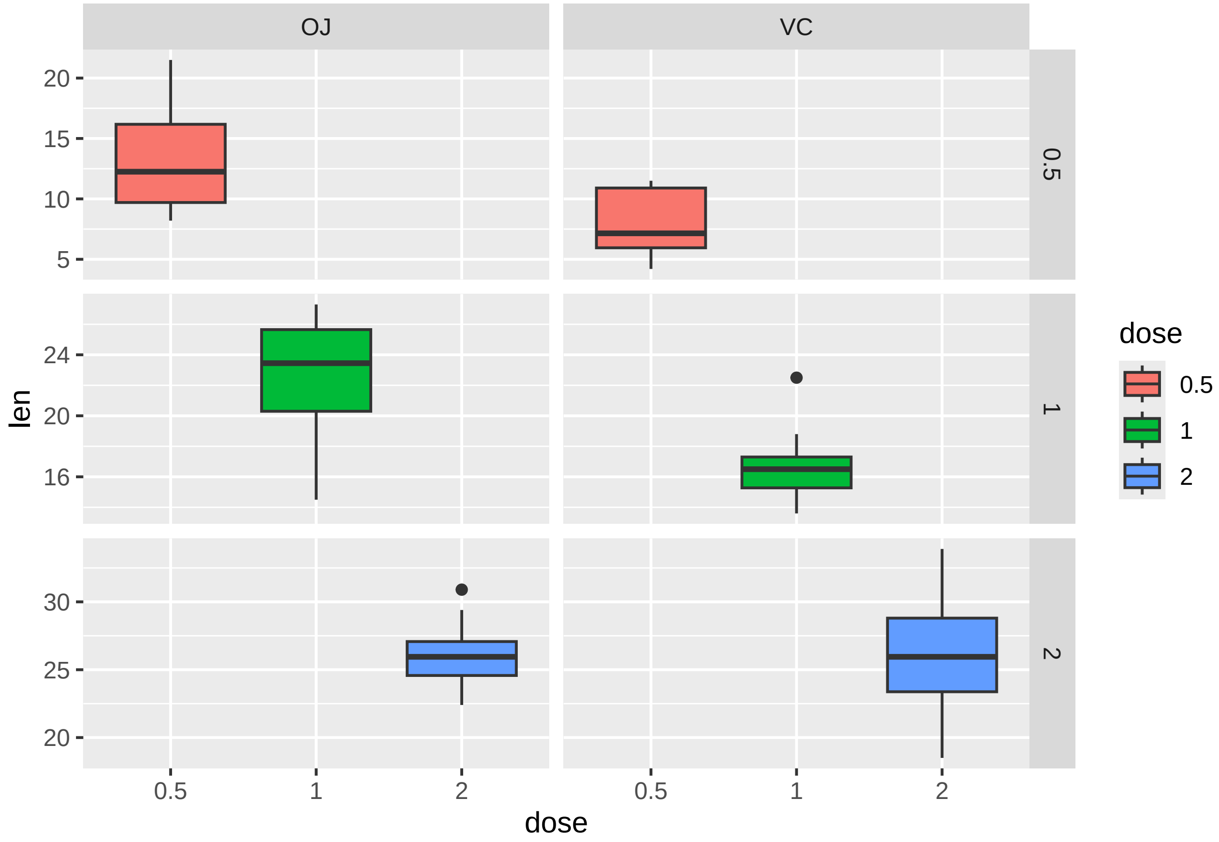
```
#To split the graph by the levels of the groups "dose" and "supp":
bp + facet_grid(dose ~ supp)
```



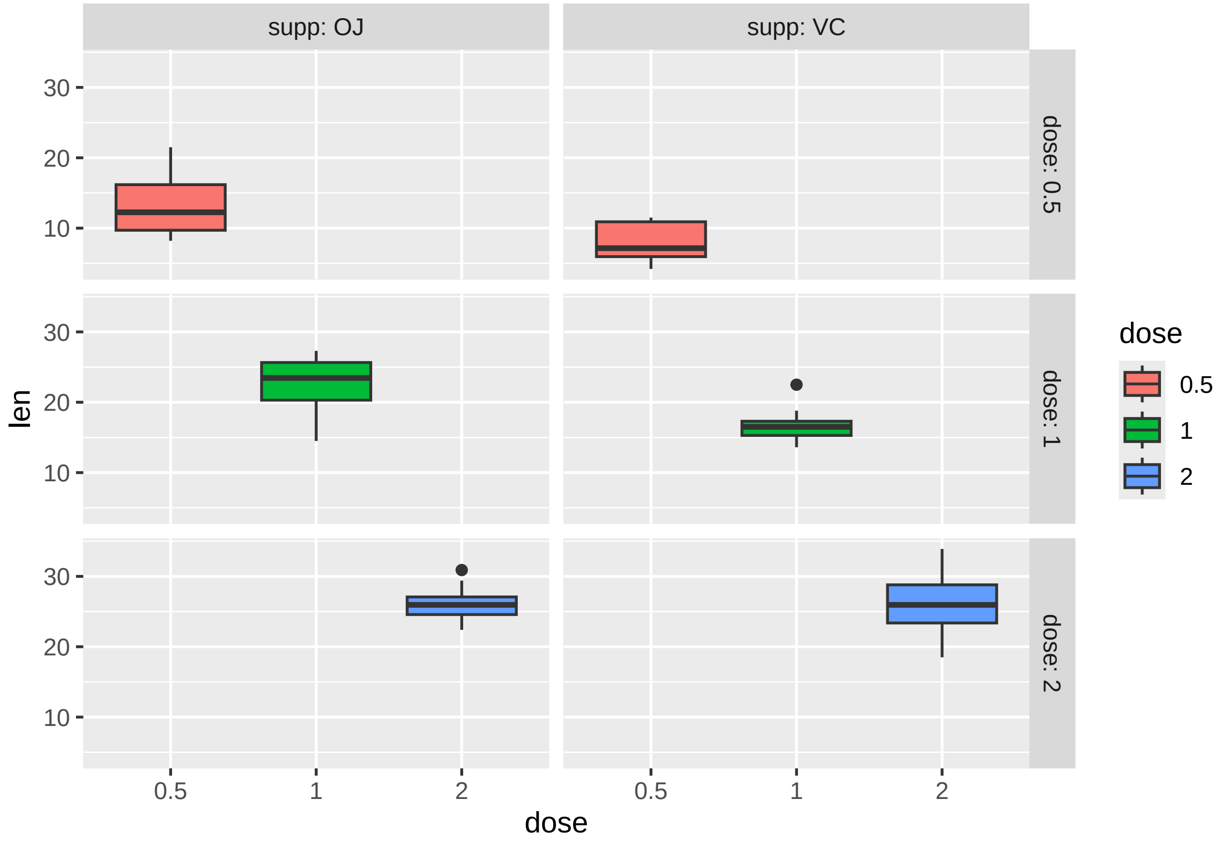
```r
bp + facet_grid(supp ~ dose)
