**Major Project Report on**

**Machine learning predictive analytic model to reduce cost of quality for software products**

**Submitted By:**

**Nitin Gupta**

**2K18/EMBA/524**

**Under the Guidance of:**

****

**DELHI SCHOOL OF MANAGEMENT**

**Delhi Technological University**

**Bawana Road Delhi 110042**

CERTIFICATE

This is to certify that the project entitled **“Machine learning predictive analytic model to reduce cost of quality for software products**” has been successfully completed by Nitin Gupta -2K18/EMBA/524

To the best of my knowledge this is assured that this project work is a record of bonafide work done by him under my guidance. The matter in the project is not submitted for award of any degree.

Nitin Gupta

**2K18/EMBA/524**

Mr Chandan Sharma

**Delhi School of Management**

ACKNOWLEDGEMENT

I wish to avail this opportunity and place on record my deep sense of gratitude and immense heartfelt thanks first and foremost to my company Synopsys team members for his valuable guidance, encouragement, meticulous suggestions and in valuable support for the successful completion of my “Research Project” in the prestigious organization.

My sincere thanks also goes to **Ms Afshan** for her generous help in project completion. She helped to successfully try different proof of concept.

Above all, last but not the least I feel highly indebted and extend my respectful thanks to **Mr Chandan Sharma** for his magnanimous support and patience for letting me complete my Project with ease.

Nitin Gupta

Executive Summary

In today’s world, high quality product are need of the time. The low-quality product results in the high cost. This can be explained from the quality graph below

1. Prevention cost can be define as the issue/bugs found out before the deployment/delivered to customer. This cost is initially very low but in the longer run goes up
2. Failure cost includes cost of losing customers, Root cause analysis and rectification. This cost is defiantly very huge

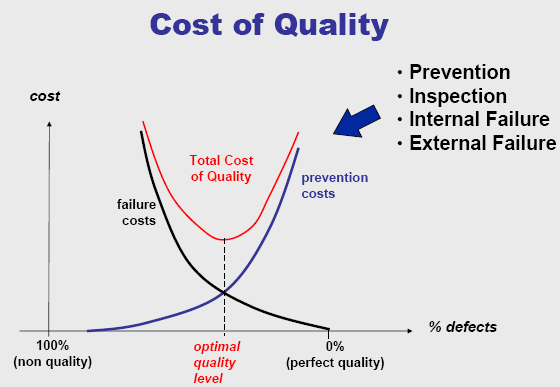
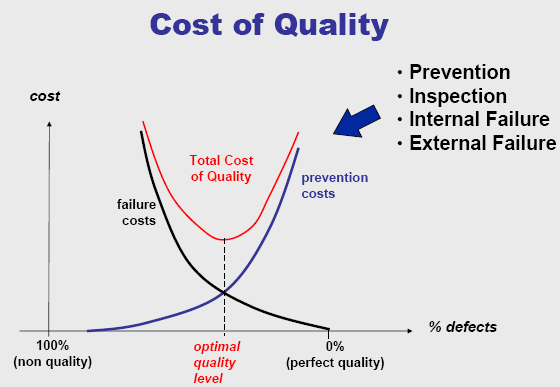
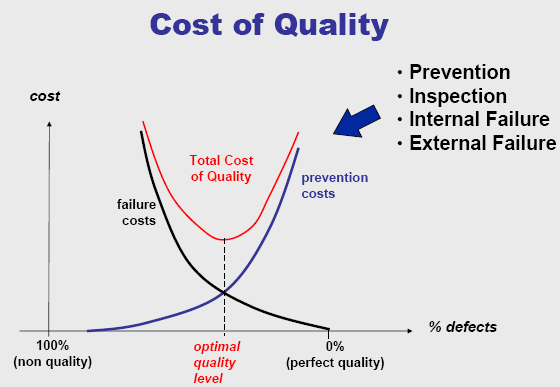
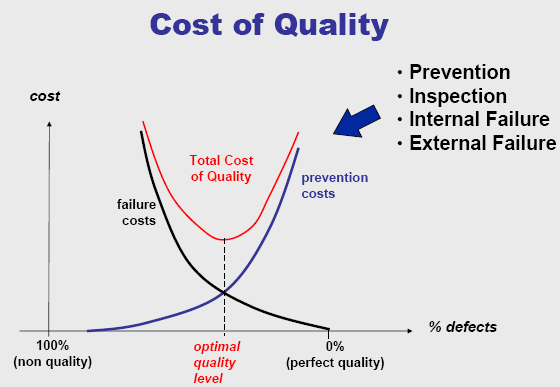


Figure 1‑1: Cost of Quality

Source: https://www.researchgate.net/

If there can be any mechanism that can help to identify the expected issues in the prevention cost then the overall all cost of quality can be reduce as shown in below graph



**Modified**

**TCQ**

**Modified**

**prevention cost**

Figure 1‑2: Modified Cost of Quality

Source: https://www.researchgate.net/

Electronic and Design Automation (EDA) Industry is backbone of Semiconductor Industry as it provide software tool aiding in the development of Semi-Conductors chips. EDA tools are from specification to the foundry input.

Below figure shows mapping of Chip design verification and currently available tools technologies

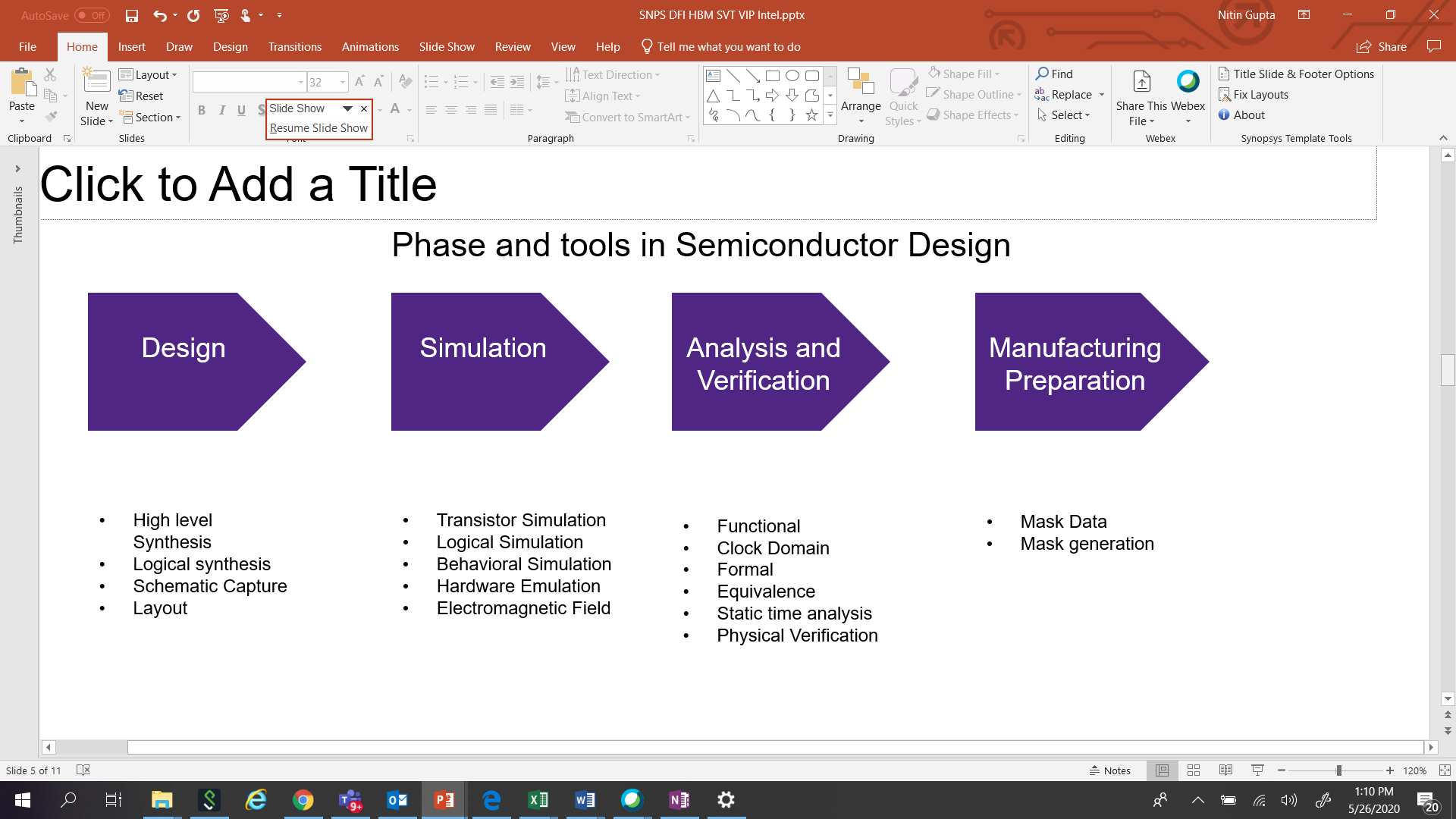


Figure 1‑3: Tools offered by EDA Industry

**Sourced: https://en.wikipedia.org/wiki/Electronic\_design\_automation**

Term tape out means the chip out of foundry and ready for use in electronic circuit. Re-spin means incident post Tape-out chips does not function as required and re-build is required. Cost of the tape out is minimum 5 million of dollars. Major re-spin reason is functionality issues, t**herefore function verification tools delivered by EDA needs to be always of high quality**.

A major problem faced by the Functional verification tool R&D team is to predict the numbers of the bugs that might have been introduced during the design phase to sign off the completeness and quality. If these bugs can be predicted, then the COQ can be reduced. Hence saving million of dollar to company and customer.

Machine learning, a upcoming new discipline, define scientific study of algorithm and using computing power develop prediction model so that certainty of the task can be managed.

In this project, prediction model for expected bugs during the development of the software is designed to help the Product manager to get confidence on quality.

For the data, explanatory research and Interview was conducted with-in the Synopsys.

This project has been successfully adopted with-in the Verification IP group of EDA leader and is in process to get it implemented in all different Business Units.

**Contents**

[CERTIFICATE ii](#_Toc42083556)

[ACKNOWLEDGEMENT iii](#_Toc42083557)

[Executive Summary iv](#_Toc42083558)

[1 INTRODUCTION: EDA INDUSTRY 3](#_Toc42083559)

[1.1 Overview 3](#_Toc42083560)

[1.2 Back Ground 3](#_Toc42083561)

[1.3 EDA Market Share company wise 4](#_Toc42083562)

[1.4 Future Growth 5](#_Toc42083563)

[2 Machine learning Overview 6](#_Toc42083564)

[2.1 Introduction 6](#_Toc42083565)

[2.2 Machine learning methods 6](#_Toc42083566)

[2.2.1 Supervised machine learning algorithms 6](#_Toc42083567)

[2.2.2 Unsupervised machine learning algorithms 8](#_Toc42083568)

[2.2.3 Semi-supervised machine learning algorithms 8](#_Toc42083569)

[2.2.4 Reinforcement machine learning algorithms 8](#_Toc42083570)

[3 Cost of Quality (COQ) 11](#_Toc42083571)

[4 Objective: 13](#_Toc42083572)

[5 Literature Review & Model Development 15](#_Toc42083573)

[5.1 Data collection 15](#_Toc42083574)

[5.1.1 Expert Interview 15](#_Toc42083575)

[5.1.2 Identification of parameter for data 15](#_Toc42083576)

[5.1.3 Secondary Data Exploration 16](#_Toc42083577)

[5.2 Data Preparation & Exploration 16](#_Toc42083578)

[5.2.1 Variable identification 16](#_Toc42083579)

[5.2.2 Univariate Analysis 17](#_Toc42083580)

[5.2.3 Bi-variate Analysis 18](#_Toc42083581)

[5.2.4 Missing values treatment 19](#_Toc42083582)

[5.2.5 Outlier treatment 19](#_Toc42083583)

[5.2.6 Variable creation 20](#_Toc42083584)

[5.3 Regression Model Selection 22](#_Toc42083585)

[5.3.1 Multiple Linear regression 22](#_Toc42083586)

[5.3.2 Assumption in Linear Regression Model 22](#_Toc42083587)

[5.3.3 Regression development model approach 23](#_Toc42083588)

[5.4 Regression Model Implementation 24](#_Toc42083589)

[6 Finding and Observations 25](#_Toc42083590)

[6.1 Correlation 25](#_Toc42083591)

[6.2 Data observation 25](#_Toc42083592)

[6.2.1 Comparison of VIP Type 25](#_Toc42083593)

[6.3 Regression 29](#_Toc42083594)

[6.4 Hypothesis 30](#_Toc42083595)

[6.5 Conclusion 30](#_Toc42083596)

[7 Refences 31](#_Toc42083597)

[8 Appendix 32](#_Toc42083598)

[8.1 R Code 32](#_Toc42083599)

[8.2 Data Set 34](#_Toc42083600)

[8.3 One hot encoding for VIP type 36](#_Toc42083601)

YCUS INFOTECH PVT. LTD.

