

# ASSIGNMENT 4

Salma Elhassa

03/03/2022

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed R codes and figures.

## plotting basics with ggplot

## my tutorial script

## lots and lots of annotation

#libraries I need (no need to install)

```
setwd("~/Desktop/BIO 47120") #set working directory anything in () are  
arguments:inputs  
library(dplyr) # Load a package into memory
```

```
##  
## Attaching package: 'dplyr'  
  
## The following objects are masked from 'package:stats':  
##  
##   filter, lag  
  
## The following objects are masked from 'package:base':  
##  
##   intersect, setdiff, setequal, union
```

```
library(ggplot2) # Load another package  
getwd() # show working directory to confirm
```

```
## [1] "/Users/salmaelhassa/Desktop/BIO 47120"
```

##There are 3 variables and 40 observations. Root and Fruit variables are numeric type (floats) and the Grazing variable is a categorical type (string).

```
rm(list = 1)
```

```
## Warning in rm(list = 1): object '1' not found
```

#get the data

```
library(readr)
setwd("~/Desktop/BIO 47120/datasets-master")
compensation <- read_csv("compensation.csv")
```

```
## Rows: 40 Columns: 3
```

```
## — Column specification
```

---

```
## Delimiter: ","
```

```
## chr (1): Grazing
```

```
## dbl (2): Root, Fruit
```

```
##
```

```
## ⓘ Use `spec()` to retrieve the full column specification for this data.
```

```
## ⓘ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

#check out the data

```
glimpse(compensation)
```

```
## Rows: 40
```

```
## Columns: 3
```

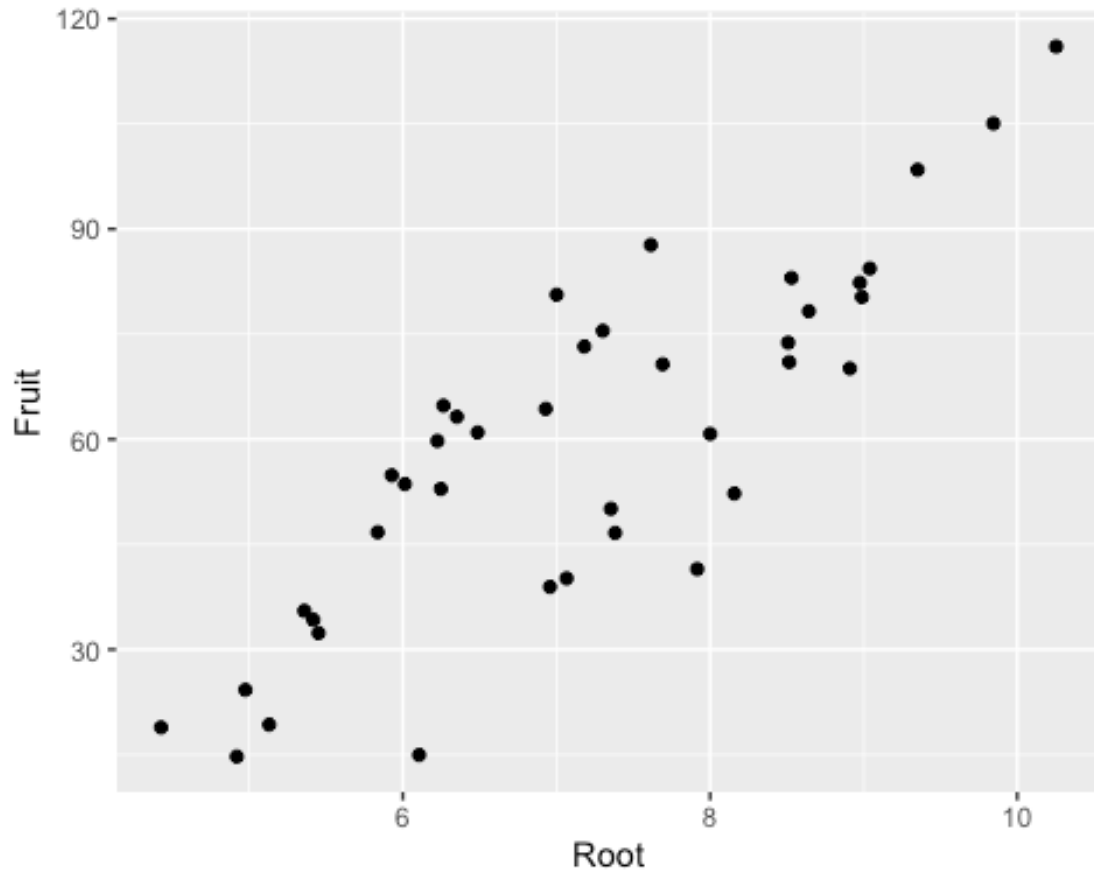
```
## $ Root    <dbl> 6.225, 6.487, 4.919, 5.130, 5.417, 5.359, 7.614, 6.352, 4.975,...
```

