Benefits of Chatbot based AI solutions for Businesses: Return on Investments on Chatbots using Monte Carlo simulations.

Anupam Banerjee

February 15, 2022

Abstract

Chat bot systems are extensively used in modern day to automate many processes related to customer engagement, and resolution and assistance. Chatbot based on Natural Language Processing (NLP) systems for example virtual assistants have proven to be highly effective in comprehending complex customer queries and providing appropriate solutions to customers and in turn improve customer engagement, enhances customer satisfaction and increased customer retention. However software engineering life cycle for chatbot system is has several components of uncertainties throughout its life cycle in general. The uncertainty is based on previous experiences, changing technological landscape, Competence and skill set, complexity of the requirements and randomness of key factors like customer response to new system. This makes it more difficult for project managers and business owners to implement these projects and hence they continue to have onshore and off shore support staff to handle a large portion of customer queries. Implementation of such systems has a lot of pros as it solves the problem pertaining to conventional onshore/off shore support is the inability to handle large volumes of call and customer queries and lack of availability and inability to handle multiple queries in Real-time . Just like any other software project, developing of such chat bots also has uncertainties in terms of parameters such duration of the project, accuracy of the chatbot in understanding the query and providing solution. The current research that has been done on the field to estimating costs, duration, efforts and benefit of any Artificial intelligent software application like chatbot uses formulas and equations to make estimates. These formulas does not take into account these uncertainties. Here, through this research report I propose a novel and robust framework of estimating the Return on investment (ROI) by quantifying all the general benefits and costs that can be attributed to a chatbot project life cycle and incorporating the uncertainties in the calculations for accurately estimating the ROI of implementing a Chatbot from scratch. This research makes three contributions. The first contribution is proposition of a novel framework to estimate the ROI of chatbot software application that include modelling uncertainty using Monte Carlo simulations. Second, is the creation of simulated dataset that records 45 features related to chatbot project life cycle and the final contribution is creation of machine learning regression models that fit well with the created dataset and make close prediction of ROI.

1 Introduction

In today's and age Artificial Intelligent solutions contribute to business in a variety of ways, whether that be by optimization of current business operations, providing insightful for data driven decision making, increased level of satisfaction for enterprise customer or employees, reducing cost of business operations and customer engagement, enables being ahead of the curve in technology demonstration and having an edge over their competitors. However, many enterprises or business that adopt these AI solutions often overlook how beneficial and effective these solutions are in achieving their business objectives. There are not many frameworks that can effectively capture the broader benefits of a specific AI solution and compare them with the efforts to have an objective estimate of profit of adopting such systems by quantifying benefits for comparison with costs. The best way in my opinion to quantify the benefits of AI is to calculate the Return on Investment (ROI) for an AI solution. The ROI can be further classified as Hard and Soft ROI. Hard ROI can be described as benefits that can be easily quantifiable in terms of financial gain or profit and soft ROI generally translate to other

important aspects of business, like customer and employee satisfaction, brand image etc. With the help of this research project, I would like to provide an all-rounded framework that can effectively gauge the ROI on chatbot AI solutions that can quantify benefits of the AI solution for a specific business and try to answer which features play a very crucial role in determining whether or not the benefits of AI will outweigh the cost of adoption of AI and translate to profitability for the enterprise or business. Quantifying benefits of AI solutions comes with its own set of challenges. One such challenge is to be able to estimate the future ROI of the project that the enterprise will reap throughout the lifetime of the project and through project reproduction. Also, quantifying Soft ROI benefits in terms of monetary profit is a huge challenge. For example- CNN for computer vision problem enable automatic video surveillance 24x7 which has great financial value which may be full of uncertainty. This research project aims at identifying the all the avenue in which the Chatbot AI solution help the enterprise and evaluate the contribution of the solution in saving time, increase productivity, saving cost, increase revenue, enhance user experience, and indirectly increase worker satisfaction, retain talent, and increase agility. The research aims at predicting the ROI (soft and hard) and aggregate scores of these benefits to accurately quantify the ROI based on enterprise requirements. The goal is to create a robust framework that estimate the benefits keeping in mind the dynamic nature of features like changing cost of the project, cost of error of the AI and deteriorating accuracy of the models over time and profits by replicating model capabilities on other problems and profits from copyright technologies. This research also try to estimate ROI of chatbot applications based on specific client requirements, adopted methodologies, third party API and software etc. Based on the research we can analyse the features relevant to the problem and adapt in an iterative fashion. Please, note that the problem statement and the methodology addressed in this research project is generalized and can be modified easily for estimating ROI for other AI applications.

In this research report we discuss various relevant features organizational, qualitative or quantitative that may influence the return on invest a Chat bot development life cycle. I have proposed a novel way of creating a framework that takes these relevant features and estimated accurately and gives statistical probability of the project being successful i.e. the benefits of conducting the project significantly outweigh the costs.