**Figures**

INFOTECH PVT. LTD.

[Figure 1‑1: Cost of Quality iv](#_Toc42083602)

[Figure 1‑2: Modified Cost of Quality v](#_Toc42083603)

[Figure 1‑3: Tools offered by EDA Industry vi](#_Toc42083604)

[Figure 1‑1: Market Share 4](#_Toc42083605)

[Figure 1‑2: Year Wise Growth 5](#_Toc42083606)

[Figure 2‑1: Training and Test Data division 7](#_Toc42083607)

[Figure 2‑2: AI vs ML vs DL 9](#_Toc42083608)

[Figure 2‑3: Supervised vs Un Supervised learning 10](#_Toc42083609)

[Figure 3‑1: Detail on COQ 13](#_Toc42083610)

[Figure 4‑1: Phases of Software development 14](#_Toc42083611)

[Figure 5‑1: Data type 17](#_Toc42083612)

[Figure 5‑2: Method for Univariate Analysis 17](#_Toc42083613)

[Figure 5‑3: Check for Outliner 20](#_Toc42083614)

[Figure 5‑4: One hot encoding in case Male and Female 21](#_Toc42083615)

[Figure 5‑5: Linear regression equation 22](#_Toc42083616)

[Figure 5.4‑1: Model Flow Diagram 24](#_Toc42083617)

[**Figure 6‑1: Comparison between VIP type and Bugs per line** 25](#_Toc42083618)

[**Figure 6‑2: Comparison between VIP type and Average Engineer productivity** 26](#_Toc42083619)

[**Figure 6‑3: Comparison between VIP type and Average Engineer experience** 26](#_Toc42083620)

[**Figure 6‑4: Comparison between Avg Bus Vs Unit level verification** 27](#_Toc42083621)

[**Figure 6‑5: Comparison between Avg Bus Vs Reference Code** 28](#_Toc42083622)

[**Figure 6‑6: Comparison between Avg Bus Vs Code Review** 28](#_Toc42083623)

# INTRODUCTION: EDA INDUSTRY

## Overview

**“Electronic Design Automation” (EDA) Market**size valued approximate $9.76 billion in year of 2018 and is expected to grow at a CAGR of 10.1% from year 2019 to 2024-25. It is projected that in year 2024-25 the size will be about $18 billion. Thus, it is one of growing sector.

During the Covid-19 pandemic, this EDA industry is least effected industry with major player keeping the earning as pre Covid-19.

## Back Ground

The EDA players provides software packages. They also provide services like

* Licensed software
* Maintenance services
* Training
* Troubleshooting

EDA has been a very R&D investing domain. EDA toold are used by Electronics and semiconductor industries.

Company in EDA:

* US Based Synopsys, Inc.
* Siemens, Mentor Graphics
* US based Cadence Design Systems

The industry requires a highly skilled talent pool as it is highly labor-intensive. EDA

* Free usage
* Paid License

EDA industry have are adopting several strategies like

* Collaborations
* Partnerships
* Agreements
* Mergers & Acquisitions (M&A).

## EDA Market Share company wise

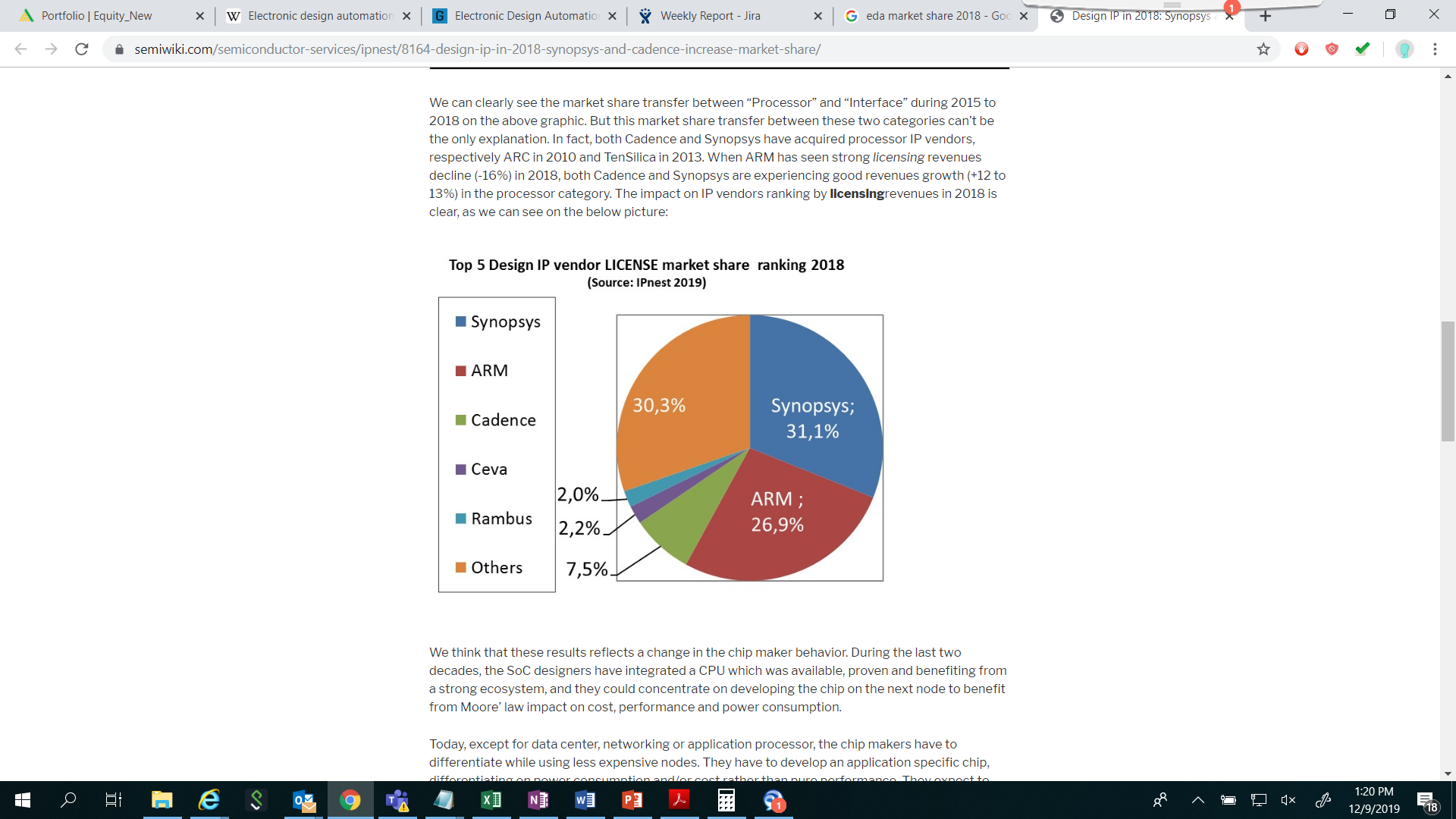


Figure 1‑1: Market Share

Sourced: <https://www.gminsights.com>

## Future Growth

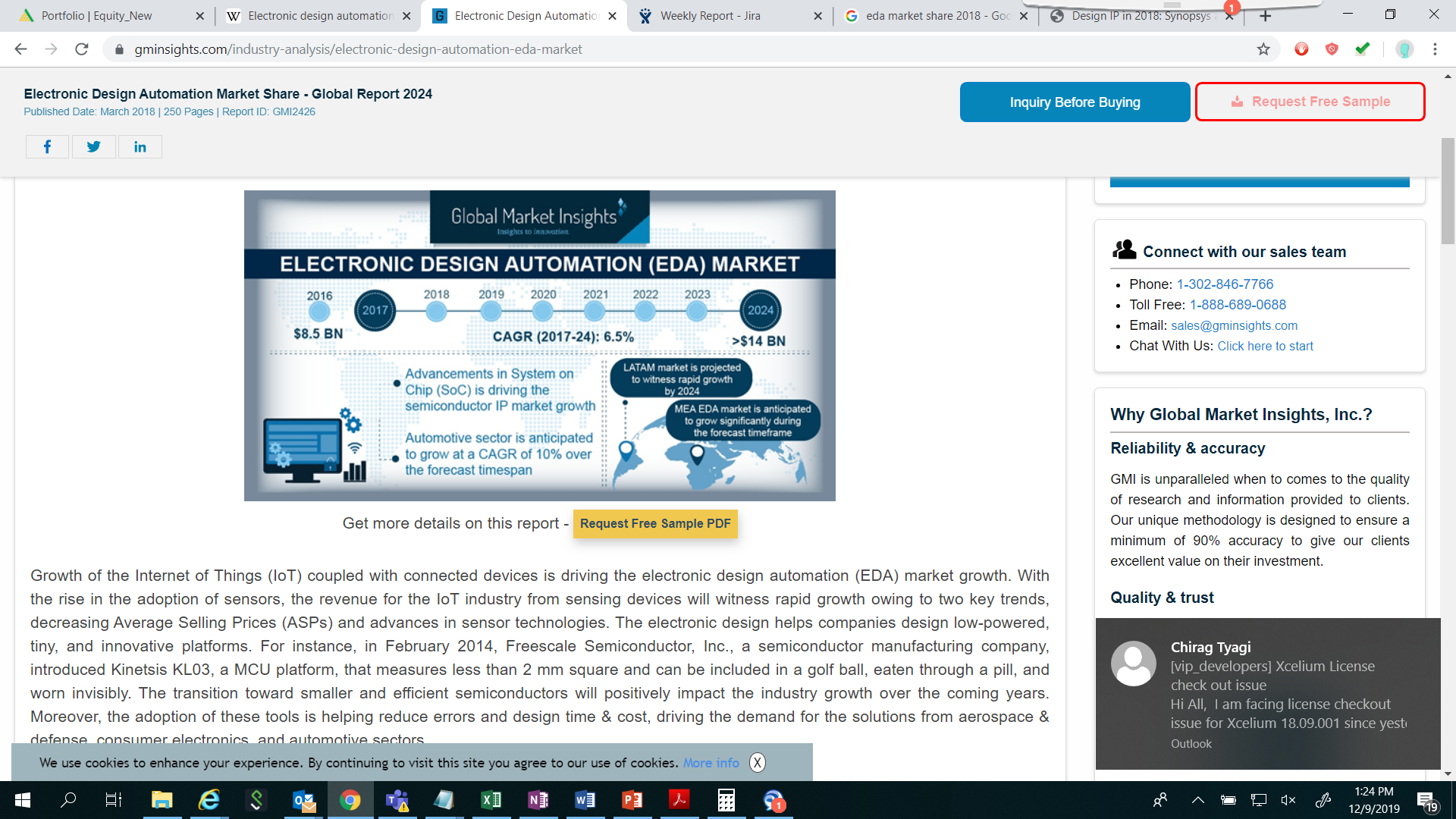
[](https://kandra.pro/)

Figure 1‑2: Year Wise Growth

Sourced: <https://www.gminsights.com/>

# Machine learning Overview

## Introduction

Traditionally all the software technologies based on instruction set. That means software doing action based on the instructions. These have been be basic of all the project management tools and mechanism designed.

Artificial Intelligence(AI) has Machine learning(ML) as an application that is to develop systems that learn and improve from experience itself. These are not software programs that are instructed to execute a particular job/function.

**Data input is mail activity and learning from data is the Machine learning.**

**The only motive of ML is to allow the application to learn automatically/itself** without

* Human intervention
* Human assistance

It always adjusts itself with the new data that is being fed.

## Machine learning

Machine learning algorithms are divided into two:

* Supervised
* Unsupervised.

### **Algorithms  for Supervised learning**

Input data must be labeled data that means that the data should be tagged. Supervised machine learning algorithms are applied where predictive analysis is needed based on the past data. The ML model to learn itself from the data captured in past and then predict the data. This is future data.