```



```
bp + facet_grid(dose ~ supp, scales='free')
```

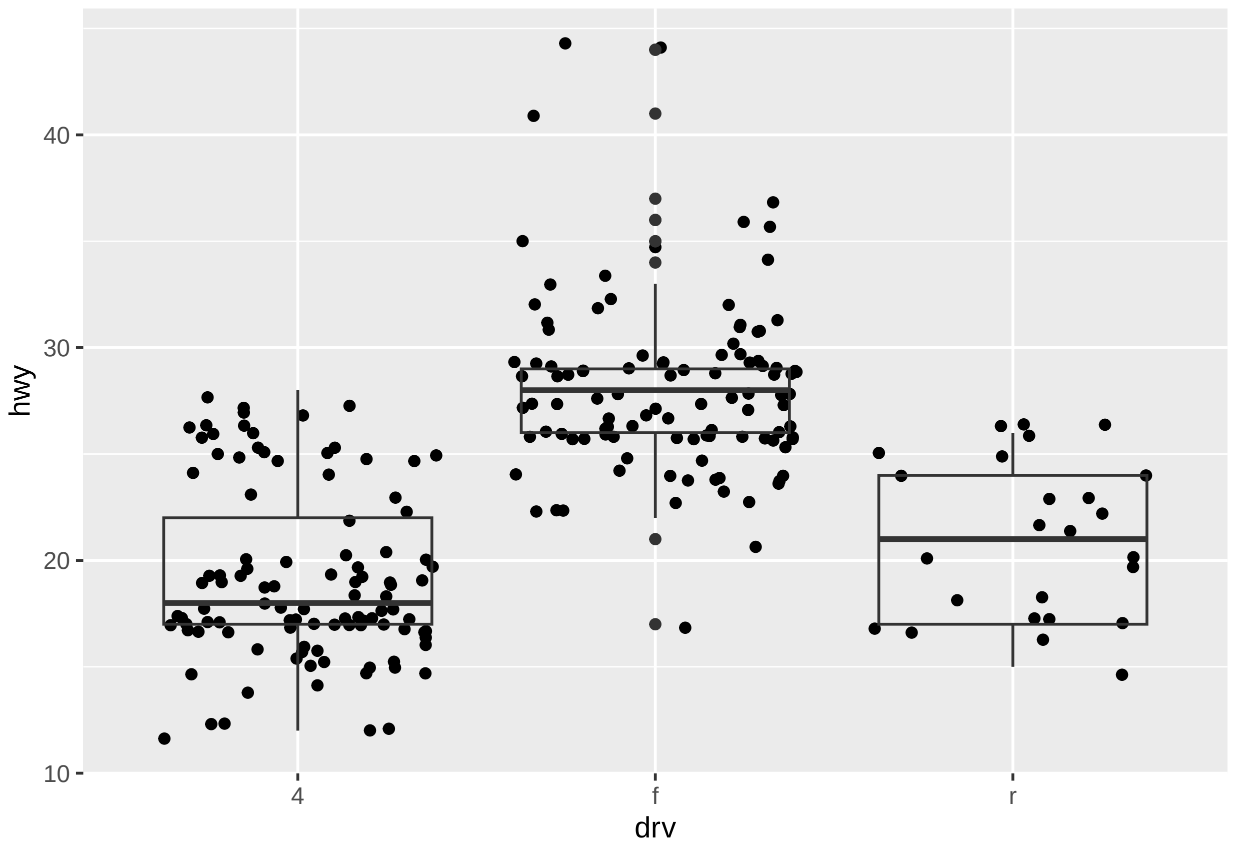


```r
bp + facet_grid(dose ~ supp, labeller=label_both)
```

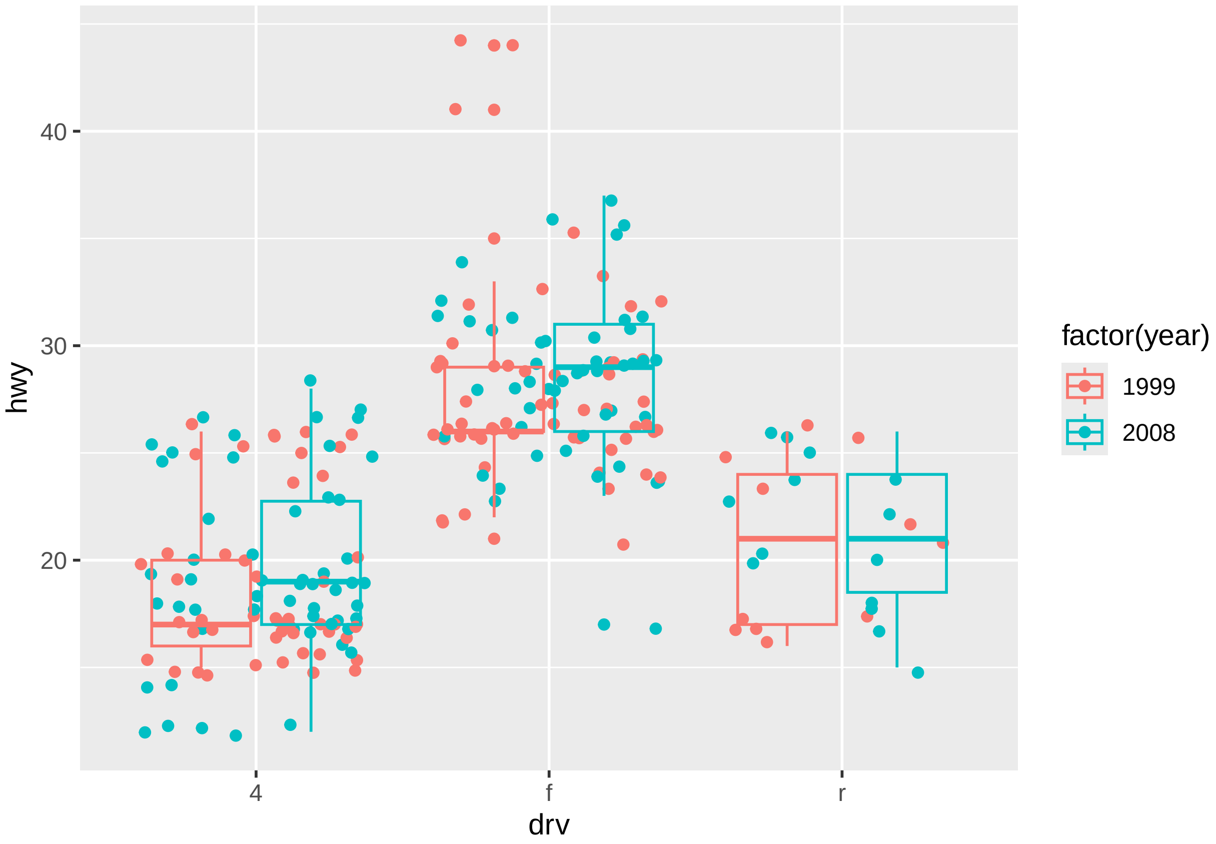


Grouping

```
