UNIT-I
INTRODUCTION AND LINEAR REGRESSION

Introduction to Data science

Data Science is the deep study of a large quantity of data, which involves extracting some meaningful from the raw, structured, and unstructured data. The extracting out meaningful data from large amounts use processing of data and this processing can be done using statistical techniques and algorithm, scientific techniques, different technologies, etc. It uses various tools and techniques to extract meaningful data from raw data.

Data science is a field that involves using statistical and computational techniques to extract insights and knowledge from data. It encompasses a wide range of tasks, including data cleaning and preparation, data visualization, statistical modeling, machine learning, and more.

How Data Science Works?

It's passes from many stages and every element is important. One should always follow the proper steps to reach the ladder. Every step has its value and it counts in your model.

- Problem Statement
- Data Collection
- Data Cleaning
- Data Analysis and Exploration
- Data Modelling
- Optimization and Deployment

Applications of Data Science

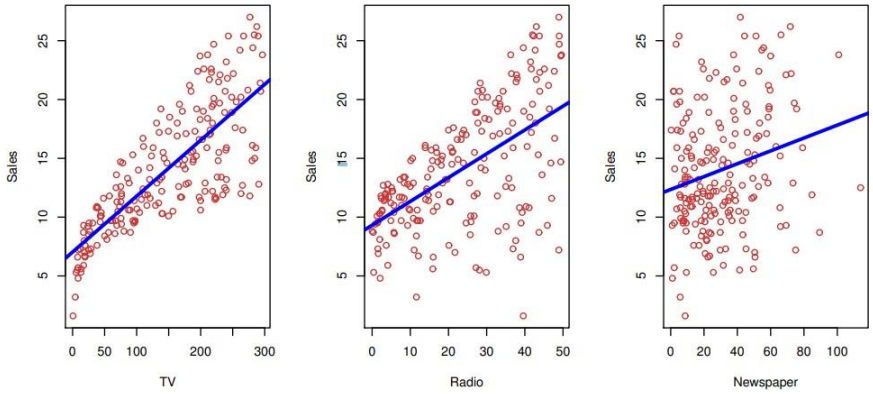
- Search Engines
- Transport
- Finance
- E-Commerce
- Health Care
- Image Recognition
- Targeting Recommendation
- Airline Routing Planning
- Gaming
- Medicine and Drug Development
- In Delivery Logistics

Statistical learning

Statistical learning refers to a vast set of tools for understanding data. These tools can be classified as supervised or unsupervised. Broadly speaking, supervised statistical learning involves building a statistical model for predicting, or estimating, an output based on one or more inputs. Problems of this nature occur in fields as diverse as business, medicine, astrophysics, and public policy. With unsupervised statistical learning, there are inputs but no

supervising output; nevertheless we can learn relationships and structure from such data.

In order to motivate our study of statistical learning, we begin with a simple example. Suppose that we are statistical consultants hired by a client to investigate the association between advertising and sales of a particular product. The Advertising data set consists of the sales of that product in 200 different markets, along with advertising budgets for the product in each of those markets for three different media: TV, radio, and newspaper. The data are displayed in Figure. It is not possible for our client to directly increase sales of the product. On the other hand, they can control the advertising expenditure in each of the three media. Therefore, if we determine that there is an association between advertising and sales, then we can instruct our client to adjust advertising budgets, thereby indirectly increasing sales. In other words, our goal is to develop an accurate model that can be used to predict sales on the basis of the three media budgets.



In this setting, the advertising budgets are *input variables* while sales is an *output variable*. The input variables are typically denoted using the symbol X, with a subscript to distinguish them. So X1 might be the TV budget, X2 the radio budget, and X3 the newspaper budget. The inputs go by different names, such as predictors, independent variables, features or sometimes just variables. The output variable—in this case, sales—is often called the response or dependent variable, and is typically denoted using the symbol Y. Throughout this book, we will use all of these terms interchangeably.

More generally, suppose that we observe a quantitative response Y and p different predictors, X1, X2,...,Xp. We assume that there is some relationship between Y and X = (X1, X2,...,Xp), which can be written in the very general form

$$Y = f(X) + \epsilon.$$

Here f is some fixed but unknown function of X1,...,Xp, and ϵ is a random error term, which is independent of X and has mean zero. In this formulation, f represents the systematic information that X provides about Y.

Why Estimate f?

There are two main reasons that we may wish to estimate f: prediction and inference.

Prediction

In many situations, a set of inputs X are readily available, but the output Y cannot be easily obtained. In this setting, since the error term averages to zero, we can predict Y using

$$\hat{Y} = \hat{f}(X)$$

Where $\hat{f}$ represents our estimate for f, and $\hat{Y}$ represents the resulting prediction for Y. In this setting, $\hat{f}$ is often treated as a black box, in the sense that one is not typically concerned with the exact form of $\hat{f}$, provided that it yields accurate predictions for Y . As an example, suppose that X1,...,Xp are characteristics of a patient's blood sample that can be easily measured in a lab, and Y is a variable encoding the patient's risk for a severe adverse reaction to a particular drug. It is natural to seek to predict Y using X, since we can then avoid giving the drug in question to patients who are at high risk of an adverse reaction—that is, patients for whom the estimate of Y is high.