2 Dataset Used

We have created a dummy dataset due to lack of Software development life cycle (SDLC) data

2.1 SEERA Dataset of SDLC

1. To provide up to date dataset with traditional cost attributes in addition socio-economic and organizational attributes. The dataset projects represent constrained technical and economic software development environments.

2. The dataset fills an urgent gap for the Sudanese and African research community with a more relevant cost estimation dataset that includes factors more aligned with the realities of their software industries.

3. SEERA dataset overcomes the current limitations in dataset quality and transparency by augmenting the cost estimation dataset with the original raw data before coding/scaling.

The SEERA dataset is used for software development cost estimation but I have created a customized dummy dataset along with added features that is able to gauge the Benefits as well. The Features Description and relevance of each features are given in the appendices section of the report.

2.2 Dataset Created

Include 1000 rows and 45 columns of randomly generated data using Monte Carlo simulation that we will see in the next sections. The attributes are majorly taken from SEERA dataset and few relevant features have been added.

3 Methodology

In-order to calculate the ROI of Chatbot projects I have made dummy dataset based on SEERA dataset for software engineering cost estimation. There are total 51 features 1000 records created to gauge the various aspects of a chatbot project. I have simulated the dataset for 7 different organizations and associated static features like IT department size, team size, team experience corresponding to each of the organizations. For simplicity I have used the same set of static features for organization related attributes. Apart from the organization related features I have used another set of static features like software quality indicators, programming languages used and degree of software reusability as standards associated to the chatbot project development type which include categories like agile, scrum and Lean. The other features in the dataset include numerical data and categorical data that require different set of treatments which we will see in the later sections.

I have included the uncertainty in numerical data with the help of Monte Carlo simulation using normal, beta and Gamma probability distribution functions using the Numpy Package in python. To visualize these distributions I have used Matplotlib and seaborn packages in python. After random variables are generated based on a specific distribution, I have also introduced correlations to mimic real life data that may not necessarily independent of one another rather may be correlated to an extent. The methods of correlations are discussed in the later sections. The discrete numerical data is handled same way as continuous numerical features accept that these features are rounded of using the round() function in python.

To simulate the categorical features I have made use of weighted random choices in python. This is done to mimic the real life scenarios where the probability of one category appearing is usually higher than the other instead of being the same. For instance - The in the feature of cloud computing tier, the probability of a middle tier computing instance used is greater than a higher tier computing instance being used.

3.1 Feature Engineering

Feature Engineering is the process of selecting and transforming the most relevant features has the ability to effectively explain the variation in the target variables. We require feature engineering since, in every machine learning algorithms use some sort of input data to create outputs. This input data comprising of several features, are represented usually in the form of structured columns. Algorithms require features to be compatible to the algorithm's requirements. A well structured, compatible and clean dataset with a precise set of features can potentially give highly accurate results for classification and regression.

For the purpose of feature engineering I have used the following techniques.

3.1.1 Max Min Scaling

Max Min Scaling is one of the simplest form of scaling techniques that takes numerical discrete and floating point data and returns a value between 0 and 1. This kind of transformation is necessary for numerical features since all the values of the input data has to be in the same scale so that there are not any bias for higher values. I have used the MinMaxScaler from scikit learn package in python. The Formula for MinMax scaling is given below.

$$x_{std} = \frac{x_i - x_{min}}{x_{max} - x_{min}}$$

Here, x_i is the input data, x_{min} is the minimum value of the feature x and x_{max} is maximum value of the feature x.

3.1.2 One hot encoding

One hot encoding is the method of introducing dummy variable that takes the value 0 or 1 for every single category of the categorical variable. This is done encode the category into numerical form as machine learning algorithm can only accept numerical data.

I have saved the dataset after feature engineering into a separate file called to be used as input data for our machine learning algorithm.

3.2 Monte Carlo Simulation

In real world we are constantly faced with uncertainty, ambiguity, and variability. Given the unprecedented access of data we can harness for any decision making task uncertainty modelling can help make more accurate and informed decision based on various statistical and analytical techniques. Monte Carlo simulation lets you simulate all the possible outcomes by simulating data and randomly selecting from a range of possible values and evaluate the risk, allowing for better decision making under uncertainty. It is a mathematical technique that allows people to account for uncertainties and risk in quantitative analysis and decision making. The technique is used by professionals in such widely especially in research and development for simulating different scenarios how that impact outcomes.

It enables decision makers with informed decisions based on all set . It shows the extreme possibilities and give an idea of the range of outcomes.