For Supervised ML model’s implementation, past data is divided into two sets

* Training dataset: Model is function on the training data set. This means that the model learns from this data
* Test dataset: Model generated is test with this data set for the correctness. This means the model correctness is checked from this data

Generally, the past data is divided into the training and test data in ration of 80:20

Test Data

Training Data

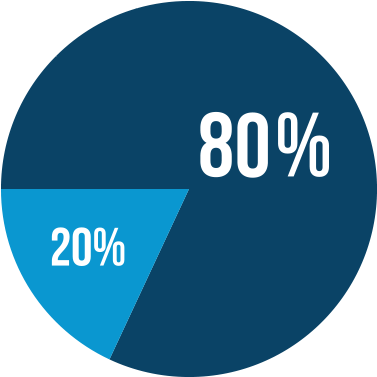


Figure 2‑1: Training and Test Data division

### **Unsupervised machine learning algorithms**

It can be said that getting information from data. Clustering and Association are two function that are used in un supervised.

### **Semi-supervised machine learning algorithms**

“Fall somewhere in between supervised and unsupervised learning, since they use both labeled and unlabeled data for training – typically a small amount of labeled data and a large amount of unlabeled data. The systems that use this method are able to considerably improve learning accuracy. Usually, semi-supervised learning is chosen when the acquired labeled data requires skilled and relevant resources in order to train it / learn from it. Otherwise, acquiring unlabeled data generally doesn’t require additional resources.”

### **Reinforcement machine learning algorithms**

Uses the concept of interaction with environment and produce action and for action it either gets rewards or punishment. Characteristics are

* Trial
* Error search
* Rewards

Based on the above, model generates the best action under the environment conditions. “This method allows machines and software agents to automatically determine the ideal behavior within a specific context in order to maximize its performance. Simple reward feedback is required for the agent to learn which action is best; this is known as the reinforcement signal.”

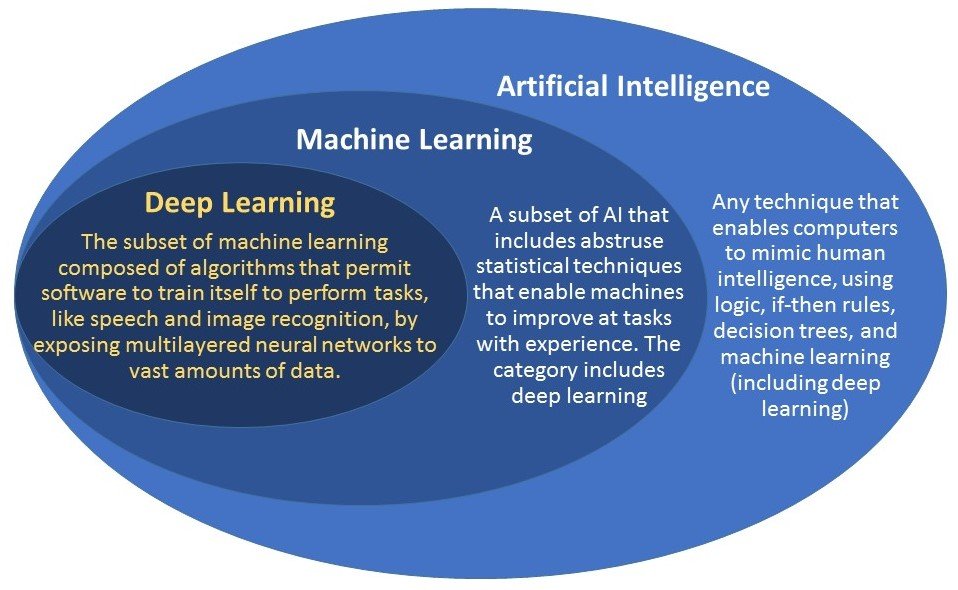


Figure 2‑2: AI vs ML vs DL

Sourced: <https://hcs-pharma.com>

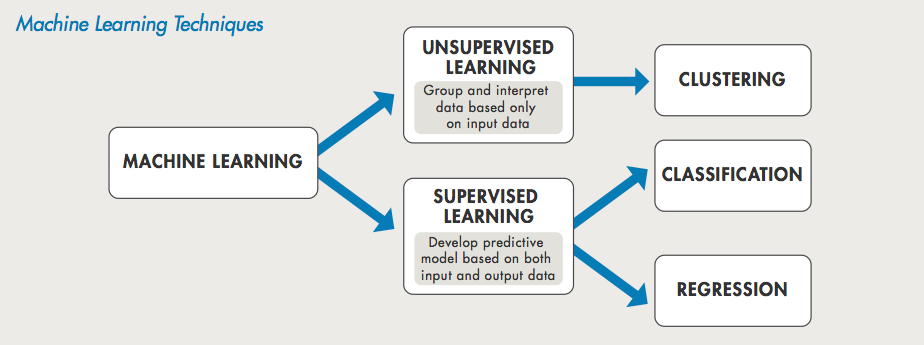


Figure 2‑3: Supervised vs Un Supervised learning

Sourced: <https://www.mathworks.com>

# Cost of Quality (COQ)

“Quality costs are the [costs](https://www.accountingtools.com/articles/2017/5/4/cost) associated with preventing, detecting, and remediating product issues related to quality. Quality costs *do not* involve simply upgrading the perceived value of a product to a higher standard. Instead, quality involves creating and delivering a product that meets the expectations of a [customer](https://www.accountingtools.com/articles/2017/5/4/customer). Thus, if a customer spends very little for an automobile, he will not expect leather seats and air conditioning - but he will expect the vehicle to run properly. In this case, quality is considered to be a vehicle that functions, rather than a luxury experience.”

COQ can be divided into following categories:

* [*Prevention costs*](https://www.accountingtools.com/articles/2018/1/27/preventive-costs). Prevention cost is the cost in order to keep a quality problem from occurring. It’s characteristic are :
  + Most least expensive type of cost,
  + Highly recommended.

It can be achieved by following:

* + Proper [employee](https://www.accountingtools.com/articles/2017/5/4/definition-of-an-employee) training in assembling products
  + Statistical process control
  + Robust product design
  + [Supplier](https://www.accountingtools.com/articles/2017/5/16/supplier) certification.
* [*Appraisal costs*](https://www.accountingtools.com/articles/what-are-appraisal-costs.html). Incurred in order to keep a quality problem from occurring. This is done through a variety of inspections. The least expensive by having
  + Code review
  + Specification Review
  + Testing
* [*Internal failure costs*](https://www.accountingtools.com/articles/2017/5/10/internal-failure-costs). Incurred when a defective product is produced results in scrap or re work. The cost of revise goods is part of this cost.
* [*External failure costs*](https://www.accountingtools.com/articles/2017/5/6/external-failure-costs). When faulaty product is delivered. Cost is high as it includes
  + Recall Cost
  + [Warranty](https://www.accountingtools.com/articles/2017/5/11/warranty) claims
  + Field service
  + Legal.
  + Losing customer

Quality costs can arise because of the following.

* Product design issues
* Coding issues.
* Incorrectly customer requirements.

In the software Industry, Quality costs is one of the major expense in the total expenses of a business. The vast part is that it is hidden and counting/recoding is again a new challenge that is faced. There is no process define that how can these hidden.

To increase profitability, it can be a good approach to manage the COQ. By reducing these over-all profitability by reducing the cost.

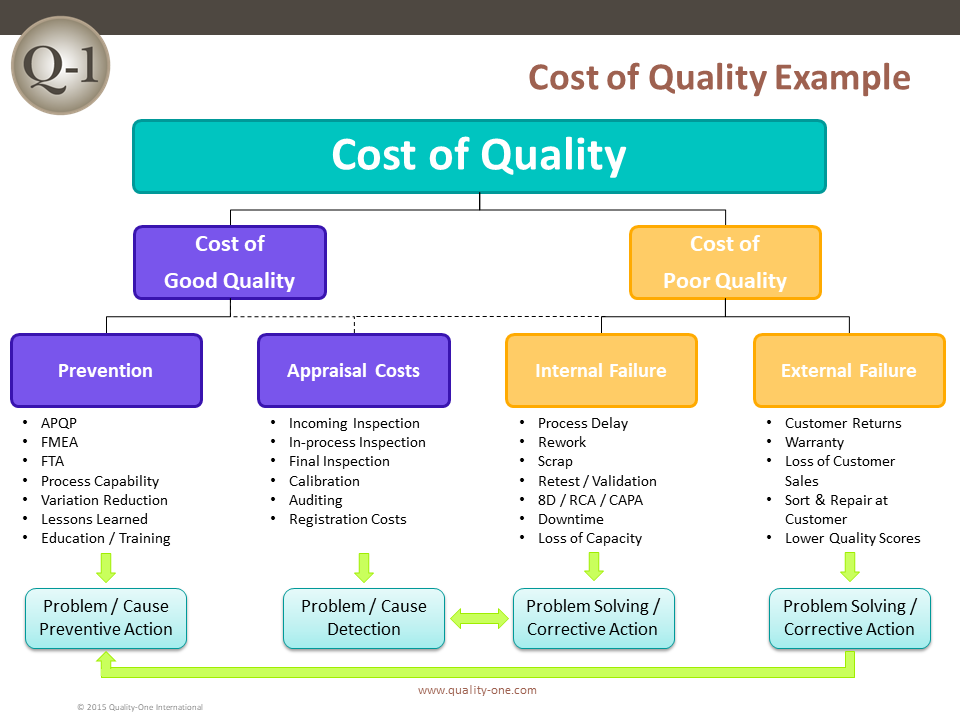


Figure 3‑1: Detail on COQ

Sourced: https://quality-one.com/coq/

# Objective:

As stated in the introduction, Bugs in the EDA industry cost huge to the customers. Therefore, it is utmost important to ensure high quality products. There are large number of the mechanics like 6 Sigma, ISO, CMM level that helps in removing the error but still the human factor is not observed in these. Following can be consider has human factors

* Complexity of the Implementation
* Rating of the engineer
* Engineer experience
* Type of protocol

The project life cycle can be detailed as

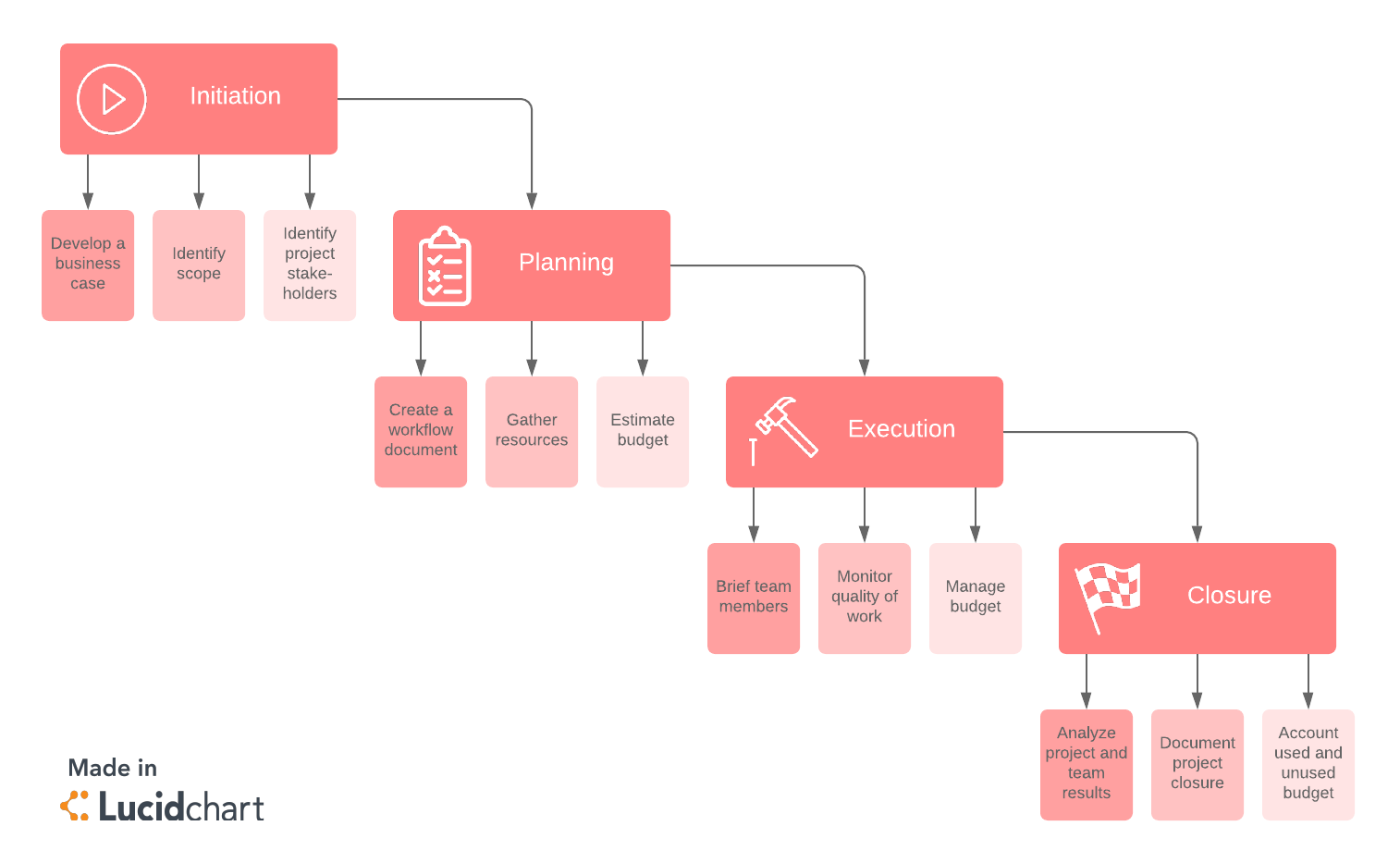


Figure 4‑1: Phases of Software development

Now during the execution phase, if the Manager has tool that helps him to predict the number of bugs expected then it will help him to monitor the project. In case there are less bugs identified then it can be either hole in the verification or design problems.

