REGRESSION ANALYSIS FOR DATA SCIENCE

KABARAK UNIVERSITY

Artificial Intelligence & Data Science Bootcamp

A. WAGALA, PhD

January 2023

Contents

List of Figures

List of Tables

1 Brief Introduction

Data science: is the study of data to extract meaningful insights for business or organization. It is a multidisciplinary approach that combines principles and practices from the fields of mathematics, statistics, artificial intelligence, and computer engineering to analyze large amounts of data. see(https://aws.amazon.com/what-is/data-science/).

Data science is an emerging new field that

- is extremely transdisciplinary– bridging between the theoretical, computational, experimental, and biosocial areas.
- deals with enormous amounts of complex, incongruent, and dynamic data from multiple sources.
- aims to develop algorithms, methods, tools, and services capable of ingesting such datasets and generating semi-automated decision support systems. These may include: data mining for patterns, predict expected outcomes, suggest clustering or labeling of retrospective or prospective observations, compute data signatures or fingerprints, extract valuable information, and offer evidence-based actionable knowledge.

Data science techniques often involve data manipulation (wrangling), data harmonization and aggregation, exploratory or confirmatory data analyses, predictive analytics, validation, and fine-tuning.

Predictive Analytics: is the process of utilizing advanced mathematical formulations, powerful statistical computing algorithms, efficient software tools and services to forecast trends, predict patterns in the data, or prognosticate the process behavior either within the range or outside the range of the observed data (e.g., in the future, or at locations where data may not be available).

2 What is Regression Analysis?

Regression analysis is a form of predictive modelling technique which investigates the relationship between a dependent (target) and independent variable (s) (predictor).

3 When should Regression Analysis be Used?

The regression analysis in general can be used mainly when

- When one suspects that there is a significant relationships between dependent variable and independent variable.
- When the strength of impact of multiple independent variables on a dependent variable.
- By performing the regression, we can confidently determine the most important factor, the least important factor, and how each factor is affecting the other factors.

Regression is a measurement of relationship between a dependent variable (value to be predicted) and a group of independent variables (predictors similar to features). We assume the relationship between our dependent variable and independent variables follows a predefined model.

3.1 Types of Regression Models

Some of the commonly used regression models include:

- Linear Regression
- Logistic Regression
- Polynomial Regression
- Support Vector Regression
- Decision Tree Regression
- Random Forest Regression
- Ridge Regression
- Lasso Regression
- ElasticNet Regression

