An overview of Statistical Learning

by

AINDRILA GARAI

MSC STATISTICS, IIT KANPUR

aindrilag22@iitk.ac.in

What is Statistical Learning?

A large set of tools to understand data. This is of two types-

1. Supervised*: It contains building statistical models for predicting and estimating based on an output depending one or more inputs. Here our data contains the output values besides inputs to supervise our model.*

Some methods of Supervised Learning:

- **Regression Problem** - When outcome measurement is quantitative.

eg. Identify the risk factors for prostate cancer, based on clinical and demographic variables.

- **Classification Problem** - When our predicted output is not a numerical value, only provide some qualitative(ordinal/ nomial) output.

eg. An automatic spam detector that can separate between email and spam, Handwritten Digit Recognition

2. Unsupervised*: Only input values are available without any output values that can supervise any model. So, here we focus to observe we observe only the features and have no measurements of the outcome.*

A method of Unsupervised Learning:

- **Clustering Problem**: When there is no output value, we are interested in determining whether there are clusters among the observations by grouping individuals according to their observed characteristics.(We can plot $\frac{p(p-1)}{2}$ scatter plots using p variables)

eg. Association between advertising(TV, radio, newspaper) and sales.

NOTES:

- The **inputs** go by different names such as predictors, independent variables or sometimes just variables features & the **output** variables are often called the response or dependent variable.

- We use least square regression for Regression Problem & logistic regression for Classification Problem.

- **Semi-supervised:** In this setting, we wish to use a statistical learning method that can incorporate the m observations for which response measurements are available as well as the n − m observations for which they are not. (In the real life when responses are expensive and predictors are cheap to collect)

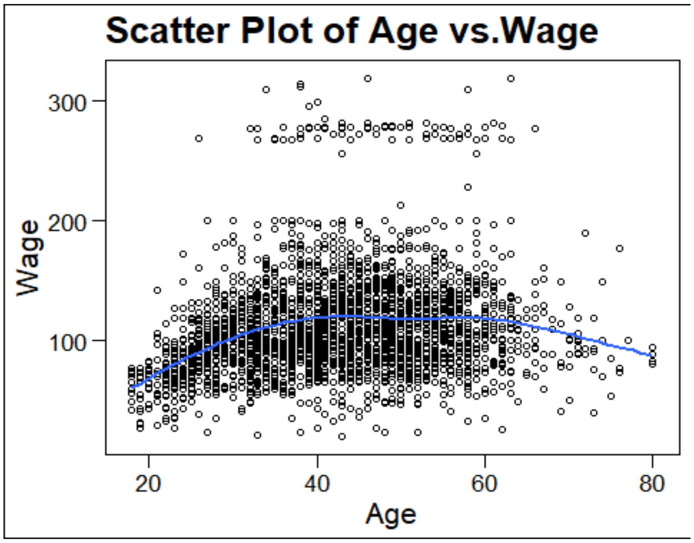
Some example of data visualizations:

1. Wage data:

Wage data which contains income survey information for men from the central Atlantic region of the United States.