The accuracy of $\hat{Y}$ as a prediction for Y depends on two quantities, which we will call the reducible error and the irreducible error. In general will not be a perfect estimate for f, and this inaccuracy will introduce some error. This error is reducible because we can potentially improve the accuracy of $\hat{f}$ by using the most appropriate statistical learning technique to estimate f. However, even if it were possible to form a perfect estimate for f, so that our estimated response took the form $\hat{Y}$ = f(X), our prediction would still have some error in it! This is because Y is also a function of ϵ, which, by definition, cannot be predicted using X. Therefore, variability associated with ϵ also affects the accuracy of our predictions. This is known as the irreducible error, because no matter how well we estimate f, we cannot reduce the error introduced by ϵ.

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

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

Inference

We are often interested in understanding the association between Y and X1,...,Xp. In this situation we wish to estimate f, but our goal is not necessarily to make predictions for Y. Now $\hat{f}$ cannot be treated as a black box, because we need to know its exact form. In this setting, one may be interested in answering the following questions:

- **Which predictors are associated with the response?** It is often the case that only a small fraction of the available predictors are substantially associated with Y. Identifying the few important predictors among a large set of possible variables can be extremely useful, depending on the application.
- **What is the relationship between the response and each predictor?** Some predictors may have a positive relationship with Y, in the sense that larger values of the predictor are associated with larger values of Y. Other predictors may have the opposite relationship. Depending on the complexity of f, the relationship between the response and a given predictor may also depend on the values of the other predictors.
- **Can the relationship between Y and each predictor be adequately summarized using a linear equation, or is the relationship more complicated?** Historically,

most methods for estimating f have taken a linear form. In some situations, such an assumption is reasonable or even desirable. But often the true relationship is more

complicated, in which case a linear model may not provide an accurate representation of the relationship between the input and output variables.

How Do We Estimate f?

Our goal is to apply a statistical learning method to the training data in order to estimate the unknown function f. In other words, we want to find a function $\hat{f}$ such that $Y \approx \hat{f}(X)$ for any observation (X, Y). Broadly speaking, most statistical learning methods for this task can be characterized as either *parametric* or *non-parametric*.

Parametric Methods

Parametric methods involve a two-step model-based approach.

1. First, we make an assumption about the functional form, or shape, of f. For example, one very simple assumption is that f is linear in X:

   $$f(X) = \beta 0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p$$

   This is a *linear model*, which will be discussed extensively in Chapter 3. Once we have assumed that *f* is linear, the problem of estimating *f* is greatly simplified. Instead of having to estimate an entirely arbitrary p-dimensional function *f*(X), one only needs to estimate the p + 1 coefficients $\beta 0, \beta 1, \ldots, \beta p$.
2. After a model has been selected, we need a procedure that uses the training data to fit or train the model. In the case of the linear model, we need to estimate the parameters $\beta_0, \beta_1, \ldots, \beta_p$. That is, we want to find values of these parameters such that

   $$Y \approx \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta p X p.$$

   The most common approach to fitting the model is referred to as (ordinary) least squares, However, least squares is one of many possible ways to fit the linear model.

Non-Parametric Methods

Non-parametric methods do not make explicit assumptions about the functional form of f. Instead they seek an estimate of f that gets as close to the data points as possible without being too rough or wiggly.

Such approaches can have a major advantage over parametric approaches: by avoiding the assumption of a particular functional form for f, they have the potential to accurately fit a wider range of possible shapes for f.

Any parametric approach brings with it the possibility that the functional form used to estimate f is very different from the true f, in which case the resulting model will not fit the data well. In contrast, non-parametric approaches completely avoid this danger, since essentially no assumption about the form of f is made. But non-parametric approaches do suffer from a major disadvantage: since they do not reduce the problem of estimating f to a small number of parameters, a very large number of observations (far more than is typically needed for a parametric approach) is required in order to obtain an accurate estimate for f.

The Trade-Off between Prediction Accuracy and Model Interpretability

Of the many methods that we examine some are less flexible, or more restrictive, in the sense that they can produce just a relatively small range of shapes to estimate f.

For example, linear regression is a relatively inflexible approach, because it can only generate linear functions such as the lines or the plane. Other methods, such as the thin plate splines are considerably more flexible because they can generate a much wider range of possible shapes to estimate f.

A representation of the tradeoff between flexibility and interpretability, using different statistical learning methods. In general, as the flexibility of a method increases, its interpretability decreases.

One might reasonably ask the following question: why would we ever choose to use a more restrictive method instead of a very flexible approach? There are several reasons that we might prefer a more restrictive model. If we are mainly interested in inference, then restrictive models are much more interpretable. For instance, when inference is the goal, the linear model may be a good choice since it will be quite easy to understand the relationship between Y and X_1, X_2,...,X_p.

when inference is the goal, there are clear advantages to using simple and relatively inflexible statistical learning methods. In some settings, however, we are only interested in prediction, and the interpretability of the predictive model is simply not of interest. For instance, if we seek to develop an algorithm to predict the price of a stock, our sole requirement for the algorithm is that it predict accurately—interpretability is not a concern.