```
## $ Fruit   <dbl> 59.77, 60.98, 14.73, 19.28, 34.25, 35.53, 87.73, 63.21, 24.25,...
```

```
## $ Grazing <chr> "Ungrazed", "Ungrazed", "Ungrazed", "Ungrazed", "Ungrazed", "U...
```

#make my first ggplot

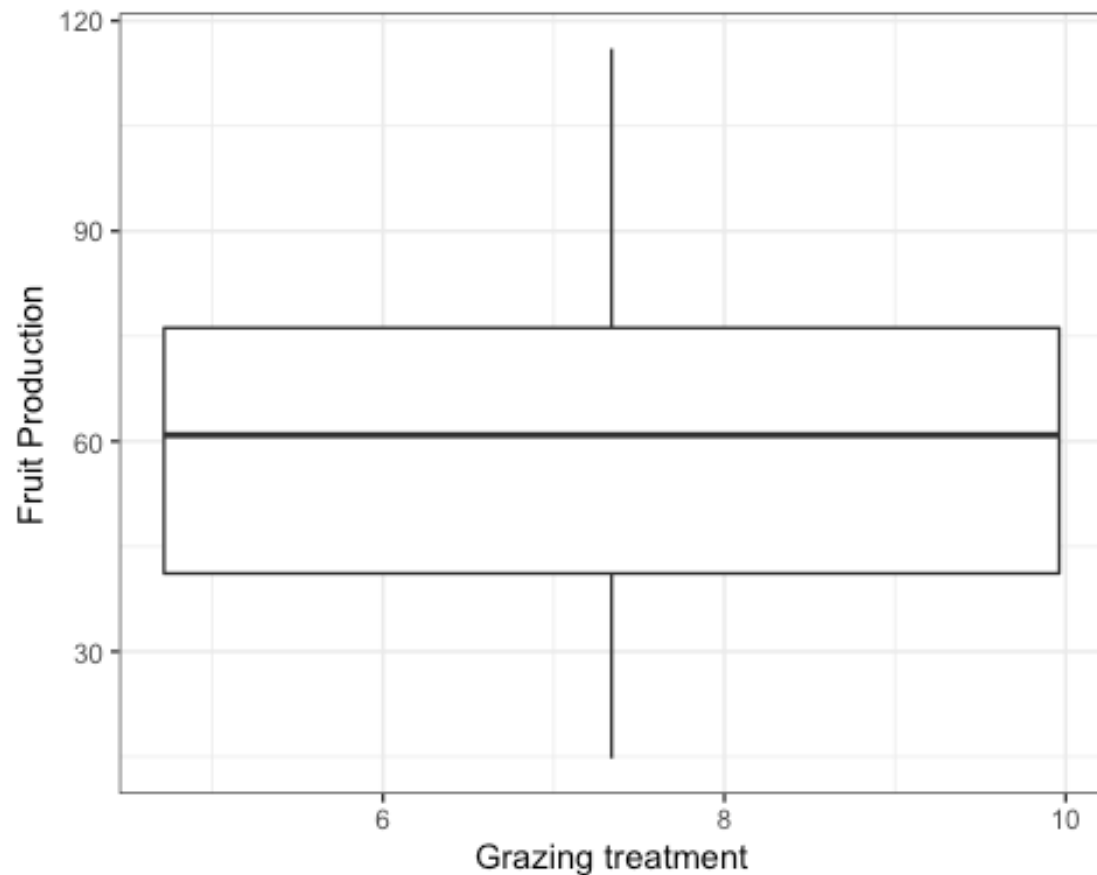
```
ggplot(compensation, aes(x = Root, y = Fruit)) + geom_point()
```



#box plot categorical variable on x-axis; numerical variable on y-axis #theme is black and white