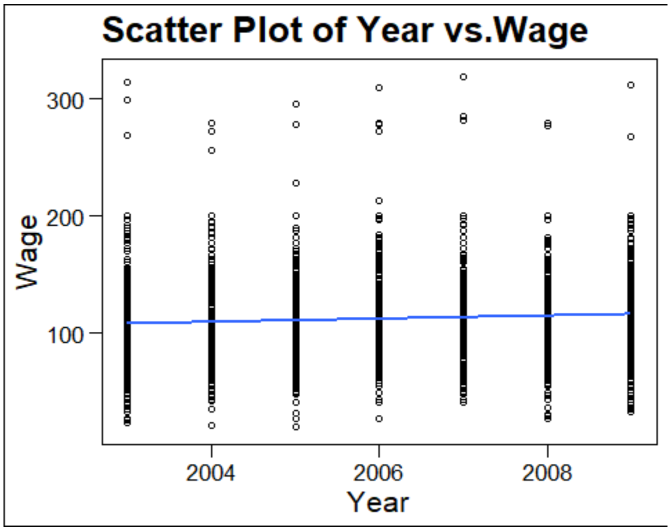
Remark: *X axis should be independent w.r.t Y axis.*

```
## calling the libraries
library(ggplot2)
library(ggthemes)
library(ISLR2)
data("Wage")

ggplot(Wage,aes(age,wage))+
  geom_point(shape=1)+
  geom_smooth(se=FALSE)+
  labs(x="Age", y="Wage")+
  theme_base()+
  labs(title="Scatter Plot of Age vs.Wage")

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```



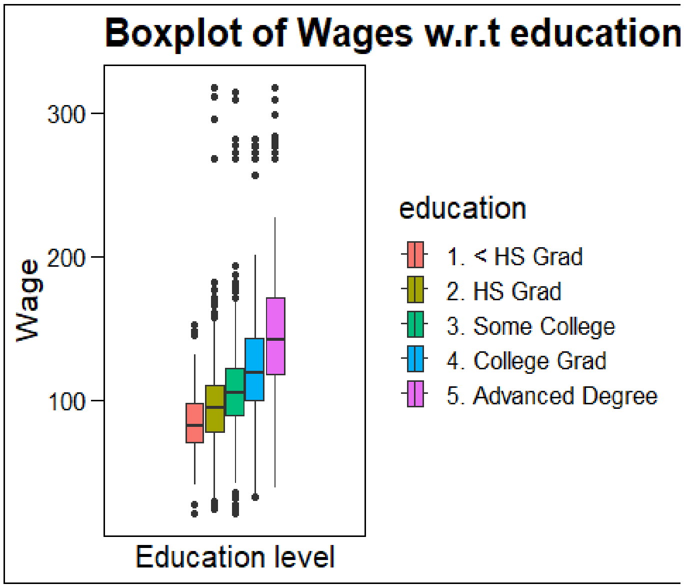
Wage increases with age until about 60 years of age, at which point it begins to decline.

```
ggplot(Wage)+
  geom_point(aes(year,wage),shape=1)+
  geom_smooth(aes(year,wage),se=FALSE,method=lm)+
  labs(x="Year", y="Wage")+
  theme_base()+
  labs(title="Scatter Plot of Year vs.Wage")

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



There is a slow but steady increase of approximately $10,000 in the average wage between 2003 and 2009.

```
ggplot(Wage)+
  geom_boxplot(aes(wage,fill=education))+
  theme_base()+
  labs(y="Education level",x="Wage")+
  scale_y_discrete(breaks=c("-0.4","-0.2","0.0","0.2","0.4"))+
  coord_flip()+
  labs(title="Boxplot of Wages w.r.t education level")
```



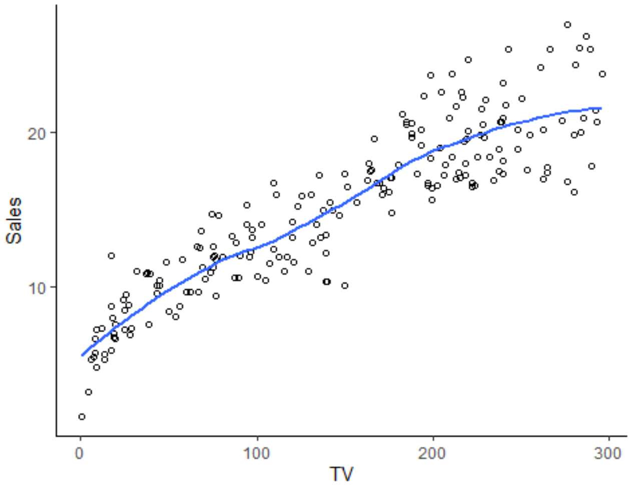
On average, wage increases with the level of education.

- Clearly, the most accurate prediction of a given man's wage will be obtained by fitting an appropriate model combining his age, his education, and the year which is known a regression problem.