Assessing Model Accuracy:

A wide range of statistical learning methods that extend far beyond the standard linear regression approach. *Why is it necessary to introduce so many different statistical learning approaches, rather than just a single best method?* There is no free lunch in statistics: no one method dominates all others over all possible data sets. On a particular data set, one specific method may work best, but some other method may work better on a similar but different

data set. Hence it is an important task to decide for any given set of data which method produces the best results. Selecting the best approach can be one of the most challenging parts of performing statistical learning in practice.

Measuring the Quality of Fit

In order to evaluate the performance of a statistical learning method on a given data set, we need some way to measure how well its predictions actually match the observed data. That is, we need to quantify the extent to which the predicted response value for a given observation is close to the true response value for that observation. In the regression setting, the most commonly-used measure is the *mean squared error (MSE)*, given by

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

Where $\hat{f}(x_i)$ is the prediction that $\hat{f}$ gives for the *i*th 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.

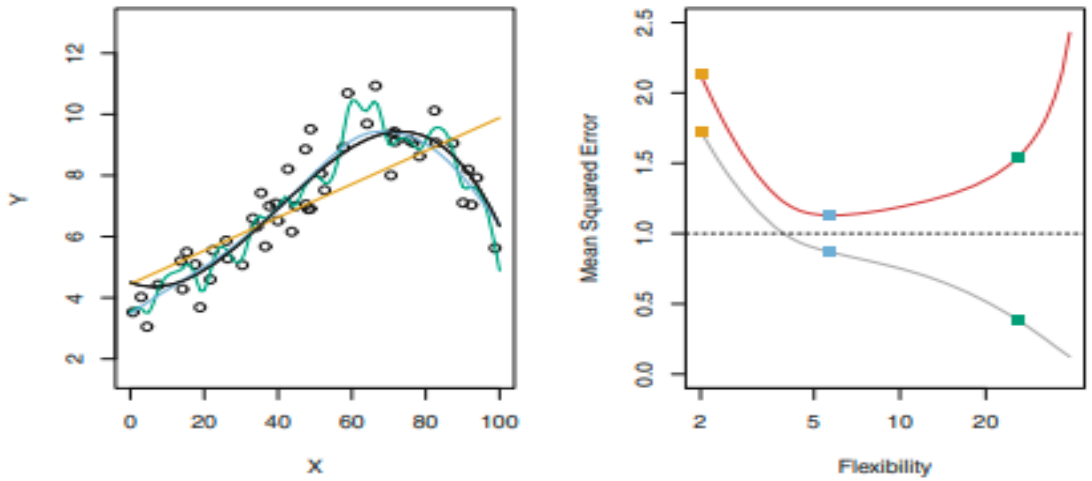
The MSE is computed using the training data that was used to fit the model, and so should more accurately be referred to as the training MSE. But in general, we do not really care how well the method works on the training data. Rather, we are interested in the accuracy of the predictions that we obtain when we apply our method to previously unseen test data.

To state it more mathematically, suppose that we fit our statistical learning method on our training observations $\{(x_1, y_1),(x_2, y_2),...,(x_n, y_n)\}$, and we obtain the estimate $\hat{f}$. We can then compute $\hat{f}(x_1)$, $\hat{f}(x_2)$,..., $\hat{f}(x_n)$. If these are approximately equal to y1, y2,...,yn, then the training MSE is small.

However, we are really not interested in whether $\hat{f}(x_i) \approx y_i$; instead, we want to know whether $\hat{f}(x_0)$ is approximately equal to y_0, where (x_0, y_0) is a previously unseen test observation not used to train the statistical learning method. We want to choose the method that gives the lowest test MSE, as opposed to the lowest training MSE.
if we had a large number of test observations, we could compute