```
ggplot(compensation, aes(x = Root, y = Fruit)) +  
  geom_boxplot() +  
  xlab("Grazing treatment") +  
  ylab("Fruit Production") +  
  theme_bw()
```

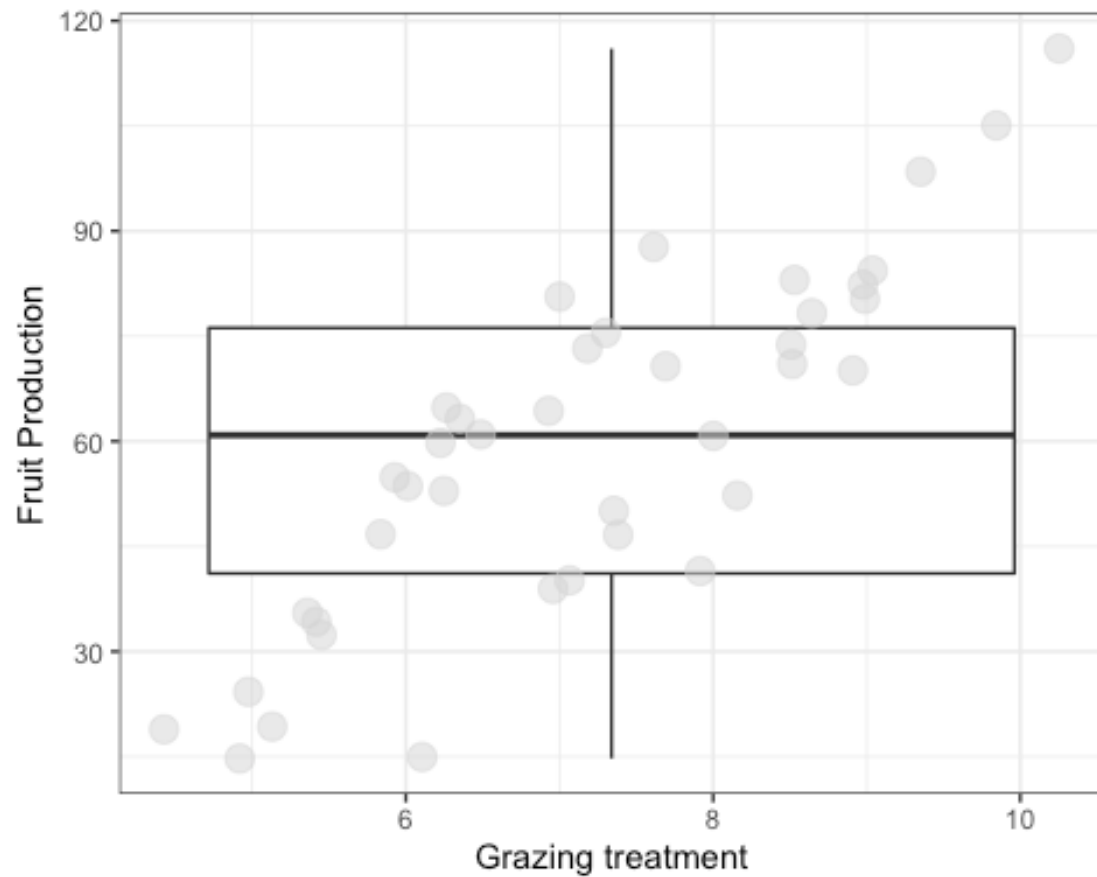
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?



#box plot categorical variable on x-axis; numerical variable on y-axis #theme is black and white #adjusting size and color

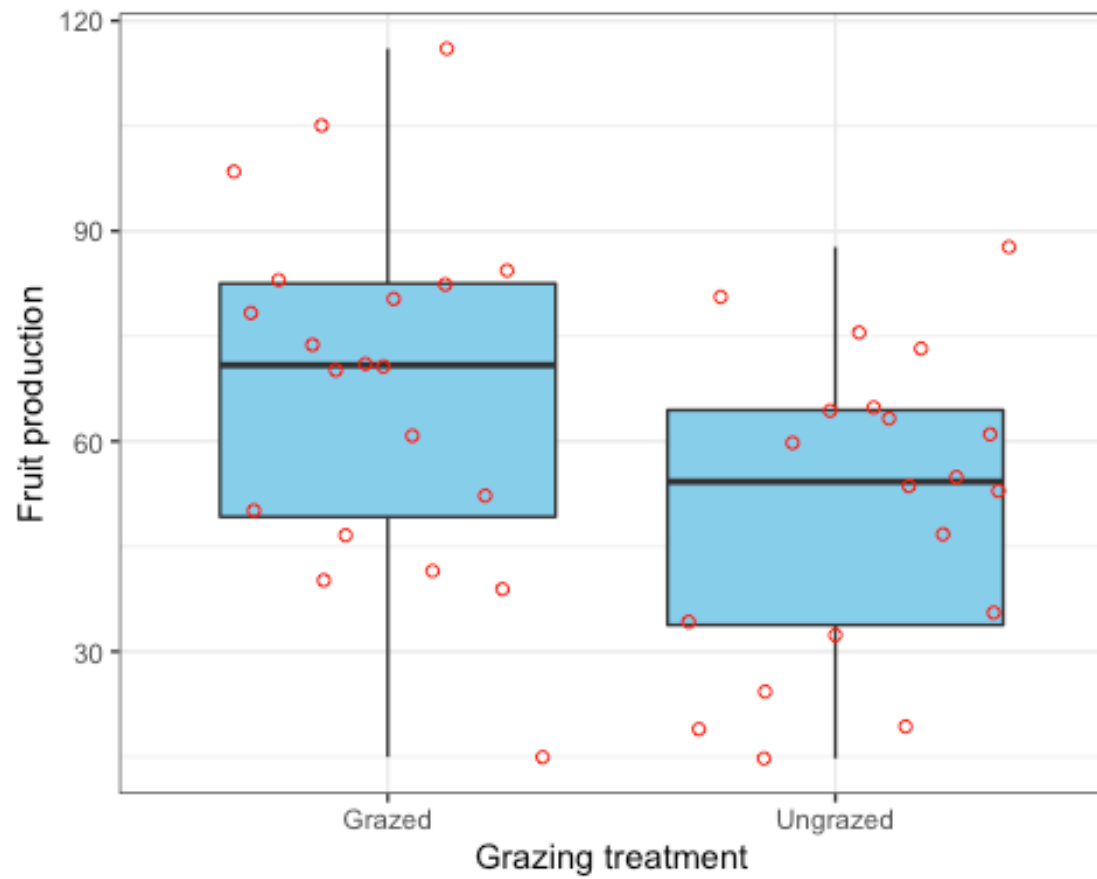
```