As it said that bugs are coded therefore bugs in the design can be related to the past bugs that have been found in the similar project/module.

Therefore, model was design to predict the expected bugs in a module so that manager can figure out if the verification is complete or not.

. LTD.

# Literature Review & Model Development

## Data collection

Data collection for the model generation was a very tedious task, there was no location where the data for the model was required. Therefore two data collection process was brought. The three data collection process were

* Expert Interview
* Identification of parameters to consider for data
* Secondary data Exploration

### Expert Interview

The first challenge for the selection of the data was to identify the parameters that should be applied to the model. For this there was no data available or any study available for it. Therefore it was decided that expert interview to be done with the product manager and also the field engineers and get information on the parameter that effect the graceful coding.

In total 10 expert interview were done. The questionnaire asked

* Does experience of the engineer given task matter into the quality productivity?
* Type of the protocol category creates complexity?
  + Serial
  + Bus Interface
  + Memory
* Expected productivity form the engineer matters in quality?
* Can the quality be related to past experience in similar module?
* Does effective code review reduces the bugs?

### Identification of parameter for data

After conducting the Interviews, the following parameters were decided for the regression model.

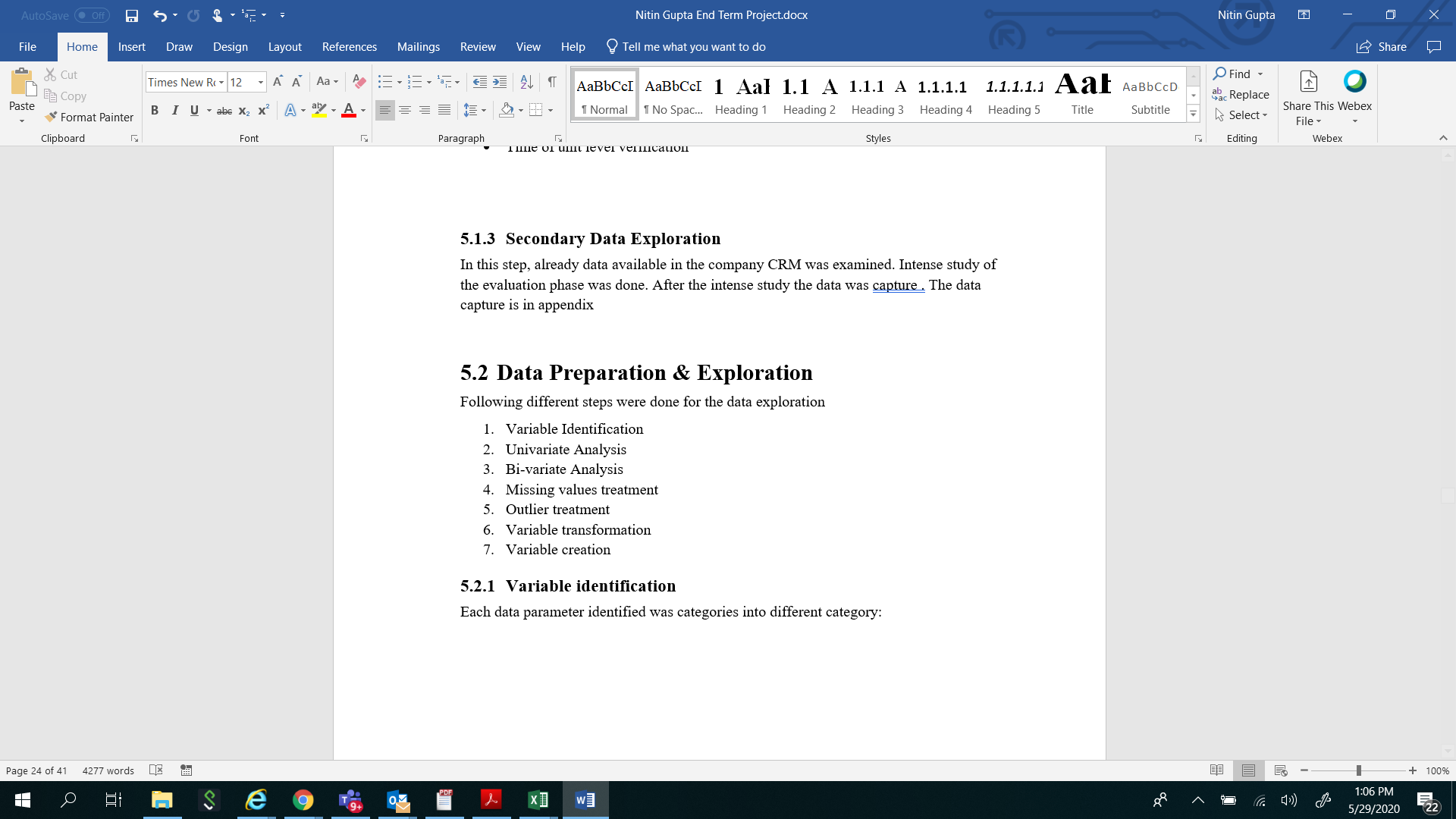
* Module relation with type of protocol
  + Bus
  + Serial
  + Memory
* Total Lines of code generated
* Total experience of engineer
* Engineer rating
* Complexity of Protocol
* Productivity of engineer expected
* Time for Prototype Module
* Reference module accessible to engineer
* Time of unit level verification

### Secondary Data Exploration

In this step, already data available in the company CRM was examined. Intense study of the evaluation phase was done. After the intense study the data was capture . The data capture is in appendix

## Preparation & Exploration: Data

Following different steps were done for the data exploration



### Variable identification

Each data parameter identified was categories into different category:

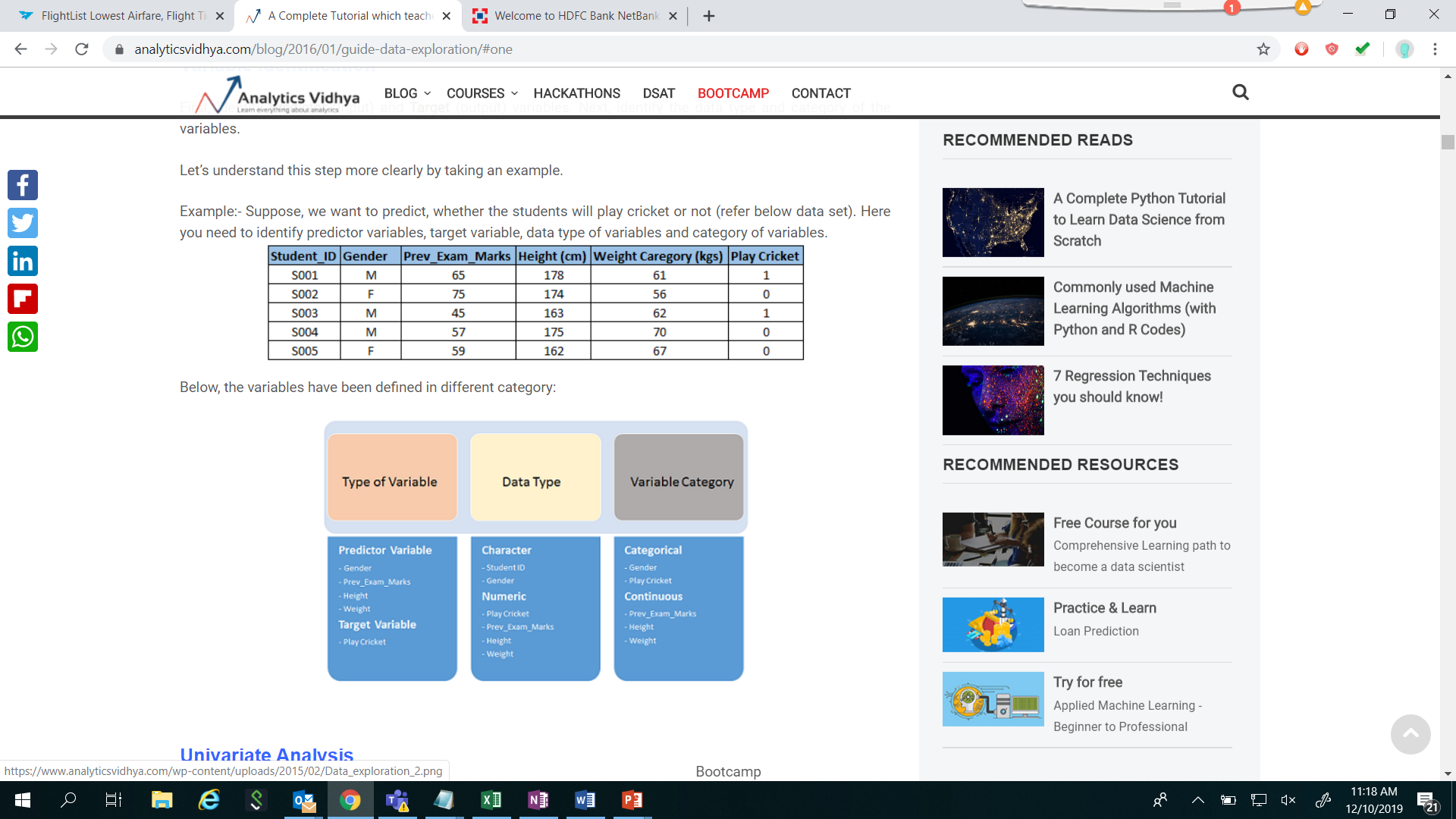


Figure 5‑1: Data type

Source: <https://www.analyticsvidhya.com/>

The identification of data was done as

* **Character, Categorical**
  + VIP Type
  + Reference Code available
  + Efficient Review conducted
  + Claimed Unit level verification Done
* **Numeric, Continuous**
  + Engineer experience
  + Engineer experience in VIP Type
  + Productivity expected
  + Number of lines coded
  + Duration of Development in man weeks
  + Bugs Found