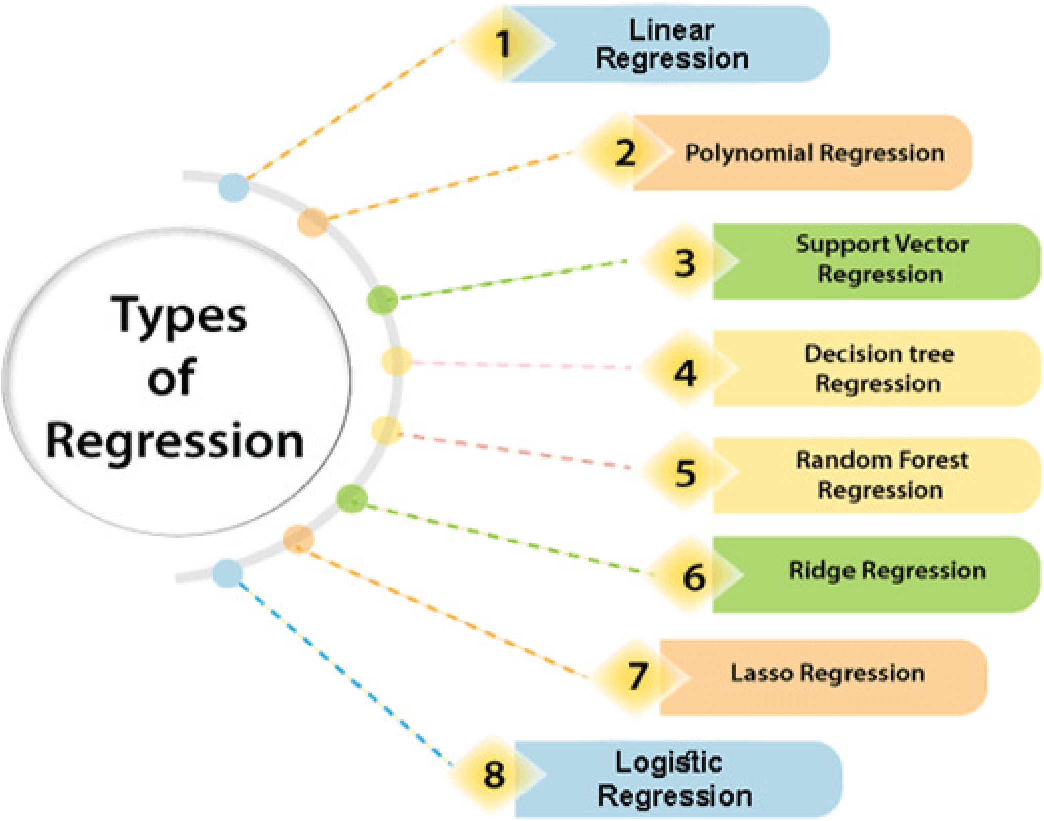

Figure 1: Types of Regression Models

4 Linear regression models

Linear regression is a statistical regression method which is used for predictive analysis. It is one of the very simple and easy algorithms which works on regression and shows the relationship between the continuous variables. It shows the linear relationship between the independent variable (X-axis) and the dependent variable (Y-axis), hence the name *linear regression.* Assumptions of Linear regression models include

- Multivariate normality,
- No or little multicollinearity,
- No auto-correlation, independence,
- Homoscedasticity

4.1 Simple Linear Regression

The simplest case of regression modeling involves a single predictor

$$y = \beta_0 + \beta_1 x + e \tag{1}$$

In equation 1, β_0 is our intercept while β_1 is the slope. That is an equation form of the simple linear regression model. If we know a and b, for any given x (input) we can predict y (output) via the above formula. If we plot x and y in a 2D coordinate system, the model is represented as a straight line. Consider the following examples

Example 1

Here, we use some data with the values of x and y inR as

```
#EXAMPLE 1, data input
x <- c(151, 174, 138, 186, 128, 136, 179, 163, 152, 131)
y <- c(63, 81, 56, 91, 47, 57, 76, 72, 62, 48)
```

We then do a scatter plot and fit a regression line using the following batch of codes.

```
# Plot the chart.
plot(y,x,col = "blue",main = "Height vs Weight Regression",
abline(lm(x˜y)),cex = 1.3,pch = 16,xlab = "Weight in (Kg)",ylab = "Height in cm")
```

The obtained graph is presented below.