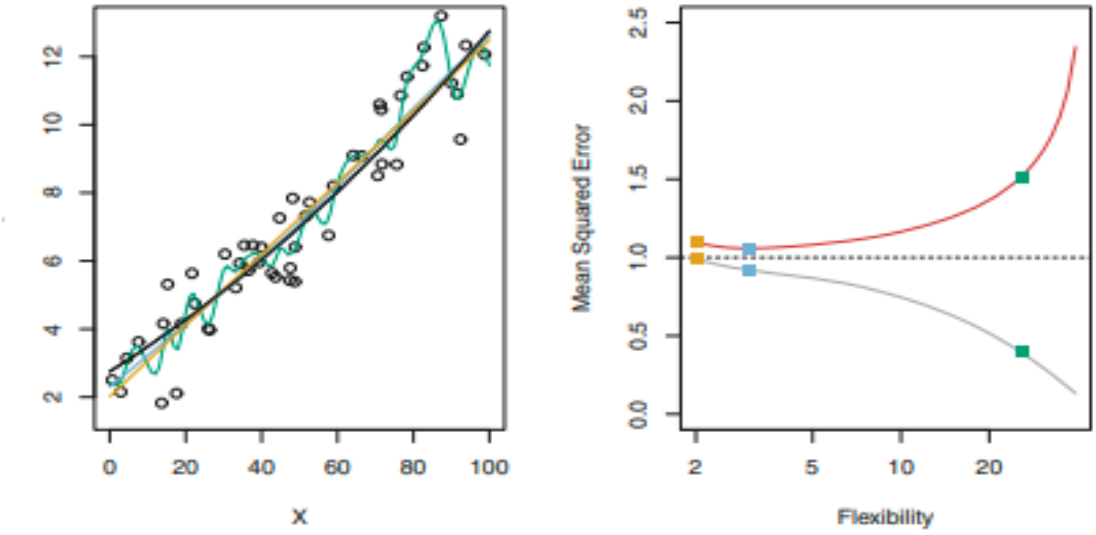
$$\text{Ave}(y_0 - \hat{f}(x_0))^2,$$

the average squared prediction error for these test observations (x0, y0). We'd like to select the model for which the average of this quantity—the test MSE—is as small as possible.



. Left: Data simulated from f, shown in black. Three estimates of f are shown: the linear regression line (orange curve), and two smoothing spline fits (blue and green curves). Right: Training MSE (grey curve), test MSE (red curve), and minimum possible test MSE over all methods (dashed line). Squares represent the training and test MSEs for the three fits shown in the left-hand panel.

The grey curve displays the average training MSE as a function of flexibility, or more formally the degrees of freedom, for a number of smoothing splines. The degrees of freedom is a quantity that summarizes the flexibility of a curve. The horizontal dashed line indicates $\mathrm{Var}(\epsilon)$, the irreducible error.



Details using a different true f that is much closer to linear. In this setting, linear regression provides a very good fit to the data.

One important method is cross-validation, which is a method for estimating test MSE using the training data.

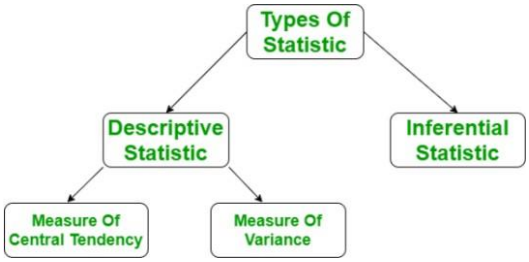
Descriptive Statistics

In Descriptive statistics, we are describing our data with the help of various representative methods like by using charts, graphs, tables, excel files etc. In descriptive statistics, we describe our data in some manner and present it in a meaningful way so that it

can be easily understood. Most of the times it is performed on small data sets and this analysis helps us a lot to predict some future trends based on the current findings. Some measures that are used to describe a data set are measures of central tendency and measures of variability or dispersion.

Types of Descriptive statistic:

1. Measure of central tendency
2. Measure of variability



1. Measure of central tendency:

It represents the whole set of data by single value.It gives us the location of central points. There are three main measures of central tendency:

a) Mean
b) Mode
c) Median

a) Mean:

It is the sum of observation divided by the total number of observations. It is also defined as average which is the sum divided by count.

$$\text{Mean } (\bar{x}) = \frac{\sum x}{n}$$

where, n = number of terms

R Program:

```
myData = read.csv("D:/ccc.csv")
print(head(myData))
mean = mean(myData$Age)
print(mean)
```

Output :

```
> myData = read.csv("D:/ccc.csv")
> print(head(myData))
  Product Age   Gender Education Martial.status Usage Fitness Income Miles
1   TM195  18   Male          14          single     3       4  29562   112
2   TM196  19   Male          15          single     2       3  31836    75
3   TM197  19 Female          14       partnered     4       3  30699    66
4   TM198  20   Male          12          single     3       3  32973    85
5   TM199  20   Male          13       partnered     4       2  35247    47
6   TM200  20 Female          14       partnered     3       3  32973    66
> mean = mean(myData$Age)
> print(mean)
[1] 19.33333
```

b) Mode:

It is the value that has the highest frequency in the given data set. The data set may have no mode if the frequency of all data points is the same. Also, we can have more than one mode if we encounter two or more data points having the same frequency.

R Program

```
myData = read.csv("D:/ccc.csv")
library(modeest)
mode = mfv(myData$Age)
print(mode)
```

Output:

```
> library(modeest)
> mode = mfv(myData$Age)
> print(mode)
[1] 20
```

R Program

```
myData = read.csv("D:/ccc3.csv")
print(head(myData))
library(modeest)
mode = mfv(myData$Age)
print(mode)
```

Output:

```
> print(head(myData))
  Product Age   Gender Education Martial.status Usage Fitness Income Miles
1   TM195  18   Male          14          single     3       4  29562   112
2   TM196  19   Male          15          single     2       3  31836    75
3   TM197  21 Female          14       partnered     4       3  30699    66
4   TM198  22   Male          12          single     3       3  32973    85
5   TM199  23   Male          13       partnered     4       2  35247    47
6   TM200  20 Female          14       partnered     3       3  32973    66
> library(modeest)
> mode = mfv(myData$Age)
> print(mode)
[1] 18 19 20 21 22 23
```