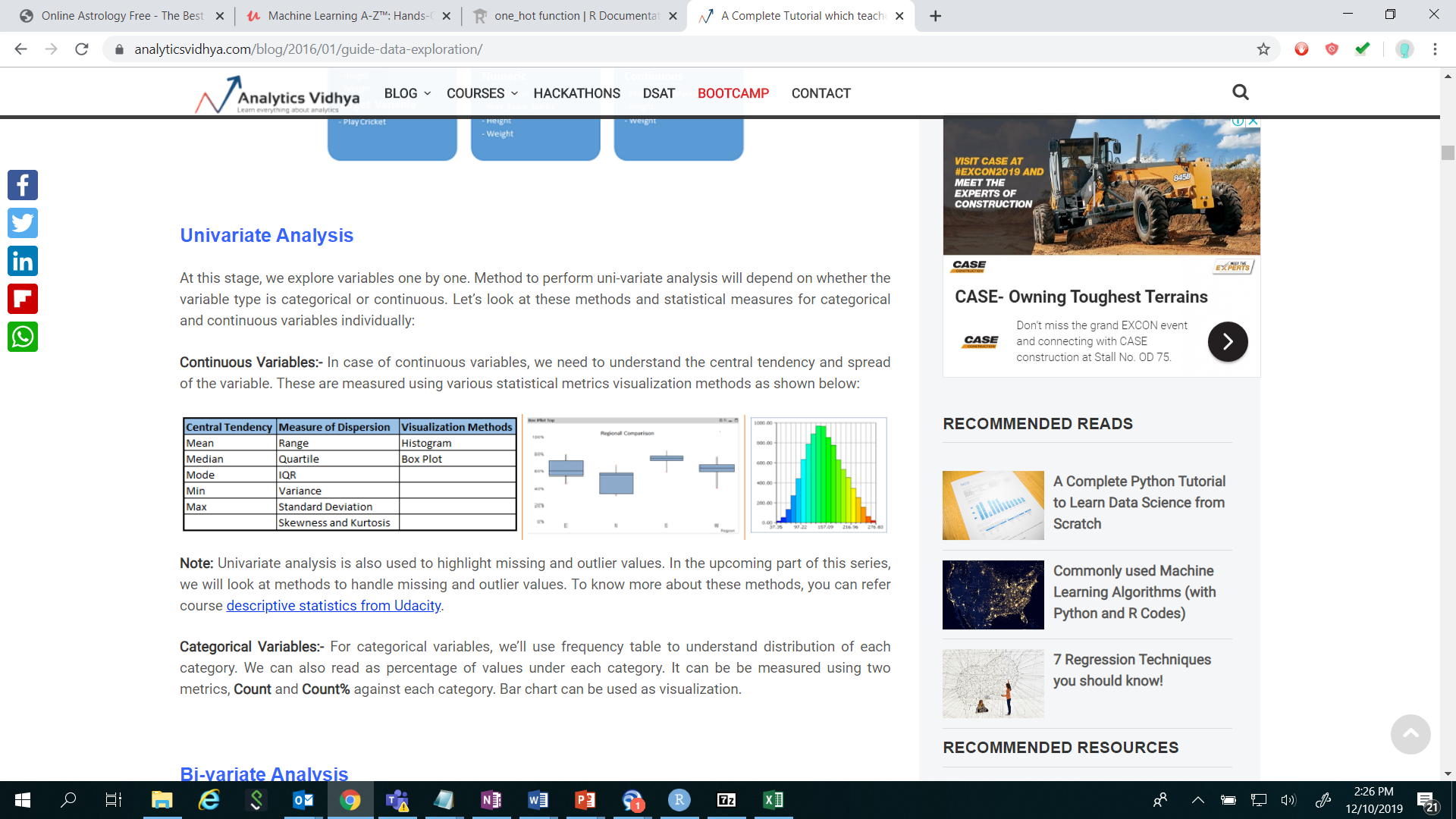
### Univariate Analysis

Each variable was explore one by one. Method used

* For Continuous data the central frequency for continuous
* For non-continuous frequency table use

Figure 5‑2: Method for Univariate Analysis

Source: <https://www.analyticsvidhya.com/>



### Bi-variate Analysis

Define as Establish relationship between two variables. Here, we check for association and disassociation between variables at a significance level with is 95%. Bi-variate can be:

* Categorical & Categorical
* Categorical & Continuous
* Continuous & Continuous.

#### Continuous & Continuous:

“Scatter plot is used for doing bi-variate analysis between two continuous variables. It is a best way to figure out relationship between two variables. The pattern of scatter plot indicates the relationship between variables that relationship can either be linear or non-linear.”

Scatter plot: relationship between two variables but without information on relationship value.

Correlation values are between minus one and positive one.

* -1: linear correlation and negative
* +1: linear correlation and positive
* 0: No Correlation does exist

#### Categorical/Categorical:

To find the relationship between two categorical variables, we can use following methods:

* **Two-way table:** Analyzing the relationship by creating a two-way table of count and count%.
  + Rows represents one variable and the
  + Columns represent the other variable.

It count or count% of observations available in each combination of row and column categories.

* **Stacked Column Chart: For** Two-way table for visual inspection.
* **Chi-Square Test:** This test derives the statistical significance of relationship between the variables. Also helps to find if sample size can be use to represent population. “Chi-square is based on the difference between the expected and observed frequencies in one or more categories in the two-way table”. It returns
  + Probability for the computed chi-square distribution with the degree of freedom.

#### Categorical & Continuous:

Z-test and ANOVA test are performed look at the statistical significance,

* **Z-Test/ T-Test:-** Either test assess whether mean of two groups are statistically
* **ANOVA:-**To assesses if average of more than two groups is statistically different.

### Missing values treatment

Three type of the treatments are done

* Average
* Remove the data
* Random Value

Recommended is Average value

### Outlier treatment

There are two type of Outlier

* **Univariate**
* **Multivariate**.

Outlier can be defined as value of a data which is far away from the mean value or can be define as value which is out of range.

An example:

“Refer scatter plot. Here, we have two values below and one above the average in a specific segment of weight and height.”

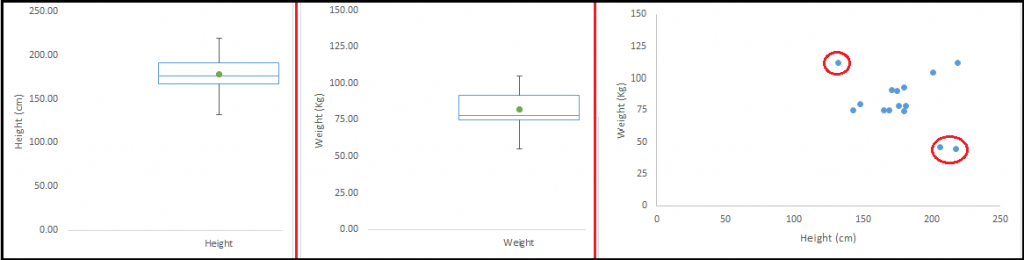
[](https://www.analyticsvidhya.com/wp-content/uploads/2015/02/Outlier_21.png)

Figure 5‑3: Check for Outliner

Source: <https://www.analyticsvidhya.com/>

### Variable creation

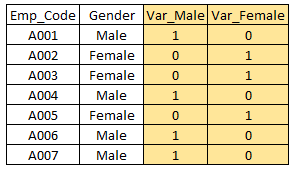
This is technique to convert the Categorical data into the numerical data. The techniqe used is one hot encoding

**Creating dummy variables:**Dummy variable also called Indicator Variables are converts from categorical variable into numerical variables. Categorical variable are required as predictor in statistical models.  0 and 1 are values that Categorical variable can take.

For Example: Two City

We can produce two variables, namely, “**Var\_Delhi**” with values 1 (Delhi) and 0 (No Delhi and “**Var\_Agra**” with values 1 (Agra) and 0 (Delhi). For n values n-1 dummy variables are created.

Figure 5‑4: One hot encoding in case Male and Female

[](https://www.analyticsvidhya.com/wp-content/uploads/2015/03/Dummy.png)

Source: <https://www.analyticsvidhya.com/>

## Regression Model Selection

The data capture is label data and has numerical output therefore Multiple Linear regression will fit the best

### Multiple Linear regression

The below figure represents the Linear regression.

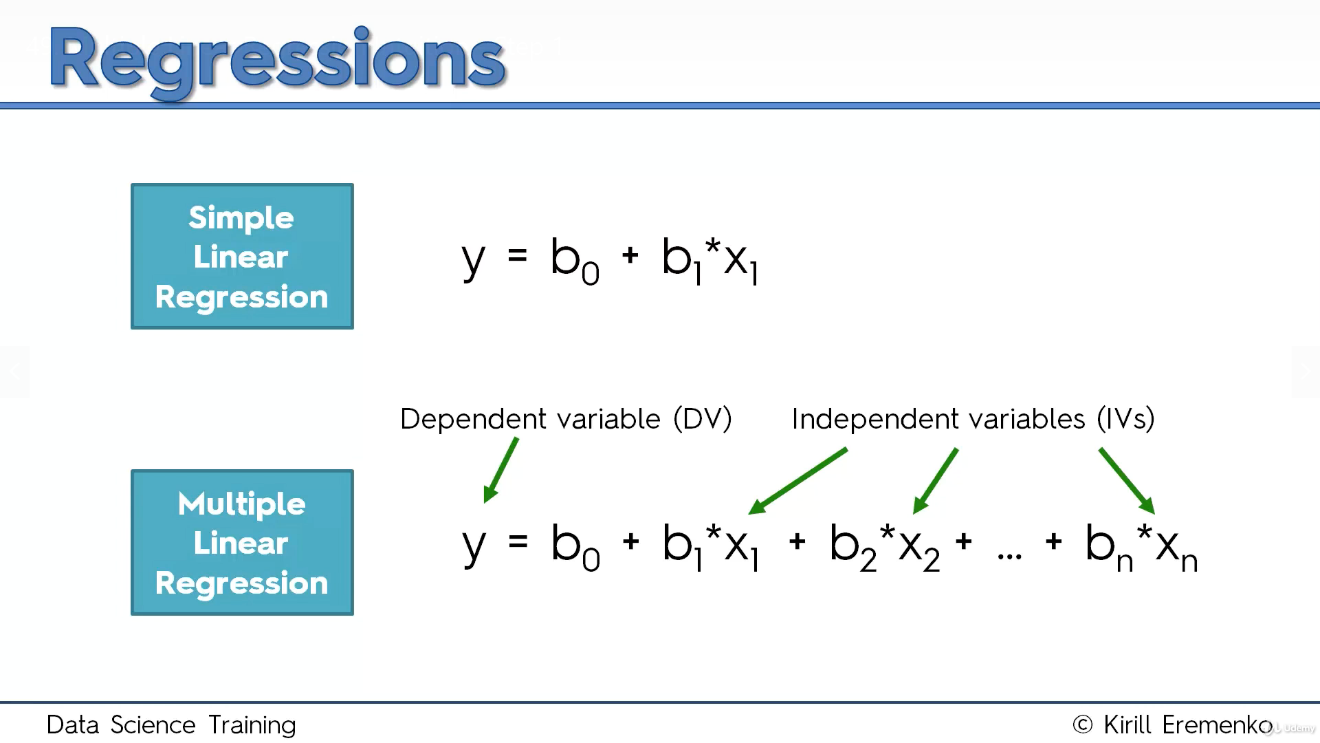


Figure 5‑5: Linear regression equation

Source: <https://www.udemy.com/>

### Assumption in Linear Regression Model

* Linearity
* Homoscedasticity
* Multivariate Normality
* Independence of errors
* Lack if Multicollinearity

### Regression development model approach

There are 5 different regression development model approaches

1. Alli-in
2. Backward Elimination
3. Forward Elimination
4. Bidirectional Elimination
5. Score Comparison

For the project, Backward elimination approach is selected.