ggplot(compensation, aes(x = Root, y = Fruit)) +  
  geom_boxplot() +  
  geom_point(size = 4, color = 'lightgrey', alpha = 0.5) +  
  xlab("Grazing treatment") +  
  ylab("Fruit Production") +  
  theme_bw()
```

## Warning: Continuous x aesthetic -- did you forget aes(group=...)?



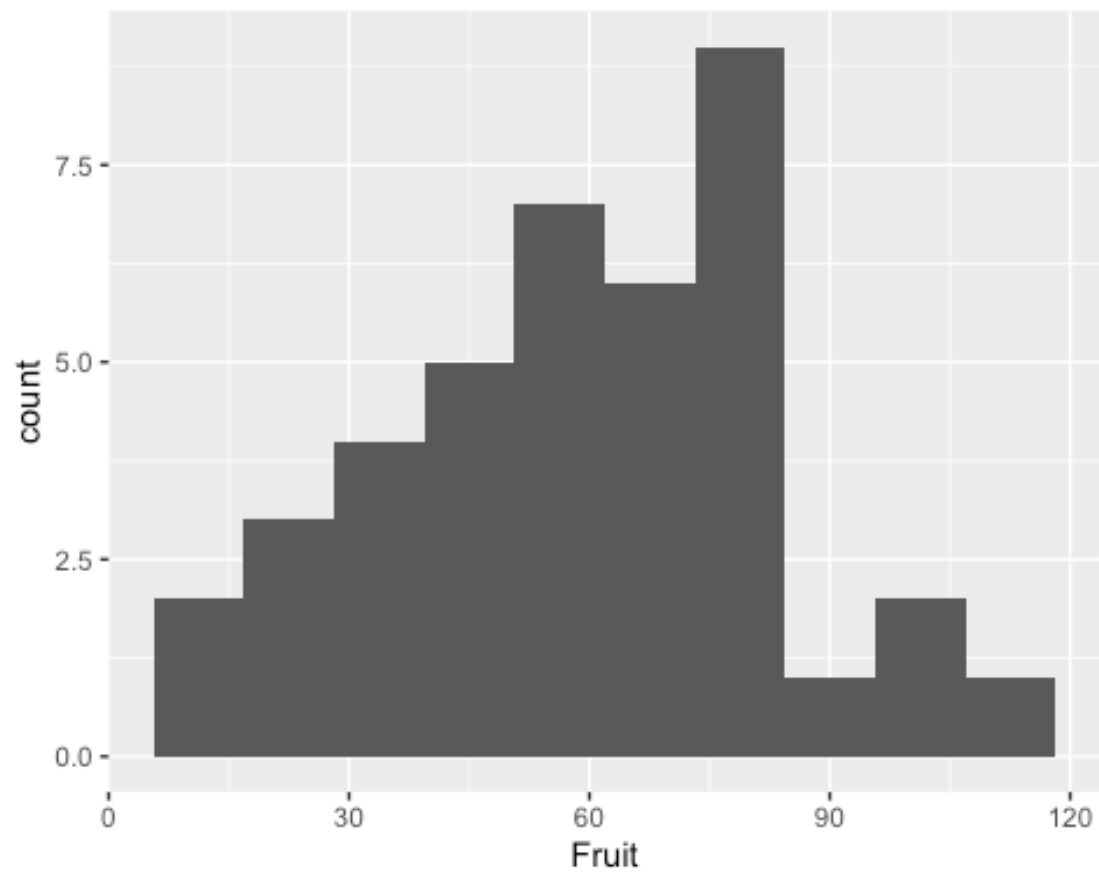
#box plot use 'jitter' to spread points and show sample sizes #show production for each category of Grazing