R Program

```
myData = read.csv("D:/ccc2.csv")
print(head(myData))
library(modeest)
mode = mfv(myData$Age)
print(mode)
```

Output:

```
> print(head(myData))
  Product Age   Gender Education Martial.status Usage Fitness Income Miles
1   TM195  18   Male          14          single     3       4  29562   112
2   TM196  19   Male          15          single     2       3  31836    75
3   TM197  19 Female          14       partnered     4       3  30699    66
4   TM198  18   Male          12          single     3       3  32973    85
5   TM199  23   Male          13       partnered     4       2  35247    47
6   TM200  20 Female          14       partnered     3       3  32973    66
> library(modeest)
> mode = mfv(myData$Age)
> print(mode)
[1] 18 19
```

c) Median:

It is the middle value of the data set. It splits the data into two halves. If the number of elements in the data set is odd then the centre element is median and if it is even then the median would be the average of two central elements.

Odd	Even
$\frac{n+1}{2}$	$\frac{n}{2}, \frac{n}{2}+1$

where, n=number of terms

Program:

```
myData = read.csv("D:/ccc.csv")
print(head(myData))
median = median(myData$Age)
print(median)
```

Output:

```
> myData = read.csv("D:/ccc.csv")
> print(head(myData))
  Product Age   Gender Education Martial.status Usage Fitness Income Miles
1   TM195  18   Male          14          single     3       4  29562   112
2   TM196  19   Male          15          single     2       3  31836    75
3   TM197  19 Female          14       partnered     4       3  30699    66
4   TM198  20   Male          12          single     3       3  32973    85
5   TM199  20   Male          13       partnered     4       2  35247    47
6   TM200  20 Female          14       partnered     3       3  32973    66
> median = median(myData$Age)
> print(median)
[1] 19.5
```

R Program

```
myData = read.csv("D:/ccc2.csv")
print(head(myData))
median = median(myData$Age)
print(median)
```

Output:

```
> myData = read.csv("D:/ccc2.csv")
> print(head(myData))
  Product Age   Gender Education Martial.status Usage Fitness Income Miles
1   TM195  18   Male          14          single     3       4  29562   112
2   TM196  19   Male          15          single     2       3  31836    75
3   TM197  19 Female          14       partnered     4       3  30699    66
4   TM198  18   Male          12          single     3       3  32973    85
5   TM199  23   Male          13       partnered     4       2  35247    47
6   TM200  20 Female          14       partnered     3       3  32973    66
> median = median(myData$Age)
> print(median)
[1] 19
```

R Program

```
myData = read.csv("D:/ccc2.csv")
print(head(myData))
median = median(myData$Age)
print(median)
```

Output:

```
> myData = read.csv("D:/ccc2.csv")
> print(head(myData))
  Product Age   Gender Education Martial.status Usage Fitness Income Miles
1   TM195  18   Male          14          single     3       4  29562   112
2   TM196  19   Male          15          single     2       3  31836    75
3   TM197  25 Female          14       partnered     4       3  30699    66
4   TM198  20   Male          12          single     3       3  32973    85
5   TM199  33   Male          13       partnered     4       2  35247    47
> median = median(myData$Age)
> print(median)
[1] 20
```

2. Measure of variability:

Measure of variability is known as the spread of data or how well is our data is distributed. The most common variability measures are:

a) Range
b) Variance
c) Standard deviation

a) Range

The range describes the difference between the largest and smallest data point in our data set. The bigger the range, the more is the spread of data and vice versa.

Range = Largest data value – smallest data value

R Program:

```
myData = read.csv("D:/ccc.csv")
print(head(myData))
max = max(myData$Age)
min = min(myData$Age)
range = max - min
cat("Range is:\n")
print(range)
```

Output:

```
> myData = read.csv("D:/ccc.csv")
> print(head(myData))
  Product Age   Gender Education Martial.status Usage Fitness Income Miles
1   TM195  18   Male          14          single     3       4  29562   112
2   TM196  19   Male          15          single     2       3  31836    75
3   TM197  19 Female          14       partnered     4       3  30699    66
4   TM198  20   Male          12          single     3       3  32973    85
5   TM199  20   Male          13       partnered     4       2  35247    47
6   TM200  20 Female          14       partnered     3       3  32973    66
> max = max(myData$Age)
> min = min(myData$Age)
> range = max - min
> cat("Range is:\n")
Range is:
> print(range)
[1] 2
```

Or

R Program

```
r = range(myData$Age)
print(r)
```

Output:

```
> r = range(myData$Age)
> print(r)
[1] 18 20
```

b) Variance

It is defined as an average squared deviation from the mean. It is being calculated by finding the difference between every data point and the average which is also known as the mean, squaring them, adding all of them and then dividing by the number of data points present in our data set.

$$\sigma^2 = \frac{\sum(\chi - \mu)^2}{N}$$

Program:

```
myData = read.csv("D:/ccc.csv")
variance = var(myData$Age)
print(variance)
```

Output:

```
> myData = read.csv("D:/ccc.csv")
> variance = var(myData$Age)
> print(variance)
[1] 0.6666667
>
```

c) Standard Deviation

It is defined as the square root of the variance. It is being calculated by finding the Mean, then subtract each number from the Mean which is also known as average and square the result. Adding all the values and then divide by the no of terms followed the square root.

$$\sigma = \sqrt{\frac{\Sigma (x - \mu)^2}{N}}$$

Program:

```
myData = read.csv("D:/ccc.csv")
std = sd(myData$Age)
print(std)
```

Output:

```
> myData = read.csv("D:/ccc.csv")
> std = sd(myData$Age)
> print(std)
[1] 0.8164966
```

Linear Regression:

Linear regression is a useful tool for predicting a quantitative response. The term regression is used when you try to find the relationship between variables. In Machine Learning and in statistical modeling, that relationship is used to predict the outcome of events.

