**NAME**

**COLLEGE NUMBER**

**Abstract**

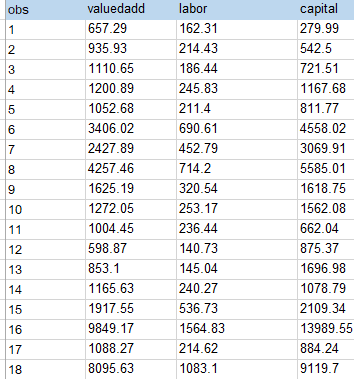
The application of data analyses systems span across nearly almost all industries today. Data science is the gold of the future, individuals and organisation are taking the advantages derived from the use and exploitation of the principle of the data science. Finanace, Healthcare and the education insdtries are currenly investing heavily as they tap into the full benefitis and capabilities of the dats scince industry. Engineers qwith a atechical bacjground have also gine ahead to develop software tools and programs that are focused on making the whole analysis process faster by developing algosthmic tools and software programs that automatically drive statistical inferences at the click of a button. Further, functions like sales and marketing, health sectos and the financial depwrtemnts are now employing the use of big data to help predict the next trend or point that their frecast might lland in. Examples here include future diseases discocveries and explorations, the next direction that the stock market might take and whether or not to invest on the particular shares and bonds, given past and current market prevailing factors. To achieve a given level of ecxpetced output and predictipm, the choice has to be made on the correct dataset, the model and the calassfication method to apply.

**Introduction**

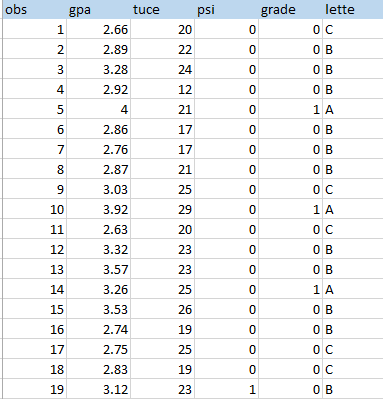
One final objective of the data engineer/data scineties is to finaly be abe to produce analysis thata speaks to anyone. The data analysis process should contain a clear scenario of where the objectibe of the snalayis is coming fomr, the problem its trying to solcve and the future of such analysis. Managers and top management consistently rely on the analysis of these data to make useful and meaningful derivations. One way to achieving this is by using R programmimg language wihich is a sttsiasticak tool as well as adata viauslalisation pack. The R studio environment gives the user the full potenatial to grab the analytical caoabilities of this tool. R as a programming lanugae on te hand has benne used for by induuvia =ls and coiorrations ovr the years ti dkiver businessneeds and meet statsistics analayis that area rmeaningful. Addiotiona,another siginificance of the language is that its higly supported, easy to learn and can be has a number of packages that suppor quick analysis and visualosations.

**Dataset description**

There are two datsets provided for this partuclur project, all the datasets are delivered in comm saeparated files. A quick sneep in to he the frist dtaastet tahts is a sprodcution datset revelas the following contents:



Whereas the snip into the second dataset reveals the following:



In order to begin the analysis, we shall load the to datstes unt o the Rstudio IDE, beginning with the first set:

##

> library(readr)

Warning message:

package ‘readr’ was built under R version 4.1.1

> production <- read\_csv("class/hello/papers/7th/Econometric decisions/production.csv")

Rows: 27 Columns: 4

-- Column specification --------------------------------------------------------------------------

Delimiter: ","

dbl (4): obs, valuedadd, labor, capital

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

i Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

> View(production)

> > str(production)

spec\_tbl\_df [27 x 4] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)

 $ obs      : num [1:27] 1 2 3 4 5 6 7 8 9 10 ...

 $ valuedadd: num [1:27] 657 936 1111 1201 1053 ...

 $ labor    : num [1:27] 162 214 186 246 211 ...

 $ capital  : num [1:27] 280 542 722 1168 812 ...