```
ggplot(data = compensation, aes(x=Grazing, y=Fruit)) +
  geom_boxplot(fill="skyblue") + geom_jitter (shape = 1,
  color = "red") + theme_bw() + xlab("Grazing treatment")+ ylab("Fruit
  production")
```

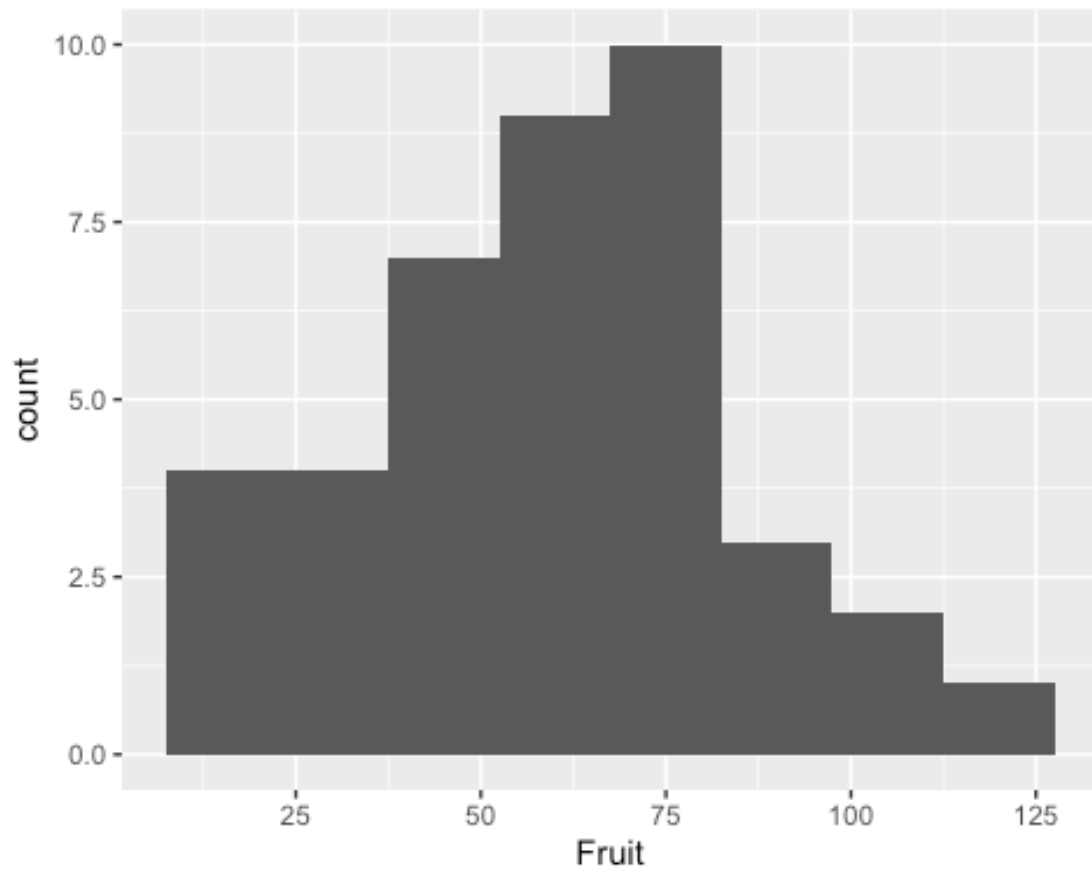


#histogram to show distribution of a numerical variable