p1 <- ggplot(mpg, aes(drv, hwy)) +
 geom_jitter() +
 stat_boxplot(fill = NA)
p1
```



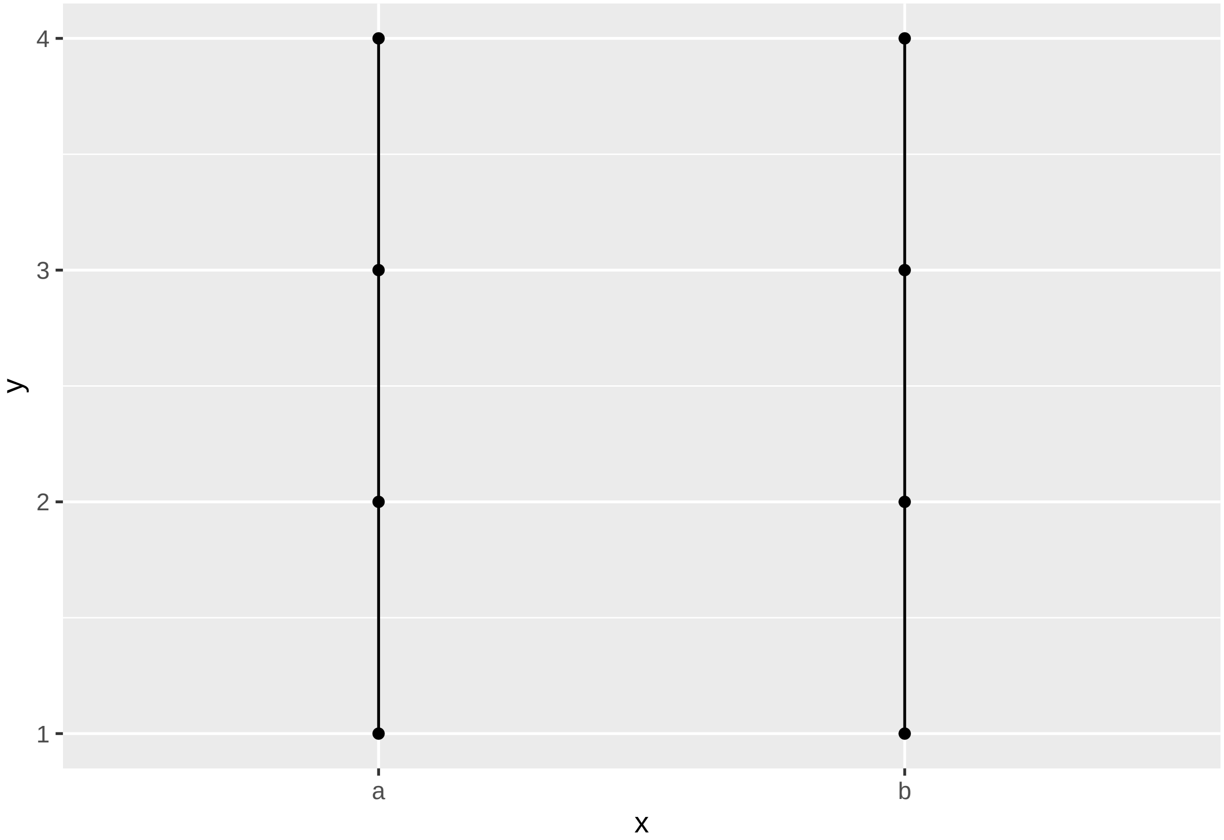
```
p2 <- ggplot(mpg, aes(drv, hwy, color = factor(year))) +
 geom_jitter() +
 stat_boxplot(fill = NA)
p2