 - attr(\*, "spec")=

  .. cols(

  ..   obs = col\_double(),

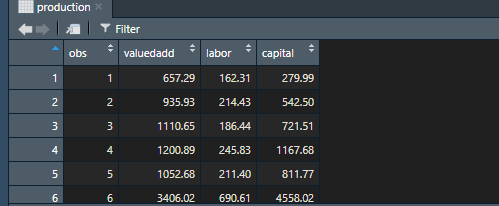
  ..   valuedadd = col\_double(),

  ..   labor = col\_double(),

  ..   capital = col\_double()

  .. )

 - attr(\*, "problems")=<externalptr>



1. Cobb-Douglas production function

The Cobb-Douglas production function usually is used in economics to evaluate and estimate the how production is affectedby othr factors of production such as laor, price and the demand.

library(micEcon)

> productionOutput <- production$valuedadd

> QuantityInputs <-production$capital

> productionTime <- c(1:20)

estResult <-  translogEst( "valueadd", c( "labor","capital", "productionTime"),

                           production, linear = TRUE )

In R, we can formulate the Cobb Douglas function by calling in the libarary Micecon that has a number of built in packges that support economic analysis. In this case the first step is to identify the variables that need to be used. We kniw very well the factors that would affct production and as such, the output in this case is the dependent variable, that change based on the independent variables of labor, time and capital. So we firt sinitalise the depdendt variable which in this stance is the $valueadd, then we initialise two more variables that is $captial and $labor, but in order to to get a smopth estimation, we need to do this over some period of time, so we add the last variables which is estResult that shall consume the translogEst built in micEcon package to give us the full estimate of these factors

Futher, we can test for the multicillinearity of this dataset using the corelanility matrix in R. For that, a built function called the corrplot() has a number of inbuolt fucntions tp help do this.

 ##DETERMINING MULTICOLINEARITY

 > library(corrplot)

corrplot 0.92 loaded

Warning message:

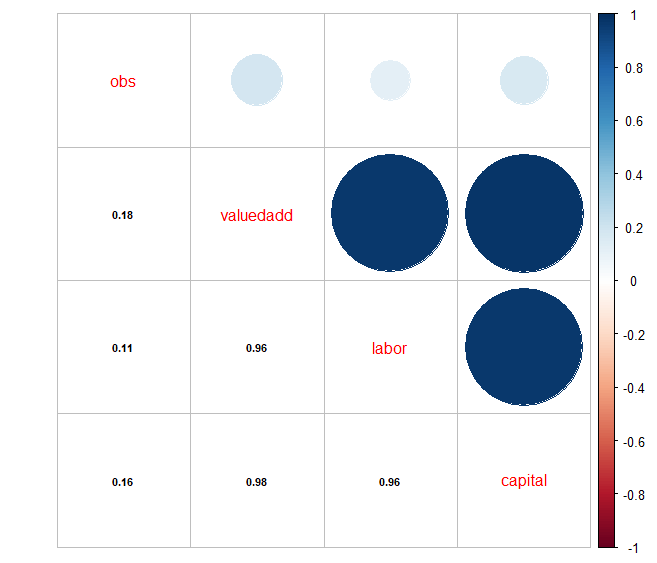
package ‘corrplot’ was built under R version 4.1.2

> cor1 = cor(production)

> corrplot.mixed(cor1, lower.col = “black”, number.cex = .7)

> corrplot.mixed(cor1, lower.col ="black", number.cex = .7)

>



The above illustatrtion shows that there is a high corraltion between the value addition and the labor inputs,also the multilinear relationhsips indicate a high realtiosnhip between the capital and the value addition of the business. The above cases can be illustrated to mean that output is directly prprotional to the labor and the capital invested.

**Model estimation and correlation**

##GETTING CORRELATION COFFICIENTS

> library(tidyverse)

> cor(production$valuedadd, production$capital)

[1] 0.9753576

> library(ggplot2)

> ggplot(production) +

+     aes(x = capital, y = valueadd) +

+     geom\_point(colour = "red") +

+     theme\_minimal()

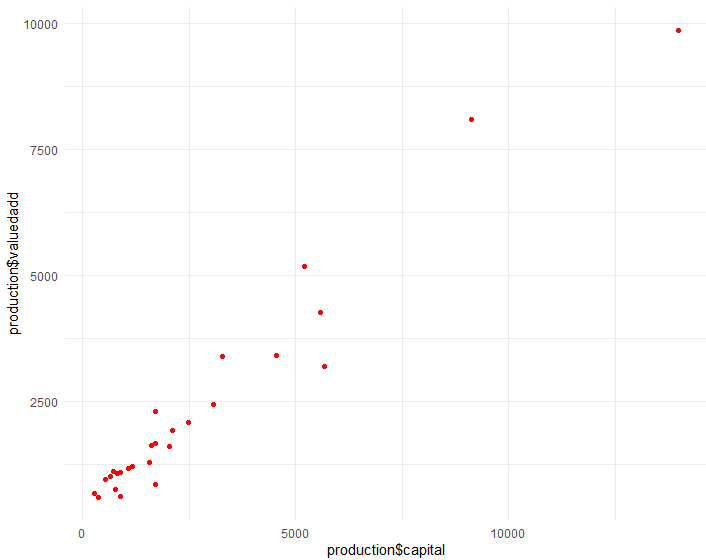
> ggplot(production) +

+     aes(x = production$capital, y = production$valuedadd) +

+     geom\_point(colour = "red") +

+     theme\_minimal()