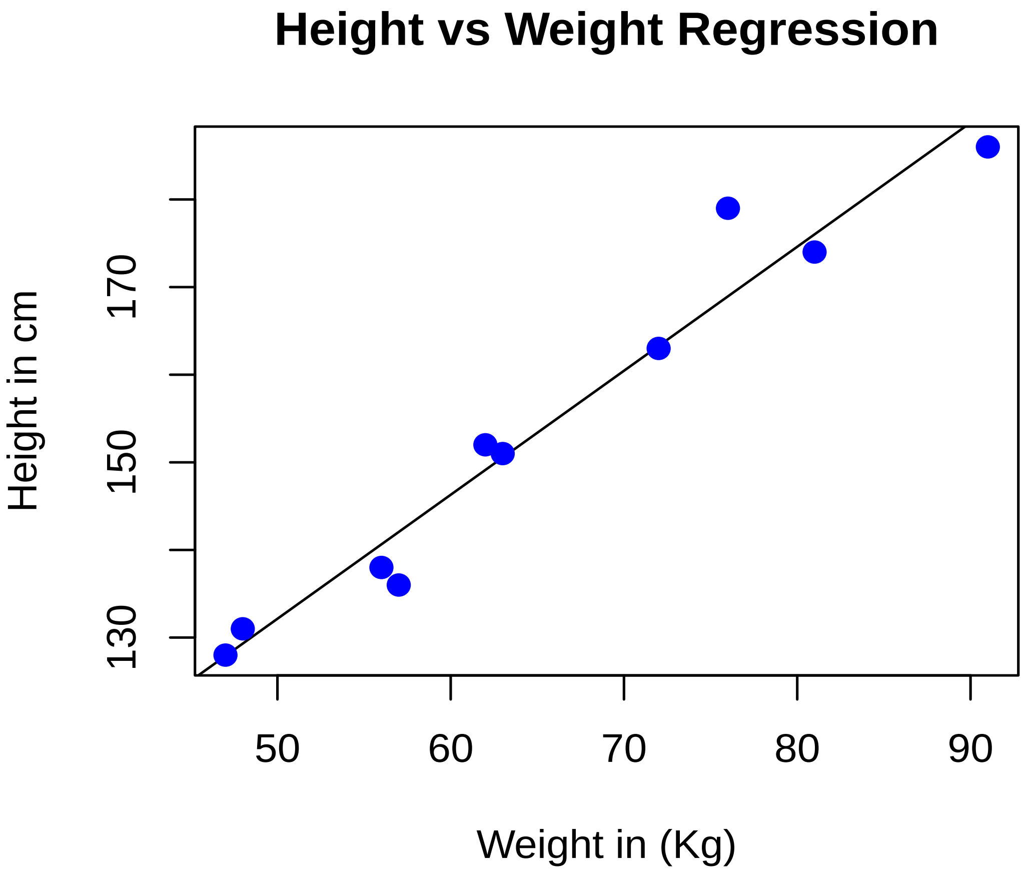

Figure 2: A scatter plot for a Simple Linear Regression Model for Height and Weight

4.2 Example 2

Consider the two variables “hospital charges” or `CHARGES` (independent variable) and length of stay in the hospital or `LOS` (predictor) found in the `heart attack data`.

```
#EXAMPLE 2
#heart_attack<- read.csv("C:/Users/HeartAttack_Data.csv")
heart_attack$CHARGES<-as.numeric(heart_attack$CHARGES)
heart_attack<-heart_attack[complete.cases(heart_attack), ]
fit1<-lm(CHARGES~LOS, data=heart_attack)
par(cex=.8)
plot(heart_attack$LOS, heart_attack$CHARGES,
xlab="LOS", ylab = "CHARGES")
abline(fit1, lwd=2)
print(summary(fit1))
```