```



```r
# make up a simple dummy data frame
df <- data.frame(x = c("a", "a", "a", "a", "b", "b", "b", "b"),
 y = c(1, 2, 3, 4, 4, 3, 2, 1),
 z = c("A", "A", "B", "B", "B", "B", "A", "A"))
```

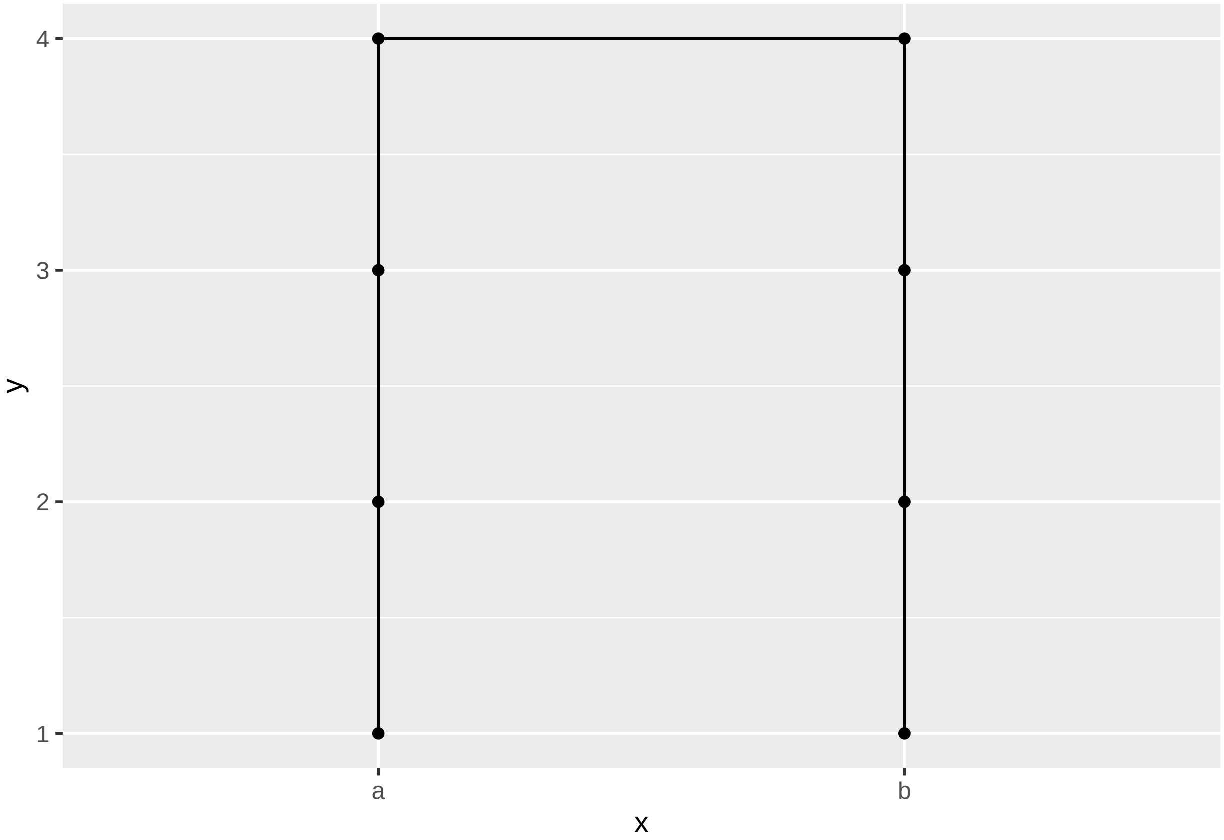