Here are a few important questions that we might seek to address:

1. Is there a relationship between advertising budget and sales?

Our first goal should be to determine whether the data provide evidence of an association

between advertising expenditure and sales. If the evidence is weak, then one might argue that no money should be spent on advertising!

2. How strong is the relationship between advertising budget and sales?

   Assuming that there is a relationship between advertising and sales, we would like to know the strength of this relationship.

3. Which media contribute to sales?

   Do all three media—TV, radio, and newspaper—contribute to sales, or do just one or two of the media contribute? To answer this question, we must find a way to separate out the individual effects of each medium when we have spent money on all three media.

4. How accurately can we estimate the effect of each medium on sales?

   For every dollar spent on advertising in a particular medium, by what amount will sales increase? How accurately can we predict this amount of increase?

5. How accurately can we predict future sales?

   For any given level of television, radio, or newspaper advertising, what is our prediction for sales, and what is the accuracy of this prediction?

6. Is the relationship linear?

   If there is approximately a straight-line relationship between advertising expenditure in the various media and sales, then linear regression is an appropriate tool. If not, then it may still be possible to transform the predictor or the response so that linear regression can be used.

7. Is there synergy among the advertising media?

   Perhaps spending $50,000 on television advertising and $50,000 on radio advertising results in more sales than allocating $100,000 to either television or radio individually. In marketing, this is known as a synergy effect, while in statistics it is called an interaction effect.

   It turns out that linear regression can be used to answer each of these questions.

Simple Linear Regression:

Simple Linear Regression is a type of Regression algorithms that models the relationship between a dependent variable and a single independent variable. The relationship shown by a Simple Linear Regression model is linear or a sloped straight line, hence it is called Simple Linear Regression.

Simple linear regression lives up to its name: it is a very straightforward simple linear approach for predicting a quantitative response Y on the basis of a sin- regression gle predictor variable X.

Mathematically, we can write this linear relationship as

$$Y \approx \beta_0 + \beta_1 X.$$

You might read "≈" as "is approximately 8eighbou as". We will sometimes describe by saying that we are regressing Y on X (or Y onto X). For example, X may represent TV

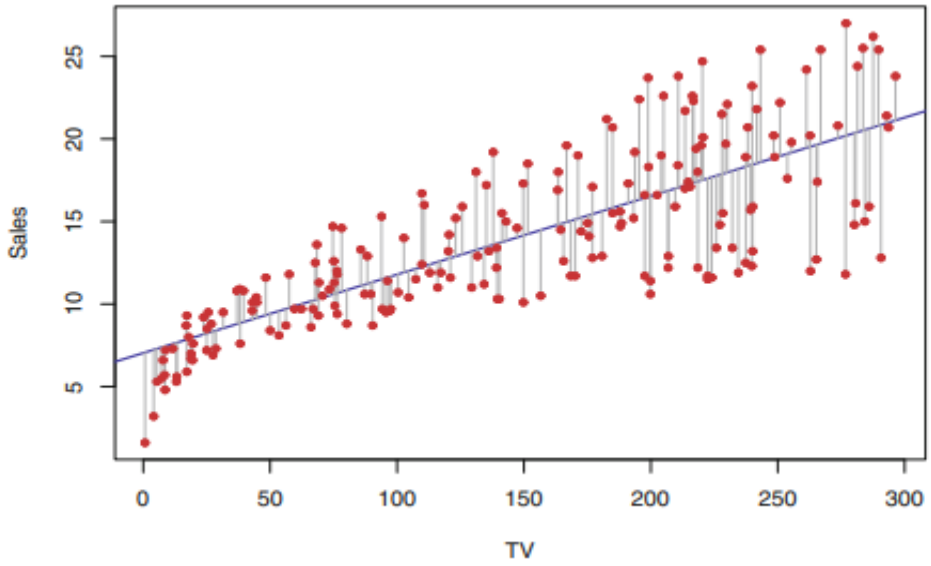
advertising and Y may represent sales. Then we can regress sales onto TV by fitting the model

$$sales \approx \beta_0 + \beta_1 \times TV$$

Estimating the Coefficients

In practice, β0 and β1 are unknown. So before we can use to make predictions, we must use data to estimate the coefficients. Let (x_1, y_1), (x_2, y_2),..., (x_n, y_n) represent n observation pairs, each of which consists of a measurement of X and a measurement of Y.