```
ggplot(data = compensation, aes(x= Fruit)) +  
  geom_histogram(bins = 10)
```



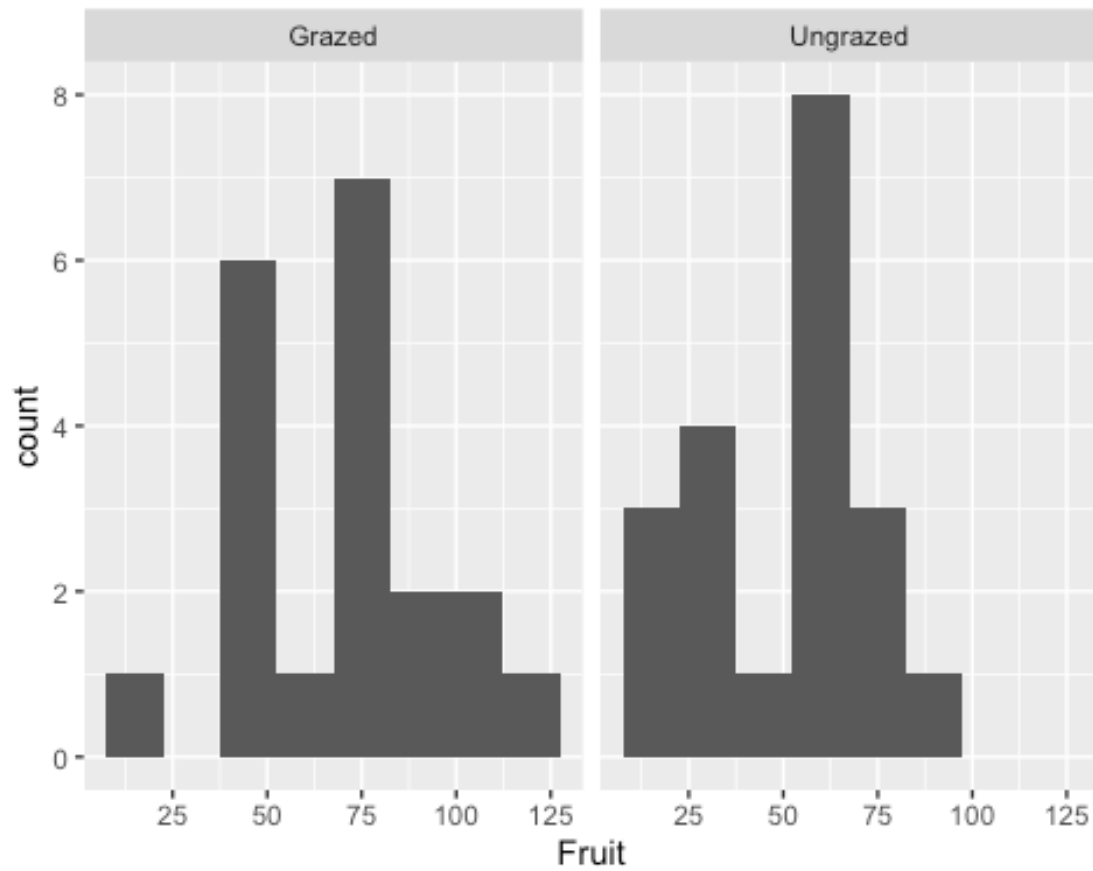
```
ggplot(data = compensation, aes(x= Fruit)) +  
  geom_histogram(binwidth = 15)
```



##histogram to show distribution of a numerical variable - seperating grazed and ungrazed  
- two variables sharing the same y - axis #panel

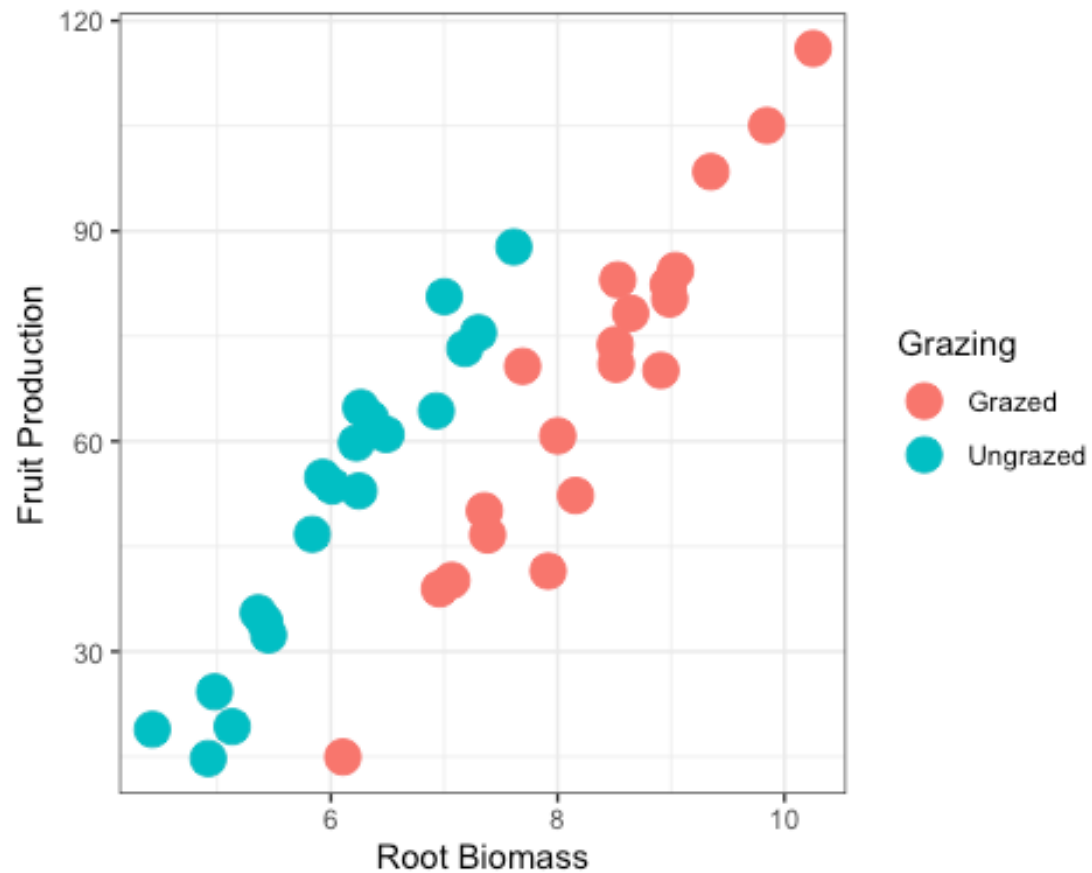
```
ggplot(data = compensation, aes(x= Fruit)) +  
  geom_histogram(binwidth = 15) +  
  facet_grid(~Grazing)
```





#using color as another dimension for scatterplot

```
ggplot(compensation, aes(x = Root, y = Fruit, colour = Grazing)) +  
  geom_point(size = 5) +  
  xlab("Root Biomass") +  
  ylab("Fruit Production") +  
  theme_bw()
```



#assign colors to a categorical variable ("Grazing") & show regression line #with confidence band

```
ggplot(compensation, aes(x = Root, y = Fruit, colour = Grazing)) +
  geom_point() + geom_smooth(method = "lm") + theme_bw()
## `geom_smooth()` using formula 'y ~ x'
```