Interpretation: creating a data frame with specified values for three columns, where x and z are categorical variables and y is a numerical variable

```r
ggplot(df, aes(x, y)) +
 geom_point() +
 geom_path()
```



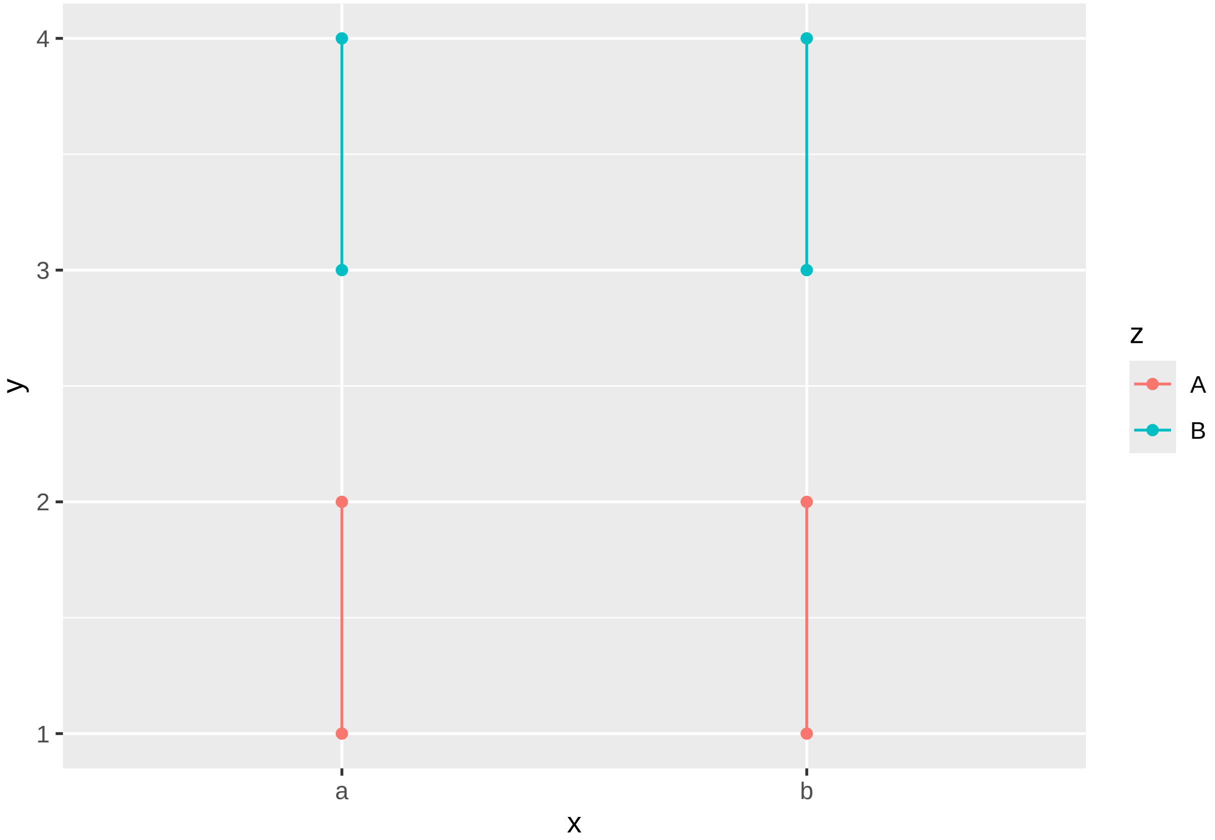
Interpretation:This code plot a scatter plot with points connected by lines in the order they are listed in the data frame, effectively showing the progression or sequence of the data.