Our goal is to obtain coefficient estimates $\hat{\beta}0$ and $\hat{\beta}1$ such that the linear model fits the available data well—that is, so that $y_i \approx \hat{\beta}_0 + \hat{\beta}_1 x_i$ for i = 1,...,n. we want to find an intercept $\hat{\beta}0$ and a slope $\hat{\beta}1$ such that the resulting line is as close as possible to the n = 200 data points. There are a number of ways of measuring closeness. However, by far the most common approach involves minimizing the least squares criterion.



For the Advertising data, the least squares fit for the regression of sales onto TV is shown. The fit is found by minimizing the sum of squared errors. Each grey line segment represents an error, and the fit makes a compromise by averaging their squares. In this case a linear fit captures the essence of the relationship, although it is somewhat deficient in the left of the plot.

Let $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$ be the prediction for Y based on the ith value of X. Then $e_i = y_i - \hat{y}_i$ represents the ith residual—this is the difference between residual the ith observed response value and the ith response value that is predicted by our linear model.

We define the residual sum of squares (RSS) as residual sum of squares

$RSS = e_1^2 + e_2^2 + \cdots + e_n^2$, or equivalently as

$RSS = (y_1 - \hat{\beta}_0 - \hat{\beta}_1 x_1)^2 + (y_2 - \hat{\beta}_0 - \hat{\beta}_1 x^2)^2 + ... + (y_n - \hat{\beta}_0 - \hat{\beta}_1 x_n)^2$

The least squares approach chooses $\hat{\beta}0$ and $\hat{\beta}1$ to minimize the RSS. Using some calculus, one can show that the minimizers are

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2},$$
$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x},$$

where $\bar{y} \equiv \frac{1}{n}\sum_{i=1}^{n} y_i$ and $\bar{x} \equiv \frac{1}{n}\sum_{i=1}^{n} x_i$ are the sample means.

The simple linear regression fit to the Advertising data, where βˆ0 = 7.03 and βˆ1 = 0.0475. In other words, according to this approximation, an additional $1,000 spent on TV advertising is associated with selling approximately 47.5 additional units of the product.

Assessing the Accuracy of the Model

It is natural to want to quantify the extent to which the model fits the data. The quality of a linear regression fit is typically assessed using two related quantities: the residual standard error (RSE) and the R^2 R^2 statistic.

Residual Standard Error

The model that associated with each observation is an error term . Due to the presence of these error terms, even if we knew the true regression line (i.e. even if β_0 and β_1 were known), we would not be able to perfectly predict Y from X. The RSE is an estimate of the

standard deviation of ϵ. It is the average amount that the response will deviate from the true regression line. It is computed using the formula.

$$\text{RSE} = \sqrt{\frac{1}{n-2}\text{RSS}} = \sqrt{\frac{1}{n-2}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}.$$

Note that RSS was defined and is given by the formula

R^2 Statistic

The RSE provides an absolute measure of lack of fit of the model (3.5) to the data. But since it is measured in the units of Y , it is not always clear what constitutes a good RSE. The R2 statistic provides an alternative measure of fit. It takes the form of a proportion—the proportion of variance explained—and so it always takes on a value between 0 and 1, and is independent

$$\text{RSS} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2.$$

of the scale of Y

To calculate R^2, we use the formula

$$R^2 = \frac{\text{TSS} - \text{RSS}}{\text{TSS}} = 1 - \frac{\text{RSS}}{\text{TSS}}$$

where $\text{TSS} = \sum(y_i - \bar{y})^2$ is the *total sum of squares*,

R^2 measures the proportion of variability in Y that can be explained using X. An R^2 statistic that is close to 1 indicates that a large proportion of the variability in the response has been explained by the regression. A number near 0 indicates that the regression did not explain much of the variability in the response.

The R^2 statistic is a measure of the linear relationship between X and Y . Recall that correlation, defined as

$$\mathrm{Cor}(X,Y) = \frac{\sum_{i=1}^{n}(x_i-\overline{x})(y_i-\overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\overline{x})^2}\sqrt{\sum_{i=1}^{n}(y_i-\overline{y})^2}},$$

is also a measure of the linear relationship between X and Y. This suggests that we might be able to use r = Cor(X, Y) instead of R^2 in order to assess the fit of the linear model.

Multiple Linear Regression

Simple linear regression is a useful approach for predicting a response on the basis of a single predictor variable. However, in practice we often have more than one predictor. For example, in the Advertising data, we have examined the relationship between sales and TV advertising. We also have data for the amount of money spent advertising on the radio and in newspapers, and we may want to know whether either of these two media is associated with sales.