## Regression Model Implementation

For the regression model implementation, the below flow chart was done

Data exploration and preparation

Implementation of Regression mode in R

Remove Variable

Check for Z value more than 95%

Test with Test Data

Figure 5.4‑1: Model Flow Diagram

# Finding and Observations

## Correlation

As stated earlier, correlation is to find relation ship between the variables. For the project correlation ship was observed numerical data For the correlation MS excel was use. The correlation value can be between the

|  |  |
| --- | --- |
| **Correlation** | |
| Engineer experience | 0.145955305 |
| Engineer experience in VIP Type | 0.125225741 |
| Productivity expected | 0.604146301 |
| Number of lines coded | 0.276170642 |

**Observation: All the coefficient variable are greater than 0 therefore the are positively related to the output that is bugs.**

## Data observation

### Comparison of VIP Type

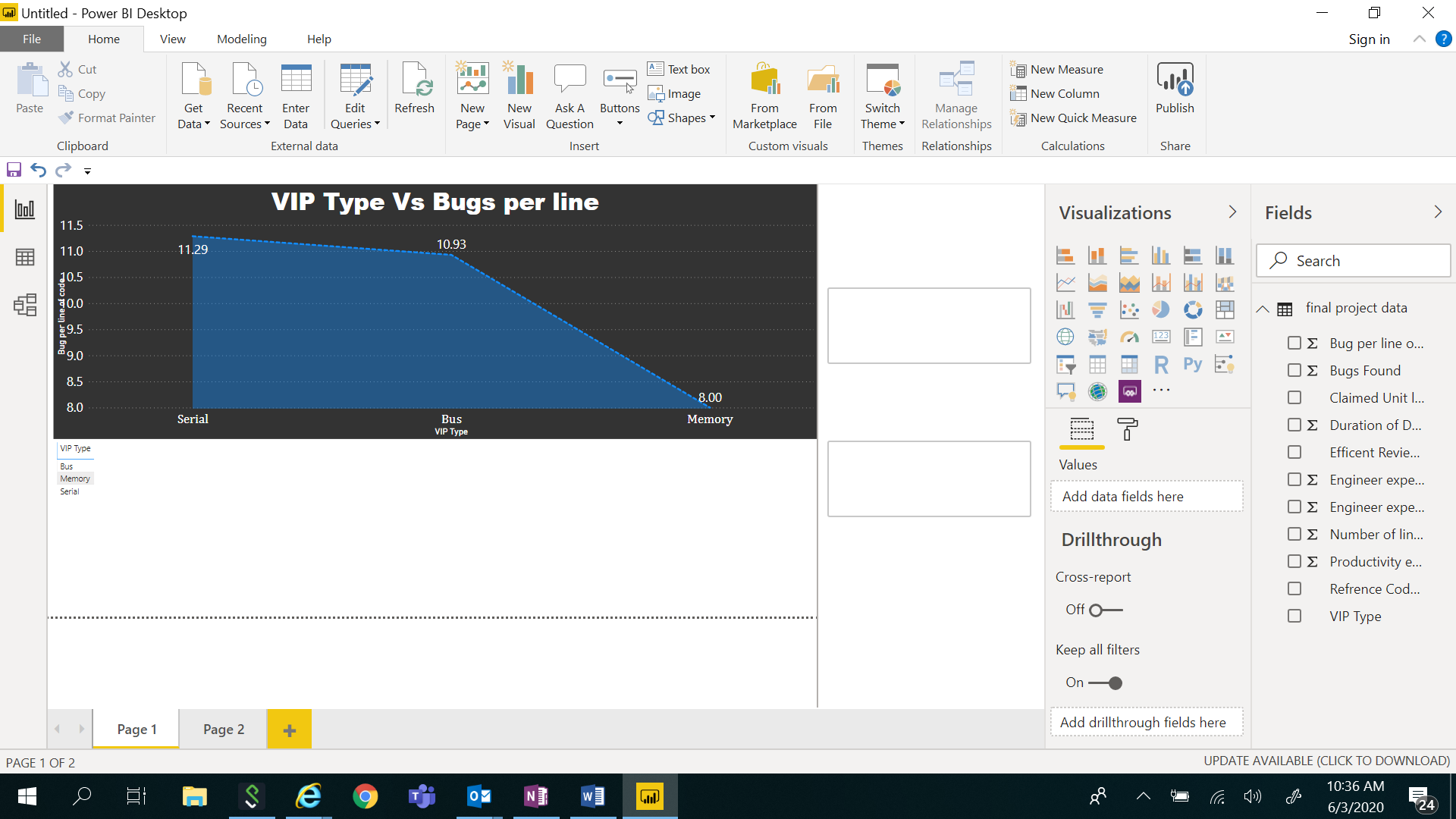


Figure 6‑1: Comparison between VIP type and Bugs per line

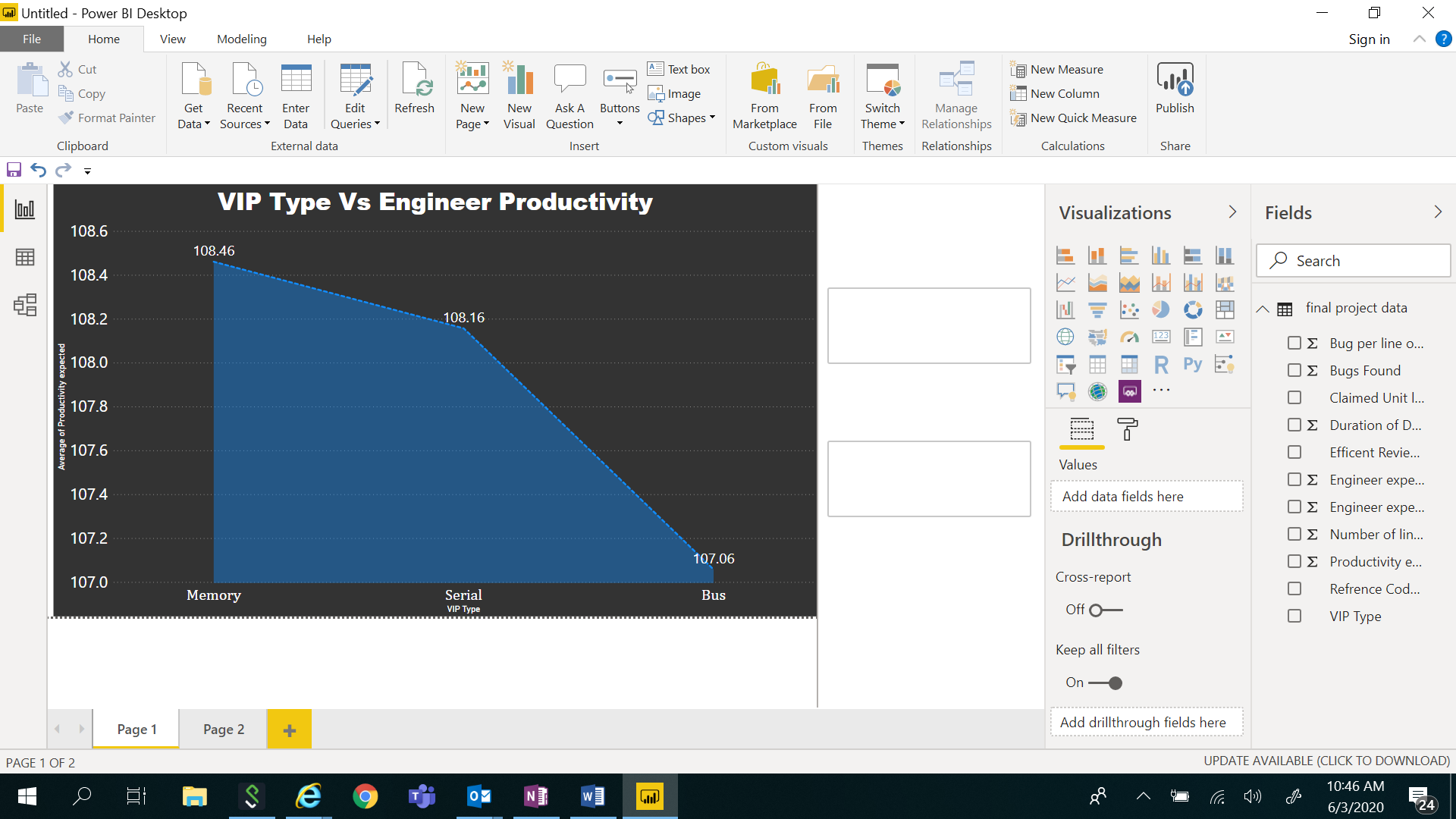


Figure 6‑2: Comparison between VIP type and Average Engineer productivity

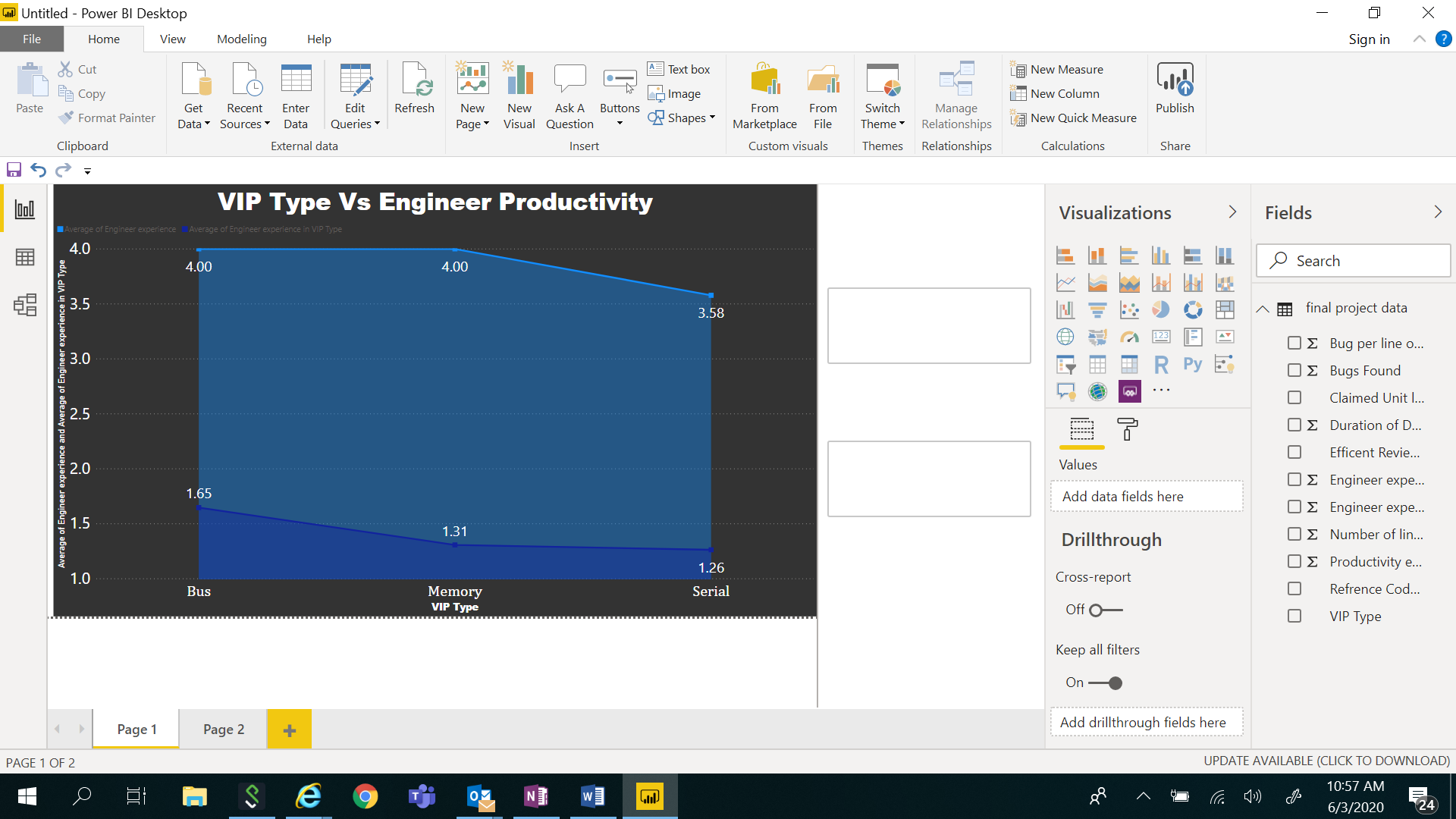


Figure 6‑3: Comparison between VIP type and Average Engineer experience

**Analysis basis on the above graph**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **VIP type** | **Engineer productivity** | **Bus per line** | **Engineer Experience** | **Engineer Experience in VIP** |
| **Bus** | **107.06** | **10.93** | **4** | **1.65** |
| **Memory** | **108.46** | **8.0** | **4** | **1.31** |
| **Serial** | **108.16** | **11.29** | **3.58** | **1.26** |

**Observation 1:**

* **Memory protocols are less complex protocol compare to others**
* **Engineers productivity has impact over all bugs**
* **Past experience in the VIP development play important role in the** **quality**

These observation matches with the Correlation data.