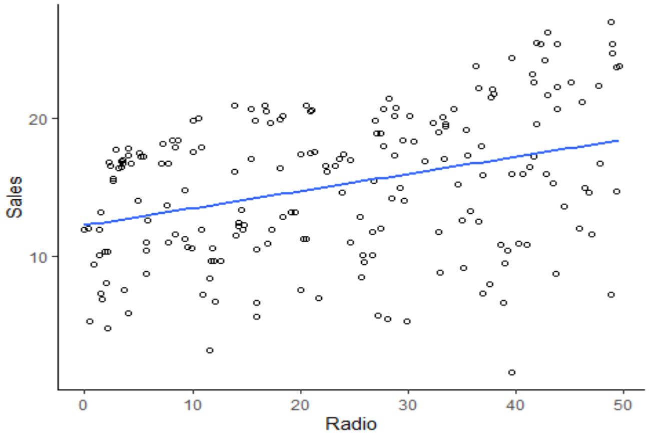
- To under stand these relationships is a clustering problem

2. Advertising data:

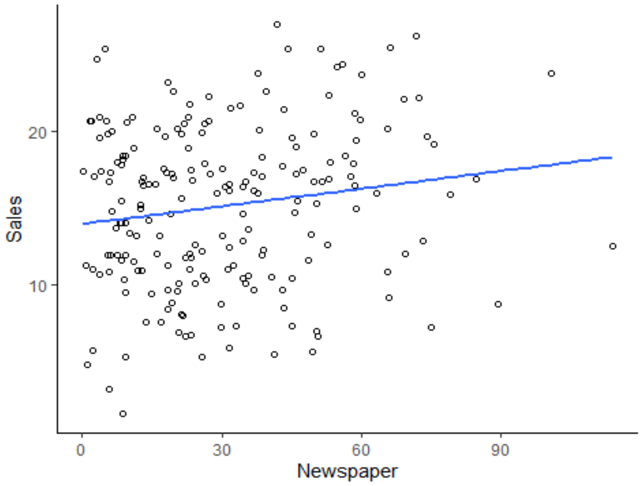
```
dat <- read.csv("C:\\Users\\AINDRILA\\OneDrive\\Documents\\advertising.csv")
ggplot(dat,aes(TV,Sales))+
  geom_point(shape=1)+
  geom_smooth(se=FALSE)+
  theme_classic()

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



```
ggplot(dat,aes(Radio,Sales))+
  geom_point(shape=1)+
  geom_smooth(se=FALSE,method=lm)+
  theme_classic()