Instead of fitting a separate simple linear regression model for each predictor, a better approach is to extend the simple linear regression model so that it can directly accommodate multiple predictors. We can do this by giving each predictor a separate slope coefficient in a single model. In general, suppose that we have p distinct predictors.
Then the multiple linear regression model takes the form

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \epsilon,$$

where X_j represents the jth predictor and β_j quantifies the association between that variable and the response. We interpret βj as the average effect on Y of a one unit increase in Xj , holding all other predictors fixed. In the advertising example,

$$\texttt{sales} = \beta_0 + \beta_1 \times \texttt{TV} + \beta_2 \times \texttt{radio} + \beta_3 \times \texttt{newspaper} + \epsilon.$$

Estimating the Regression Coefficients

As was the case in the simple linear regression setting, the regression coefficients $\beta_0, \beta_1,\ldots,\beta_p$ are unknown, and must be estimated. Given estimates $\hat{\beta}_0, \hat{\beta}_1,\ldots, \hat{\beta}_p$, we can make predictions using the formula

$$\begin{aligned} \text{RSS} &= \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \\ &= \sum_{i=1}^{n}(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_{i1} - \hat{\beta}_2 x_{i2} - \cdots - \hat{\beta}_p x_{ip})^2. \end{aligned}$$

For the Advertising data, least squares coefficient estimates of the multiple linear regression of number of units sold on radio, TV, and newspaper advertising budgets.

	Coefficient	Std. error	t-statistic	p-value
Intercept	2.939	0.3119	9.42	< 0.0001
TV	0.046	0.0014	32.81	< 0.0001
radio	0.189	0.0086	21.89	< 0.0001
newspaper	−0.001	0.0059	−0.18	0.8599

In the multiple regression setting, the coefficient for newspaper represents the average effect of increasing newspaper spending by $1,000 while holding TV and radio fixed.

Consider the correlation matrix for the three predictor variables and response variable, displayed in Table

	TV	radio	newspaper	sales
TV	1.0000	0.0548	0.0567	0.7822
radio		1.0000	0.3541	0.5762
newspaper			1.0000	0.2283
sales				1.0000

Notice that the correlation between radio and newspaper is 0.35. This reveals a tendency to spend more on newspaper advertising in markets where more is spent on radio advertising. Now suppose that the multiple regression is correct and newspaper advertising has no direct impact on sales, but radio advertising does increase sales.

K-Nearest neighbor (KNN) Algorithm:

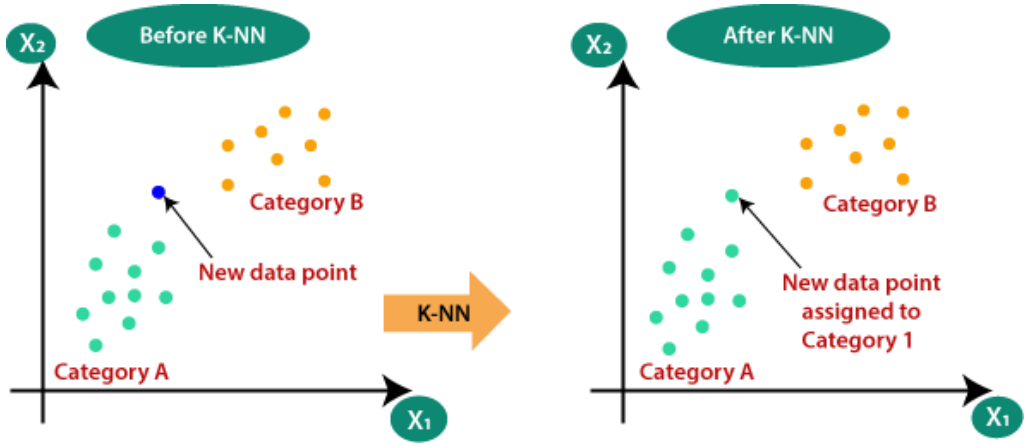
- K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.
- K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.
- K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.
- K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.
- K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data.
- It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs

an action on the dataset.

- KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.

- **Example:** Suppose, we have an image of a creature that looks similar to cat and dog, but we want to know either it is a cat or dog. So for this identification, we can use the KNN algorithm, as it works on a similarity measure. Our KNN model will find the similar features of the new data set to the cats and dogs images and based on the most similar features it will put it in either cat or dog category.

Suppose there are two categories, i.e., Category A and Category B, and we have a new data point x1, so this data point will lie in which of these categories. To solve this type of problem, we need a K-NN algorithm. With the help of K-NN, we can easily identify the category or class of a particular dataset. Consider the below diagram:



How does K-NN work?

The K-NN working can be explained on the basis of the below algorithm:

Step-1: Select the number K of the neighbors.

Step-2: Calculate the Euclidean distance of K number of neighbors.

Step-3: Take the K nearest neighbors as per the calculated Euclidean distance.

Step-4: Among these k neighbors, count the number of the data points in each category.

Step-5: Assign the new data points to that category for which the number of the neighbor is maximum.

Step-6: Our model is ready.

K-Nearest neighbor (KNN) regression Algorithm

In k-NN regression, the output is the property value for the object. This value is the average of the values of k nearest neighbors**.**

Training algorithm:

- For each training example $\langle x, f(x) \rangle$, add the example to the list *training_examples*

Classification algorithm:

- Given a query instance x_q to be classified,
  - Let $x_1 \ldots x_k$ denote the k instances from *training_examples* that are nearest to x_q
  - Return

$$\hat{f}(x_q) \leftarrow \frac{\sum_{i=1}^{k} f(x_i)}{k}$$