The resulting scatter plot is presented in Figure 3

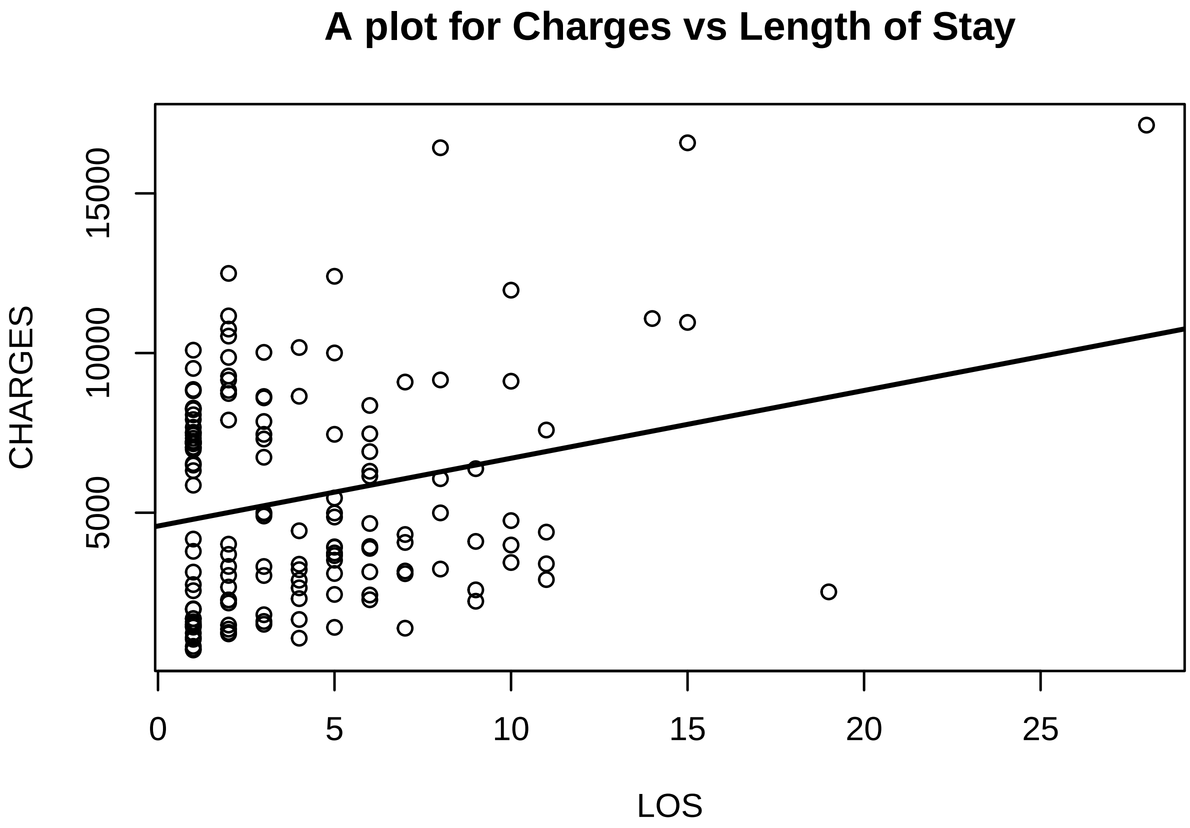

Figure 3: A scatter plot for a Simple Linear Regression

The output for the fitted model is found as

```
Call:
lm(formula = CHARGES ~ LOS, data = heart_attack)

Residuals:
Min      1Q  Median      3Q     Max
```

```
-6095.2 -2849.6  -995.1  2652.8 10148.0

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept)  4582.70     399.64  11.467  < 2e-16 ***
LOS           212.29      69.53   3.053  0.00269 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3359 on 146 degrees of freedom
Multiple R-squared:  0.06001,Adjusted R-squared:  0.05357
F-statistic: 9.321 on 1 and 146 DF,  p-value: 0.002692
```

Or simply put as

$$y = 4582.70 + 212.29x \tag{2}$$

The equation 2 can be used for prediction purposes. For instance, to predict `CHARGES` when `LOS=35`, we use the code

```
#To predict CHARGES FOR LOS=35
AA <- data.frame(LOS = 35)
result <- predict(fit1,AA)
print(result)
```

The result is `12012.74` .

Ordinary Least Squares Estimation (OLSE)

The least-squares procedure for fitting a line through a set of n data points is similar to the method that we might use if we fit a line by eye; that is, we want the differences between the observed values and corresponding points on the fitted line to be “small” in some overall sense. A convenient way to accomplish this, and one that yields estimators with good properties, is to minimize the sum of squares of the vertical deviations from the fitted line. That is- minimizing the sum of the squared errors – (the sum of squared vertical distances between each point on the scatter plot and its predicted value on the regression line), see Figure 4