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



```
ggplot(dat,aes(Newspaper,Sales))+
  geom_point(shape=1)+
  geom_smooth(se=FALSE,method=lm)+
  theme_classic()

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



The plot displays sales, in thousands of units, as a function of TV, radio, and newspaper budgets, in thousands of dollars, for 200 different markets. Each blue line represents a simple model that can be used to predict sales using TV, radio, and newspaper, respectively(Regression problem).But how much Sales are dependent on each advertising medium, our answer would be sales are positively correlated with TV and radio(clustering problem).

prediction and inference:

Suppose that we observe a quantitative response Y and p different predictors $X_1, X_2, \ldots, X_p$. We assume that there is some relationship between Y and X= $(X_1, X_2, \ldots, X_p)$, which can be written in the very general form $Y = f(X) + \epsilon$. Here f is some fixed but unknown function of $X_1, \ldots, X_p$ and ϵ is a random error term, which is independent of X and has mean zero.

we may wish to estimate f for two reasons: prediction and inference.

- **Prediction**:

$\hat{Y} = \widehat{f(X)}$,where $\hat{f}$ represents our estimate for f, and $\hat{Y}$ represents the resulting prediction for Y which depends on reducible error and irreducible error.

- **Reducible error:** We can potentially improve the accuracy of $\hat{f}$ by using the most appropriate statistical learning technique to estimate f.

eg. Patient's risk for a severe adverse reaction to a particular drug.

- **Irreducible error:** Error introduced by ϵ(unmeasured variable) that affects the accuracy of our predictions.

eg. The risk of an adverse reaction might vary for a given patient on a given day, depending on manufacturing variation in the drug itself or the patient's general feeling of well-being on that day.

$$\mathrm{E}\left(Y-\hat{Y}\right)^2 = \underbrace{\left[f(X)-\hat{f}(X)\right]^2}_{\text{Reducible}} + \underbrace{\mathrm{Var}(\epsilon)}_{\text{Irreducible}} \ ,$$

where $\mathrm{E}\left(Y-\hat{Y}\right)^2$ represents the average, or expected value, of the squared expected value difference between the predicted and actual value of Y , and $\mathrm{Var}(\epsilon)$ represents the variance associated with the error term ϵ.

- **Inference**:

To understand the association between Y and $X_1, \ldots, X_p$. Our inquiries are:

- Which predictors are associated with the response? Identify the few important predictors.
- What is the relationship between the response and each predictor? Positive or negative relationships.
- Can the relationship between Y and each predictor be adequately summarized using a linear equation or is the relationship more complicated? Yes, no or maybe.

NOTES:

- For accurate prediction we use non-linear approaches that is less interpretable and requires a large number of observations.
- For inference we use linear model that is interpretable but not provide accurate predictions and doesn't require a small no of observations.
- In many scenario we conduct both prediction and inference problem.
- **Training data:** These observations(inputs and also outputs) are used to train or teach our model or to find $\hat{f}$(for regression)
- **Test data:** These observations(inputs and also outputs) are known to us but we don't use this to fit our model rather than we use these data to examine accuracy of the fitted model.

Parametric Method:

1. First, let us assume a linear model of (p+1) unknown coefficients($\beta_i, i = 0(1)p$)

$$f(X) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p$$

2. Now we use mostly least square method to obtain the unknown coefficients.

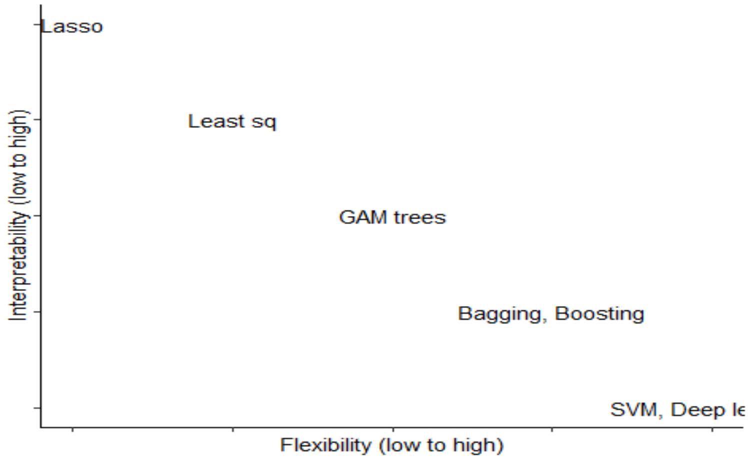
eg- linear regression, lasso, . Generalized additive models (non-linear) etc.

Remark:

- Good for inference but not flexible, estimate is poor(because it can only generate linear functions).
- To fit a flexible model, requires a large no of parameters.
- These more complex models can lead to a phenomenon known as overfitting the data, which essentially means they follow the errors or noise too closely.

Non-Parametric Method:

This approach does not impose any pre-specified model on f. eg- splines, boosting methods(used for both quantitative and qualitative) etc.



Remark:

- Flexible but inference can't be done(because they can generate a much wider range of possible shapes to estimate f).
- Properly select level of smoothness otherwise overfitting will occur.
- The degrees of freedom is a quantity that summarizes the flexibility of a curve.

Measuring the Quality of Fit:

1. In the regression the most commonly-used measure is the mean squared error (MSE) given by

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}(x_i)\right)^2$$

where $\hat{f}(x_i)$ is the prediction that $\hat{f}$ gives for the ith observation. The MSE will be small if the predicted responses are very close to the true responses, and will be large if for some of the observations, the predicted and true responses differ substantially.

2. In the classification setting, we use

$$\frac{1}{n}\sum_{i=1}^{n} I\,(y_i \neq \hat{y}_i)$$

Here $\hat{y}_i$ is the predicted class label for the ith observation using $\hat{f}$. And $I(y_i \neq \hat{y}_i)$ is an indicator variable that equals 1 if $(y_i \neq \hat{y}_i)$ and zero if $(y_i = \hat{y}_i)$.It computes the fraction of incorrect classifications.

Now, we are interested in the accuracy of the predictions that we obtain when we apply our method to previously unseen test data. Lowest train MSE doesn't imply lowest test MSE.

NOTE:

- As the flexibility of a model increases, we observe a monotone decrease in the training MSE and a U-shape in the test MSE.
- When a given method yields a small training MSE but a large test MSE, we are said to be overfitting the data because the supposed patterns that the method found in the training data simply don't exist in the test data.
- A good classifier is one for which the test error is smallest.
- **Cross-validation:** A crossmethod for estimating test MSE using the training data due to lack of test data.

The Bias-Variance Trade-Of:

$$E\left(y_0 - \hat{f}(x_0)\right)^2 = \mathrm{Var}\left(\hat{f}(x_0)\right) + \left[\mathrm{Bias}\left(\hat{f}(x_0)\right)\right]^2 + \mathrm{Var}(\epsilon).$$

This relationship is known as Bias-Variance Trade-Of which tells us that in order to minimize the expected test error, we need to select a statistical learning method that simultaneously achieves low variance and low bias. As variance and squared bias are inherently a non negative quantity, expected test MSE can never lie below Var(ϵ), the irreducible error.For flexible methods, the variance will increase and the bias will decrease.

The Bayes Classifier:

In another method of supervised learning, we should simply assign a test observation with predictor vector x_0 to the class j for which

$$Pr(Y = j|X = x_0)$$

is largest(> 0.5). This very simple classifier is called the **Bayes classifier**. The line which divides the groups is the **Bayes decision boundary**. The Bayes classifier produces the lowest possible test error rate, called the **Bayes error rate**(irreducible error) which is given by,

$$1 - E\left(\max_{j} \Pr(Y = j \mid X)\right)$$

where the expectation averages the probability over all possible values of X.

K-Nearest Neighbors (modification of Bayes Classifier):

For real data, we do not know the conditional distribution of Y given X, and so computing the Bayes classifier is impossible.

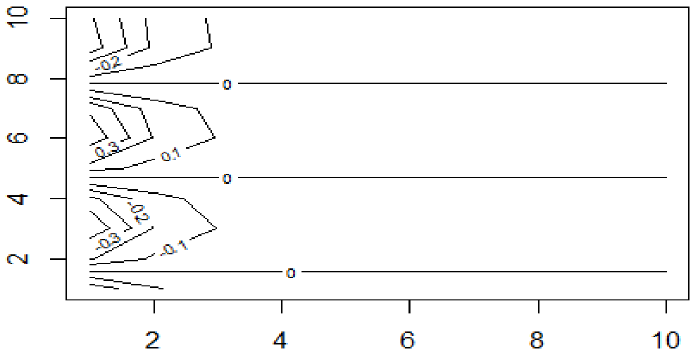
Method: First the process identify K(given) points in the training data that are closest to x_0(given), these k points are represented by N_0.It then estimates the conditional probability for class j as the fraction of points in N_0 whose response values equal j. Finally, KNN classifies the test observation x0 to the class with the largest probability from

$$\Pr(Y = j \mid X = x_0) = \frac{1}{K} \sum_{i \in \mathcal{N}_0} I\,(y_i = j)$$

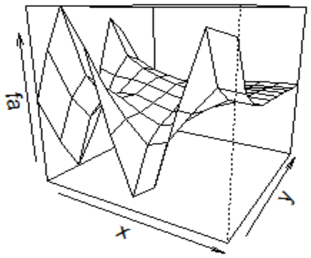
K-Nearest Neighbors will be more flexible for the lowest value of k with training error rate 0 and test error rate high.

Simple codes:

```
set.seed (3) # it reproduces the result
round(rnorm (10),2)

##  [1] -0.96 -0.29  0.26 -1.15  0.20  0.03  0.09  1.12 -1.22  1.27

x <- 1:10
y <- x
f <- outer (x, y, function (x, y) cos (y) / (1 + x^2))
contour (x, y, f) # to identify any noticeable difference in elevation of the
existing land
```



```
x <- 1:10
y <- x
f <- outer (x, y, function (x, y) cos (y) / (1 + x^2))
fa <- (f - t(f)) / 2
persp (x, y, fa , theta = 30) # produce a 3D plot
```



```
library(ISLR2)
data(Auto) # one can use read.csv(), read.table(), load() to work with data
head (Auto) # some few rows of the data
```

```
##   mpg cylinders displacement horsepower weight acceleration year origin
## 1  18         8          307        130   3504         12.0   70      1
## 2  15         8          350        165   3693         11.5   70      1
## 3  18         8          318        150   3436         11.0   70      1
## 4  16         8          304        150   3433         12.0   70      1
## 5  17         8          302        140   3449         10.5   70      1
## 6  15         8          429        198   4341         10.0   70      1
##                        name
## 1 chevrolet chevelle malibu
## 2         buick skylark 320
## 3        plymouth satellite
## 4             amc rebel sst
## 5               ford torino
## 6          ford galaxie 500
```