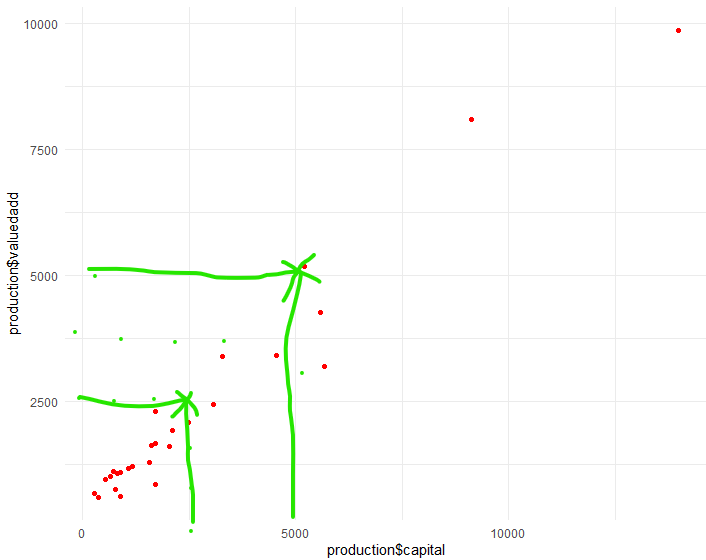
>

****

In the above we have done a coefecint corealtion of the value addition against the capital invested and thw coefficnet vakue returned is 0.9, this is very strong, showing that there is a big positive relationship between the two observed variables

**Constant returns to scale**

The theory of socntant retturns in scale indicates that as the input variables in the factors of production like labor and capital increase, so does the value addition, in paractica we can say that, if for instance we double the capital then the output will also double since these two are closedly related. In the case above, if we take the initial value of 2500 income at a value of 2500 and decide to double it, we shall get the value of retruns shall also double as per the below:



**Question two**

**Data inspection**

A quick observatiom of the data is dine by loading it on the R environment and inspecting its structueR

##GRADE DATA

> View(grade)

> str(grade)

spec\_tbl\_df [32 x 6] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)

 $ obs  : num [1:32] 1 2 3 4 5 6 7 8 9 10 ...

 $ gpa  : num [1:32] 2.66 2.89 3.28 2.92 4 2.86 2.76 2.87 3.03 3.92 ...

 $ tuce : num [1:32] 20 22 24 12 21 17 17 21 25 29 ...

 $ psi  : num [1:32] 0 0 0 0 0 0 0 0 0 0 ...

 $ grade: num [1:32] 0 0 0 0 1 0 0 0 0 1 ...

 $ lette: chr [1:32] "C" "B" "B" "B" ...

 - attr(\*, "spec")=

  .. cols(

  ..   obs = col\_double(),

  ..   gpa = col\_double(),

  ..   tuce = col\_double(),

  ..   psi = col\_double(),

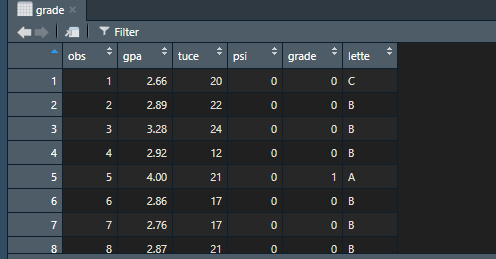
  ..   grade = col\_double(),

  ..   lette = col\_character()

  .. )

 - attr(\*, "problems")=<externalptr>

>



1. **Logistic regression model**

#CREATE TRAINIGN AND TEST DATA

> set1 <- grade$gpa

> set2 <- grade$tuce

> set.seed(1000)

> logitmodel <- glm(set1,set2)

#MODEL

> glm.fits=glm(psi∼ tuce + gpa + grade + lette, data=grade,family =binomial)

> summary(glm.fits)