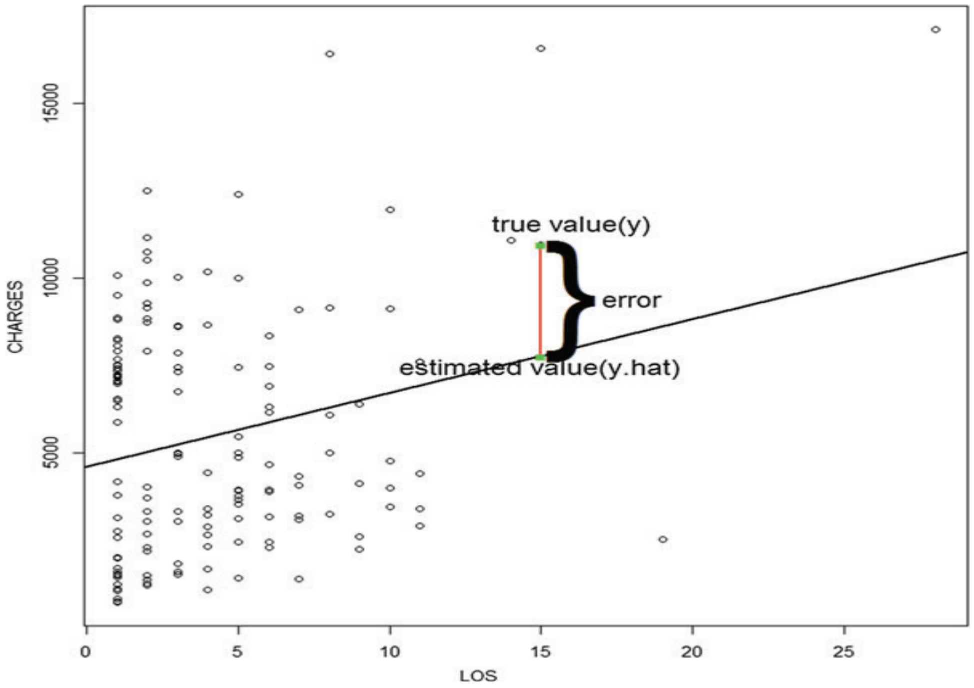

Figure 4: A representation of the residuals (the difference between observed and predicted values)

If

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x \tag{3}$$

is the $\hat{y}_i$ is the predicted value of the i^{th} value of y (when $x = x_i$), then the deviation (sometimes called the error) of the observed value of y_i from $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x$ is the difference $y_i - \hat{y}_i$) and the sum of squares of deviations (also known as sum of squares due to error (SSE)) to be minimized is

$$SSE = e_i^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} \left[y_i - (\hat{\beta}_0 + \hat{\beta}_1 x) \right]^2 \tag{4}$$

Some simple mathematical operations to minimize the sum square error yield the following solution for the slope parameter $\hat{\beta}_1$ as

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n} (x_i - \bar{x}) (y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2} \tag{5}$$

while for the intercept $\hat{\beta}_0$ we have

$$\bar{\beta}_0 = \hat{y} - \hat{\beta}_1 \bar{x} \tag{6}$$

We need to note the following.

- $Var(x) = \frac{1}{(n-1)} \sum_{i=1}^{n} (x_i - \bar{x})^2$
- $Cov(x, y) = \frac{1}{(n-1)} \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})$

Therefore Equation 5 can be rewritten as

$$\bar{\beta}_1 = \frac{Cov(x, y)}{Var(x)} \tag{7}$$

From Example 1, we can run the following codes to get the closed form estimates for the regression co-efficients;

```
> ###Closed form analysis
> beta1<-cov(x, y)/var(x)
> beta1
[1] 0.6746104
> beta0<-mean(y)-beta1*mean(x)
> beta0
[1] -38.45509
```

We can apply the input `R` function `lm()`to get much more better results

```
> # Apply the lm() function.
> rel <- lm(y~x)
> print(summary(rel))

Call:
lm(formula = y ~ x)

Residuals:
Min      1Q  Median      3Q     Max
-6.3002 -1.6629  0.0412  1.8944  3.9775
```

```
Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) -38.45509    8.04901  -4.778  0.00139 **
x             0.67461    0.05191  12.997 1.16e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.253 on 8 degrees of freedom
Multiple R-squared:  0.9548,Adjusted R-squared:  0.9491
F-statistic: 168.9 on 1 and 8 DF,  p-value: 1.164e-06
```

The same can be done for the Heart Attack data. (Do it as an assignment).