```
summary(Auto)
```

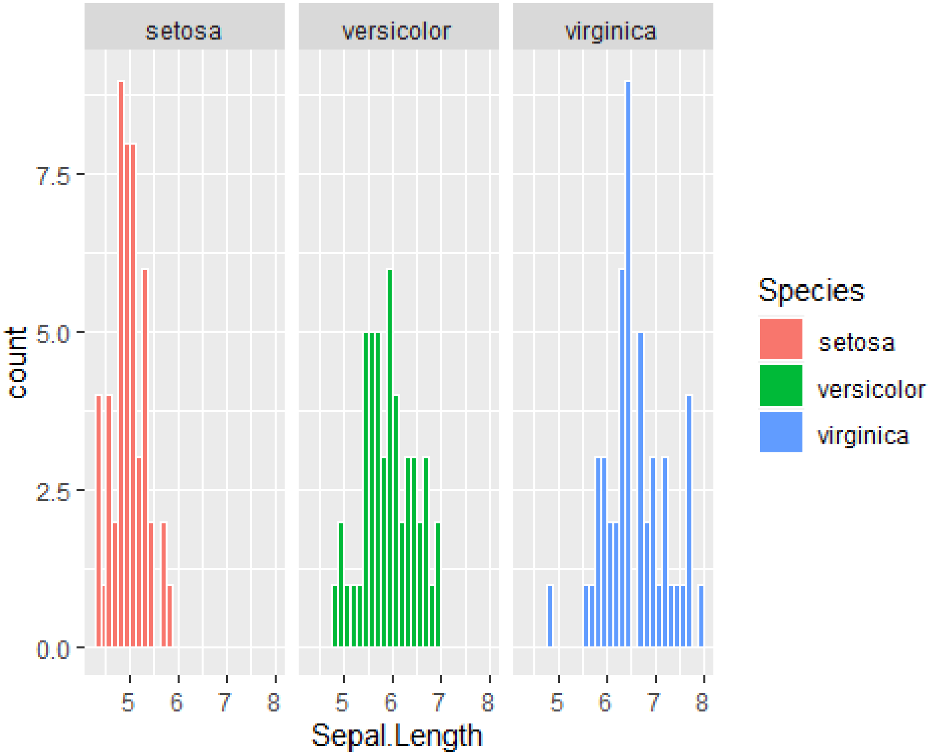
```
##       mpg          cylinders      displacement     horsepower        weigh
t
##  Min.   : 9.00   Min.   :3.000   Min.   : 68.0   Min.   : 46.0   Min.   :1
613
##  1st Qu.:17.00   1st Qu.:4.000   1st Qu.:105.0   1st Qu.: 75.0   1st Qu.:2
225
##  Median :22.75   Median :4.000   Median :151.0   Median : 93.5   Median :2
804
##  Mean   :23.45   Mean   :5.472   Mean   :194.4   Mean   :104.5   Mean   :2
```

```
978
##  3rd Qu.:29.00   3rd Qu.:8.000   3rd Qu.:275.8   3rd Qu.:126.0   3rd Qu.:3
615
##  Max.   :46.60   Max.   :8.000   Max.   :455.0   Max.   :230.0   Max.   :5
140
##
##   acceleration        year           origin                      name
##  Min.   : 8.00   Min.   :70.00   Min.   :1.000   amc matador       :  5
##  1st Qu.:13.78   1st Qu.:73.00   1st Qu.:1.000   ford pinto        :  5
##  Median :15.50   Median :76.00   Median :1.000   toyota corolla    :  5
##  Mean   :15.54   Mean   :75.98   Mean   :1.577   amc gremlin       :  4
##  3rd Qu.:17.02   3rd Qu.:79.00   3rd Qu.:2.000   amc hornet        :  4
##  Max.   :24.80   Max.   :82.00   Max.   :3.000   chevrolet chevette:  4
##                                                  (Other)           :365
```

We have already seen how to use ggplot to draw a scatterplot and boxplot, so here we provide a histogram code of ggplot.

```
library(gganimate)
data(iris)
ggplot(iris,aes(Sepal.Length,fill=Species))+
  geom_histogram(col="white")+
  facet_wrap(~Species)
```

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


setosa
versicolor
virginica
count
Sepal.Length
Species
setosa
versicolor
virginica