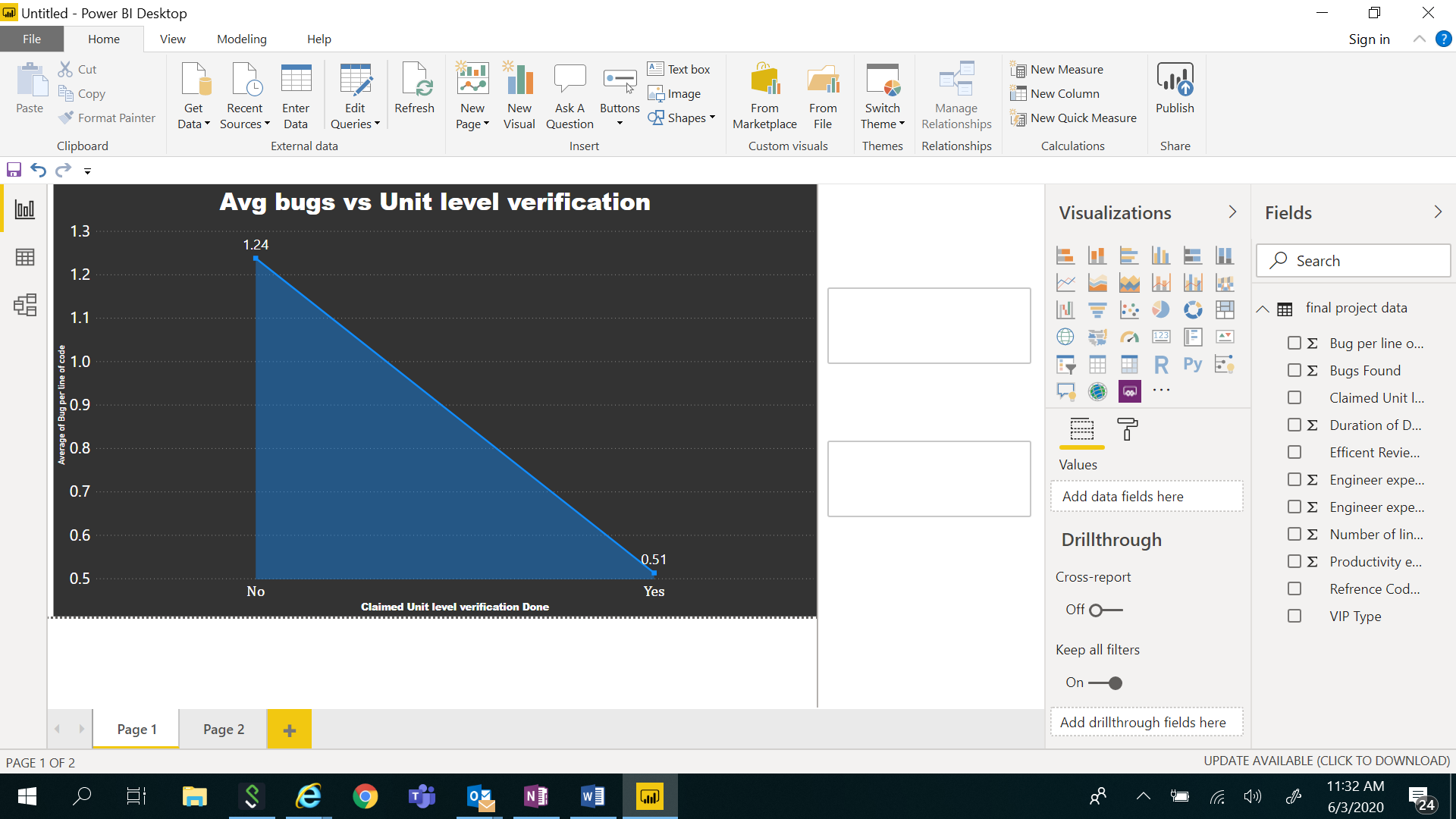


Figure 6‑4: Comparison between Avg Bus Vs Unit level verification

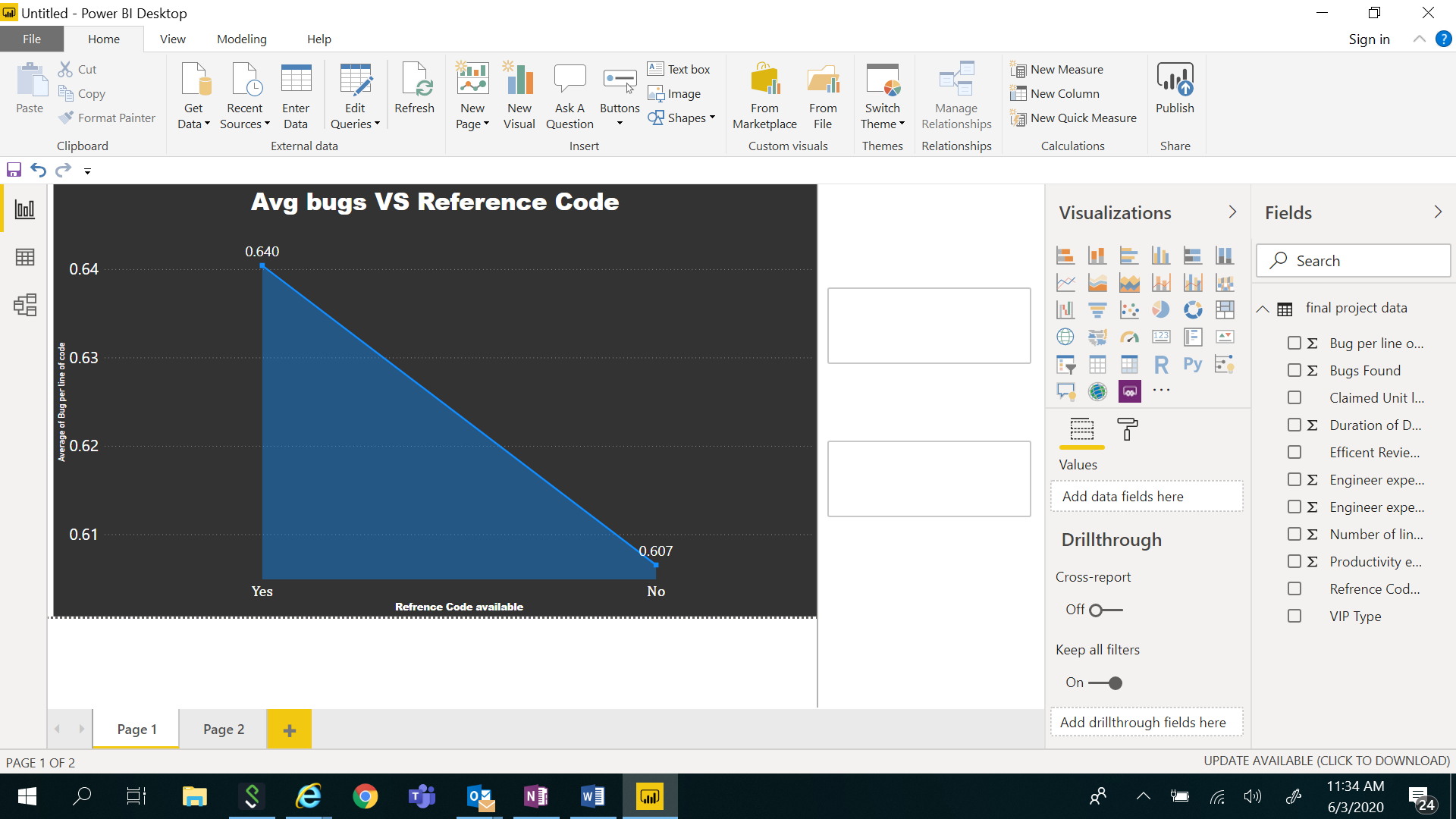


Figure 6‑5: Comparison between Avg Bus Vs Reference Code

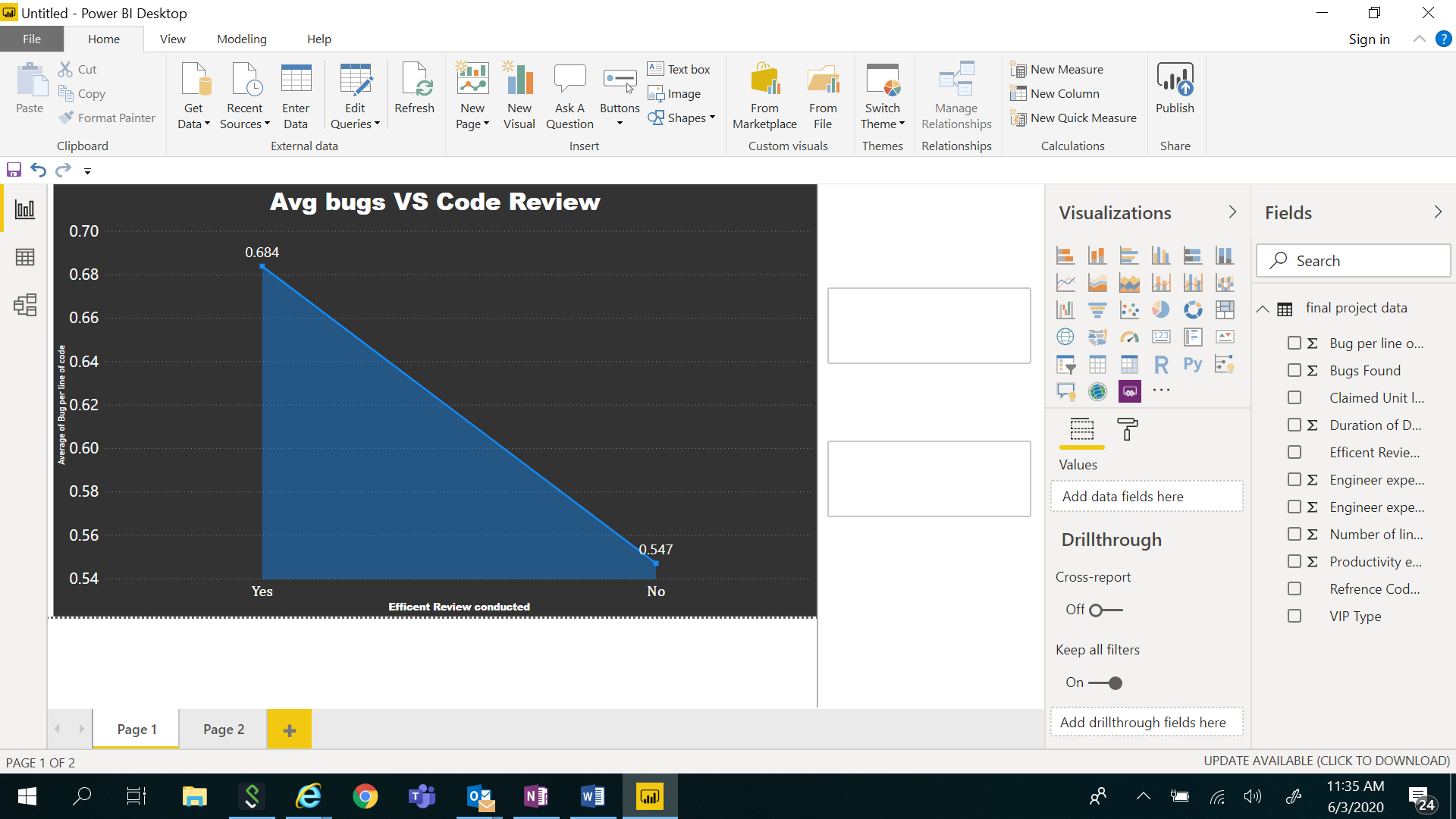


Figure 6‑6: Comparison between Avg Bus Vs Code Review

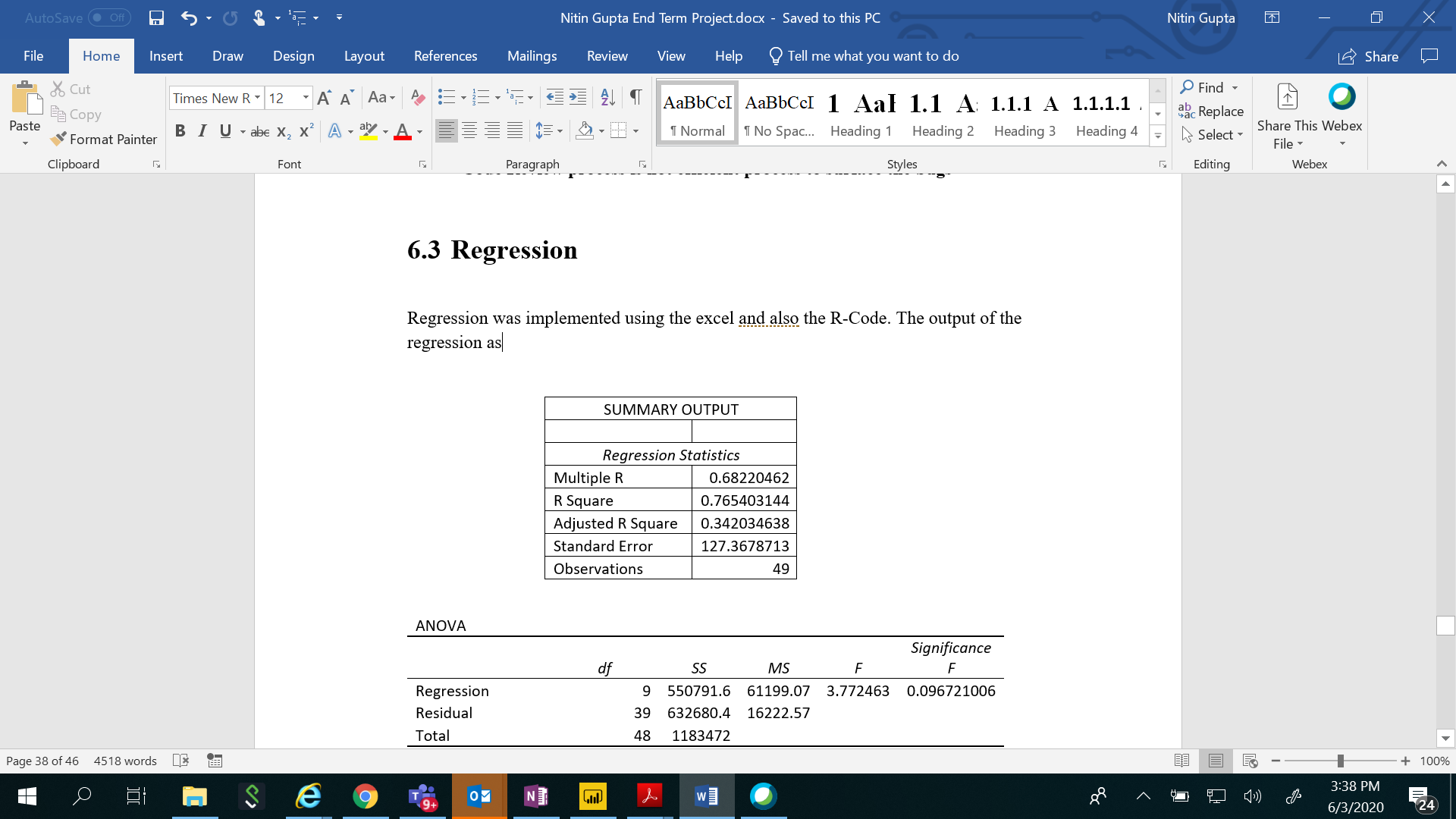
|  |  |  |  |
| --- | --- | --- | --- |
| **Answer** | **Code Review** | **Reference Code** | **Unit level verification** |
| **Yes** | **0.684** | **0.64** | **.51** |
| **No** | **0.547** | **0.607** | **1.24** |

**Observation 2:**

* **Unit level verification helps to increase the quality**
* **Code Review process is not efficient process to surface the bugs**

## Regression

Regression was implemented using the excel and also the R-Code. The output of the regression as



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| “ANOVA“ |  |  |  |  |  |
|  | *“df”* | *“SS”* | *“MS”* | *“F”* | *“Significance F”* |
| “Regression“ | 9 | 550791.6 | 61199.07 | 3.772463 | 0.096721006 |
| Residual | 39 | 632680.4 | 16222.57 |  |  |
| Total | 48 | 1183472 |  |  |  |

## Hypothesis

H0: all partial slopes are equal to 0

H1: At least one slope is not equal 0

As the significance level is 95% and the overall p value of the test came to 96% therefore the H0 is accepted.

## Conclusion

The overall all the variables play important role. Therefore this model can be used to predict the number of bugs that must be surfaced. As the effort invested resolving bugs before release to the customers is low cost, therefore this model can help in the following:

* Increase the prevention cost
* Decrease the failure cost
* Over all reduce the cost of Quality

This model also have following observation

* The company needs to improve the Code review process
* Experience of the engineer in the VIP development result in lower cost
* Prototype or reference code helps in reducing the bugs

# Refences

Figures reference

* <https://www.gminsights.com/>
* <https://hcs-pharma.com>
* <https://www.mathworks.com/>
* <https://quality-one.com/coq/>
* <https://www.udemy.com/>

Table Reference

* SNPS internal data

Content Reference

* <https://www.dataquest.io>
* <https://asq.org>
* <https://www.udemy.com/>

# Appendix

## R Code

“# Setting working director

setwd("C:\\Users\\nitingup\\Documents\\MBA\\3rdSemProjectWork")

#” Importing the dataset”

“dataset = read.csv('Data.csv')”

str(dataset)

#Checking data characteristics

dim(data45set)

str(data45set)

head(data45set)

tail(data45set)

# Taking care of missing data

#Manual Checking done therefore no missing data check is performed

# Encoding categorical data

dataset$CustomerCategory = factor(dataset$CustomerCategory,

levels = c('Tier1', 'Tier2', 'Tier3'),

labels = c(1, 2, 3))

dataset$TitleUsed = factor(dataset$TitleUsed,

levels = c('No', 'Yes'),

labels = c(0, 1))

dataset$Result = factor(dataset$Result,

levels = c('Loss', 'Win'),

labels = c(0, 1))

#Select training sample

train<-dataset[sampling,]

test<-dataset[-sampling,]

nrow(train)

nrow(test)

#Table of y for the train dataset

table(train$Good.Bad)

table(test$Good.Bad)

#Logistic Regression

myresult<-glm(data=train,Good.Bad ~ Check\_Account\_Status+CreditHistory,family=binomial)

summary(myresult)

exp(0.47369)

exp(0.681595)/(1+exp(0.681595)) # it gives the chances of a person being good when there is a unit change in A12.

#Finding Predicted Values

predicted <- myresult$fitted.values

summary(predicted)

#Confusion Matrix

predbkt<-ifelse(predicted>0.5,'1','0')

table(predbkt,train$Good.Bad)

predbkt<-ifelse(predicted>0.7,'G','B')

table(predbkt,train$Good.Bad)

#”Plotting ROC Curve”

“library(ROCR)”

“pred<-prediction(predicted,train$Good.Bad)”

“perf<-performance(pred,"tpr","fpr")”

“plot(perf)”

“auc<-performance(pred,"auc")”

auc<-unlist(slot(auc, "y.values")) #slot gives the information about the individual slot