Correlation

We can calculate the correlation, which indicates how closely the relationship between two variables follows a straight line.

$$\rho_{x,y} = \frac{Cov(x, y)}{\sqrt{Var(x)Var(y)}} \tag{8}$$

If the correlation is a positive number then the variables have a positive relationship. If we have a negative correlation estimate, it suggests a negative linear association. We have a weak association when $0.1 \leq \rho_{x,y} \leq 0.3$, a moderate association for $0.3 \leq \rho_{x,y} \leq 0.5$, and a strong association for $0.5 \leq \rho_{x,y} \leq 1$. If the correlation is below 0.1 then it suggests little to no linear relation between the variables. From Example 1, we get

```
> ##correlation
> (r<-cov(x, y)/(sd(x)*sd(y)))
[1] 0.9771296
> cor(x,y)
[1] 0.9771296
```

4.3 Multiple Linear Regression

In real life, most interesting problems involve multiple predictors and one dependent variable, which calls for estimating a multiple linear model. That is:

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 \ldots + \beta_k x_k + e \tag{9}$$

If we make we make n independent observations, $y_1, y_2 \dots y_n$, on Y . We can write the observation y_i as

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} \dots + \beta_k x_{i1} + e_i \tag{10}$$

where x_{ij} is the j^{th} independent variable for the i^{th} observation, $i = 1, 2, \dots, n$. It easy to se that

$$\begin{aligned}
y_1 &= \beta_0 + \beta_1 x_{11} + \beta_2 x_{12} \dots + \beta_k x_{11} + e_1 \\
y_2 &= \beta_0 + \beta_1 x_{21} + \beta_2 x_{22} \dots + \beta_k x_{21} + e_2 \\
y_3 &= \beta_0 + \beta_1 x_{31} + \beta_2 x_{32} \dots + \beta_k x_{31} + e_2 \\
&\vdots \\
y_n &= \beta_0 + \beta_1 x_{n1} + \beta_2 x_{n2} \dots + \beta_k x_{n1} + e_n
\end{aligned}$$

The above system of equations can be written as

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & x_{12} & x_{1k} \\ 1 & x_{21} & x_{22} & x_{2k} \\ 1 & x_{31} & x_{32} & x_{3k} \\ \vdots & & & \\ 1 & 1x_{n1} & x_{n2} & x_{nk} \end{bmatrix} \times \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ \vdots \\ e_n \end{bmatrix} \tag{11}$$

The above system can be presented as

$$\mathbf{Y} = \mathbf{X}\beta + \mathbf{e} \tag{12}$$

where:

- $\mathbf{X}$ is called the design matrix.
- β is the vector of parameters
- $\mathbf{e}$ is the error vector
- $\mathbf{Y}$ is the response vector

Using the Least Squares Method, it can be shown that

$$\left(\mathbf{X}'\mathbf{X}\right)\hat{\beta} = \mathbf{X}'\mathbf{Y} \tag{13}$$

Thus,

$$\hat{\beta} = \left(\mathbf{X}'\mathbf{X}\right)\mathbf{X}'\mathbf{Y} \tag{14}$$

Equation 13 can be solved in R using a simple function given below

```
reg<-function(y, x){
x<-as.matrix(x)
x<-cbind(Intercept=1, x)
solve(t(x)%*%x)%*%t(x)%*%y
}
```

Furthermore we can have

```
str(heart_attack)
reg(y=heart_attack$CHARGES, x=heart_attack[, c(7, 8)])
```

4.4 Case Study

References