```
ggplot(df, aes(x, y)) +
 geom_point() +
 geom_path(aes(group = 10))
```



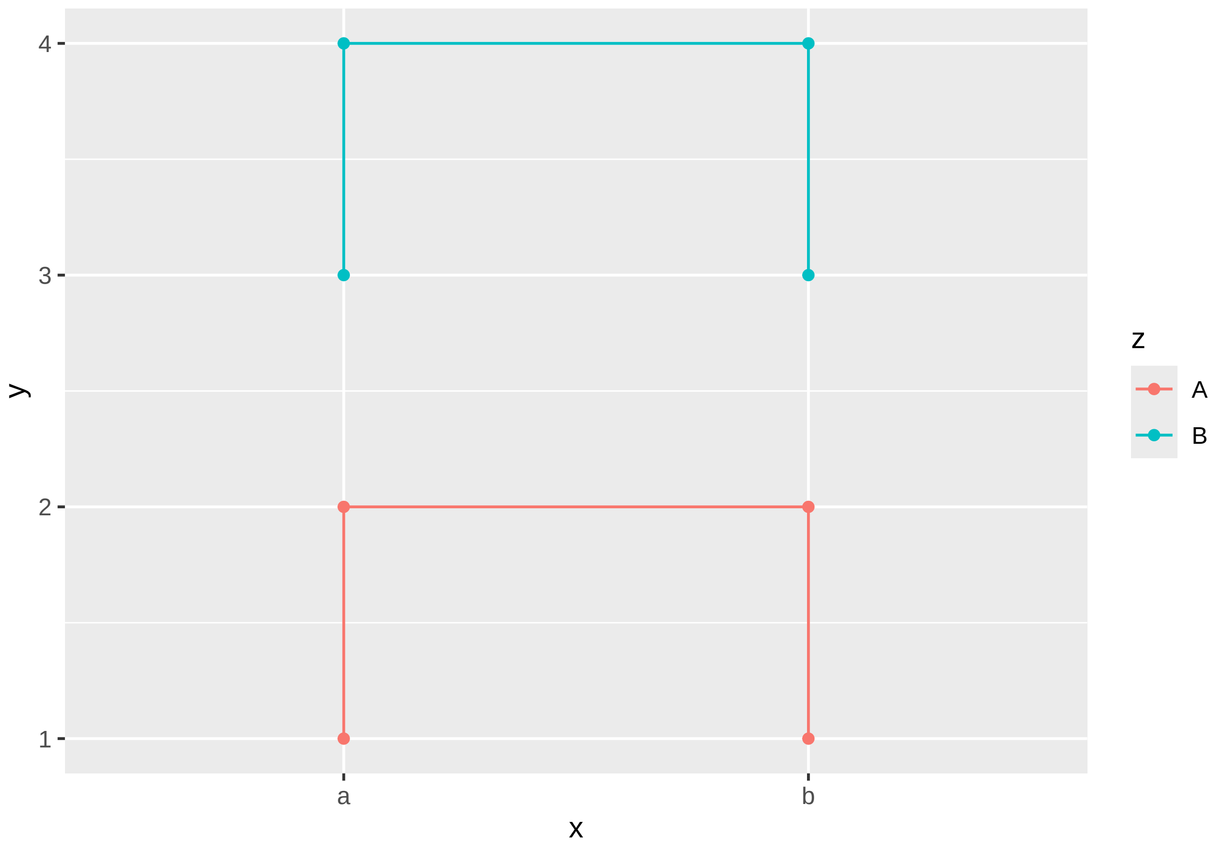
Interpretation:This code plots scatter plot with a single line connecting all the points in the order they are listed in the data frame, indicating the sequence of the data as a continuous path

```
ggplot(df, aes(x, y, color = z)) +
 geom_point() +
 geom_path()
```



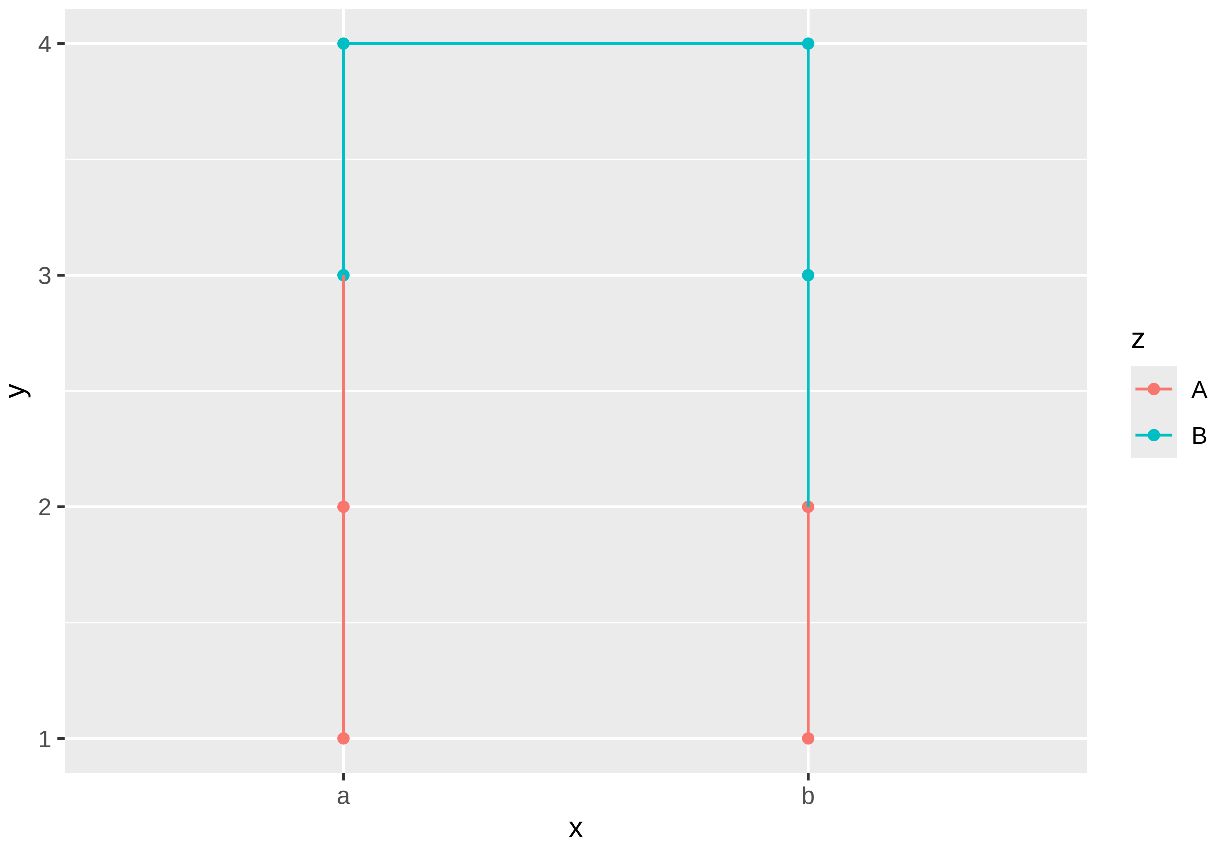
Interpretation:This code generates a scatter plot with points colored according to the variable z and connects these points with lines to show the sequence or trend based on their order in the data.

```
ggplot(df, aes(x, y, color = z)) +
 geom_point() +
 geom_path(aes(group = z))
```



Interpretation:This code creates a scatter plot where points are colored by z, and lines connect the points within each group defined by z. Each group will have a distinct line connecting its points, which helps visualize trends or paths for each category.

```
ggplot(df, aes(x, y, color = z)) +
 geom_point() +
 geom_path(aes(group = 10))
```



Interpretation: This code creates a scatter plot where points are colored based on the variable z, and a single line is drawn connecting all points in the dataset, since all points are grouped together under group = 10