# in an object”

## Data Set

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **VIP Type** | **Engineer experience** | **Engineer experience in VIP Type** | **Productivity expected** | **Number of lines coded** | **Reference Code available** | **Efficient Review conducted** | **Claimed Unit level verification Done** | **Duration of Development** | **Bugs Found** |
| Serial | 1.2 | 0 | 100 | 5000 | Yes | Yes | Yes | 3 | 5 |
| Bus | 2 | 0.5 | 110 | 1000 | No | No | Yes | 1 | 7 |
| Memory | 4 | 2 | 110 | 10000 | No | No | No | 5 | 110 |
| Bus | 5 | 1 | 120 | 15000 | Yes | Yes | No | 5 | 315 |
| Serial | 7 | 7 | 130 | 150000 | Yes | No | No | 12 | 100 |
| Memory | 5 | 2 | 100 | 3000 | No | Yes | No | 1 | 3 |
| Bus | 3.3 | 1 | 100 | 10000 | No | Yes | Yes | 3 | 6 |
| Serial | 2.5 | 0 | 100 | 50000 | Yes | Yes | Yes | 1 | 34 |
| Serial | 4 | 0 | 100 | 2000 | No | Yes | Yes | 3 | 15 |
| Memory | 5 | 3 | 100 | 1000 | No | Yes | Yes | 1 | 1 |
| Serial | 8 | 2 | 100 | 12000 | No | Yes | Yes | 3 | 8 |
| Bus | 3 | 2 | 100 | 7500 | Yes | No | Yes | 1 | 5 |
| Serial | 1.2 | 0 | 100 | 4000 | No | Yes | Yes | 1 | 4 |
| Serial | 1.5 | 0 | 110 | 7500 | No | Yes | Yes | 2 | 55 |
| Memory | 4 | 1 | 120 | 40000 | No | No | No | 10 | 840 |
| Serial | 3 | 2 | 130 | 8000 | Yes | Yes | No | 3 | 248 |
| Bus | 4 | 4 | 110 | 25000 | No | Yes | Yes | 4 | 177 |
| Serial | 2 | 1 | 110 | 5000 | Yes | Yes | Yes | 5 | 30 |
| Serial | 3 | 0 | 90 | 1000 | No | Yes | No | 1 | 1 |
| Memory | 4 | 0 | 100 | 10000 | No | No | Yes | 12 | 9 |
| Serial | 5 | 3 | 100 | 7500 | No | Yes | Yes | 8 | 5 |
| Serial | 5 | 0 | 100 | 1000 | Yes | Yes | Yes | 9 | 9 |
| Serial | 4 | 2 | 100 | 7500 | No | Yes | Yes | 7 | 5 |
| Memory | 3 | 1 | 100 | 11000 | Yes | No | Yes | 12 | 5 |
| Bus | 3 | 1 | 100 | 8000 | No | No | Yes | 4 | 7 |
| Memory | 4.5 | 3 | 100 | 10000 | Yes | No | Yes | 12 | 7 |
| Bus | 4 | 0 | 100 | 7000 | Yes | No | Yes | 5 | 7 |
| Bus | 4 | 1 | 100 | 6000 | Yes | No | Yes | 7 | 4 |
| Bus | 4 | 2 | 100 | 10000 | Yes | No | Yes | 12 | 9 |
| Serial | 5 | 2 | 105 | 25000 | Yes | Yes | Yes | 12 | 100 |
| Serial | 2 | 1 | 110 | 23000 | No | No | Yes | 15 | 148 |
| Bus | 8 | 3 | 120 | 23000 | No | No | Yes | 13 | 322 |
| Memory | 3 | 0 | 100 | 19000 | No | No | Yes | 12 | 13 |
| Memory | 4 | 1 | 120 | 1000 | No | Yes | Yes | 1 | 14 |
| Memory | 5 | 1 | 110 | 10000 | Yes | Yes | Yes | 4 | 50 |
| Memory | 6 | 1 | 110 | 23000 | No | No | Yes | 12 | 168 |
| Bus | 7 | 2 | 110 | 23000 | No | No | Yes | 15 | 177 |
| Bus | 2 | 2 | 120 | 21000 | No | No | Yes | 9 | 200 |
| Bus | 3 | 3 | 100 | 1000 | No | No | Yes | 1 | 5 |
| Serial | 5 | 2 | 100 | 10000 | No | Yes | Yes | 9 | 8 |
| Serial | 4 | 2 | 100 | 25000 | No | Yes | Yes | 12 | 25 |
| Bus | 6 | 2 | 100 | 30000 | No | No | Yes | 15 | 30 |
| Memory | 3 | 1 | 105 | 7500 | No | No | Yes | 4 | 45 |
| Memory | 2 | 1 | 110 | 21000 | No | No | Yes | 12 | 231 |
| Bus | 1 | 0 | 120 | 1000 | No | No | Yes | 1 | 21 |
| Serial | 2 | 0 | 100 | 10000 | No | Yes | Yes | 12 | 10 |
| Serail | 4 | 0 | 100 | 1200 | No | No | Yes | 1 | 3 |
| Bus | 4 | 1 | 100 | 23000 | No | Yes | Yes | 15 | 23 |
| Bus | 5 | 2 | 100 | 25000 | No | Yes | Yes | 12 | 25 |

## One hot encoding for VIP type

|  |  |  |
| --- | --- | --- |
| **VIP Type Serial** | **VIP Type Bus** | **VIP Type Memory** |
| 1 | 0 | 0 |
| 0 | 1 |  |
| 0 | 0 | 1 |
| 0 | 1 |  |
| 1 | 0 | 0 |
| 0 | 0 | 1 |
| 0 | 1 |  |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 |  |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 1 |  |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 0 | 0 | 1 |
| 0 | 1 |  |
| 0 | 0 | 1 |
| 0 | 1 |  |
| 0 | 1 |  |
| 0 | 1 |  |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 0 | 1 |  |
| 0 | 0 | 1 |
| 0 | 0 | 1 |
| 0 | 0 | 1 |
| 0 | 0 | 1 |
| 0 | 1 |  |
| 0 | 1 |  |
| 0 | 1 |  |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 0 | 1 |  |
| 0 | 0 | 1 |
| 0 | 0 | 1 |