Using Monte Carlo simulation we can generate random values based on any probability distribution function PDF $f(x)$. (youtube channel/montecarlo1.ipynb at main · lukepolson/youtube channel, 2022)

Theorem : If a random variable X that has a cumulative distribution function (CDF) of $F(x)$ then the $F^{-1}(U)$ also has the CDF of $F(x)$

$$X \sim F(x) \Rightarrow F^{-1}(U) \sim F(x)$$

Above X is random variable denoted by f(x) where F(x) is the Cumulative Density Function (CDF) and U is uniform random variable between 0 and 1, where $F^{-1}(U)$ has the same distribution as F(x) For example: The exponential function

$$f(x) = \lambda e^{-\lambda x} \Rightarrow F(x) = 1\ e^{-\lambda x}$$

Using this technique we can transform any uniform random variable into a CDF of any random variable where the above criteria are satisfied. For simplicity I have used predefined distributions using the Numpy.random package in python.

3.2.1 Normal Distribution (Gaussian Distribution)

The Normal distribution or Gaussian distribution denoted by $N(\mu, \sigma^2)$ where μ is mean and σ is the standard deviation.

The formula for normal distribution is given below

$$P(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}$$

The many real world quantities follow the normal distribution. Hence, I have used normal distribution as the distribution for several random variables in the dataset.

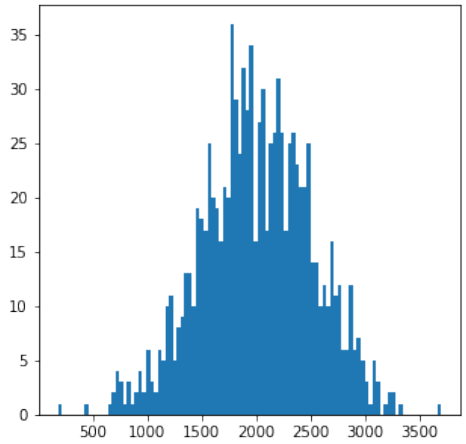

Figure 1: Visualization of random numbers generated using normal distribution

3.2.2 Gamma Distribution

Gamma Distribution is also a widely used probability distribution function. It is an extension of the factorial function for real and complex numbers.

Formula for Gamma distribution is given below.

$$\Gamma(a) = \int_0^\infty s^{a-1} e^{-s} ds$$

Here α can be any real or complex number. if α = n then

$$n! = n.(n-1)!$$

Here in the figure below we can see that Gamma is a right skewed distribution.

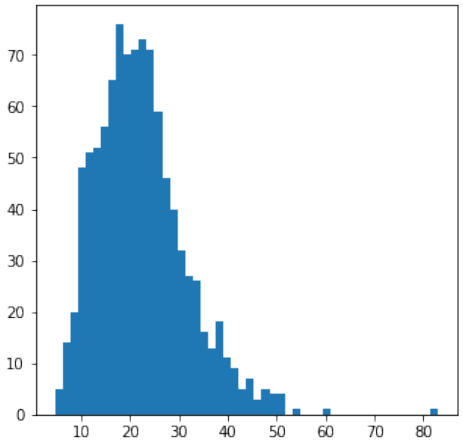

Figure 2: Visualization of random numbers generated using Gamma distribution

3.2.3 Beta Distribution

Beta distribution is also a continuous probability distribution function denote by two parameters α and β that determines the shape of the distribution. The beta distribution provides a value between 0 and 1. Formula for Beta distribution is given below.

$$f_X(x : \alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1}$$

I have used Beta distribution for simulating the Project Duration feature.

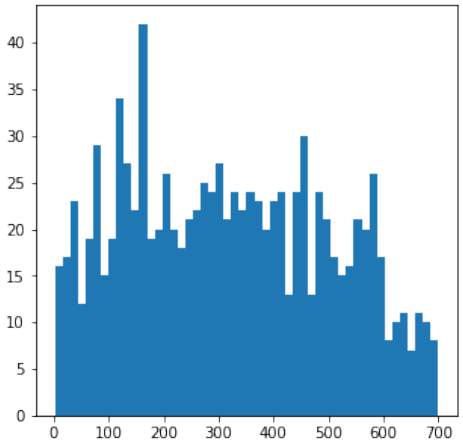

Figure 3: Visualization of random numbers generated using Beta distribution

3.3 Introducing correlations

Since, all the numeric variables that are generated using the above probability distributions are random therefore, there is negligible correlation between each variables. Since, in real world related features are correlated I have introduced correlations in a specific feature by adding and value from the correlated feature multiplied a correlation coefficient.

For example: c1 = 0.4
df['IT dpt size'] = df['IT dpt size'] + c1 * df['Organization size']

Here, c1 is the correlation coefficient and df is the data frame.

The correlation matrix after the transformation is given below. The dark color show negative correlation and lighter color shows positive correlation.

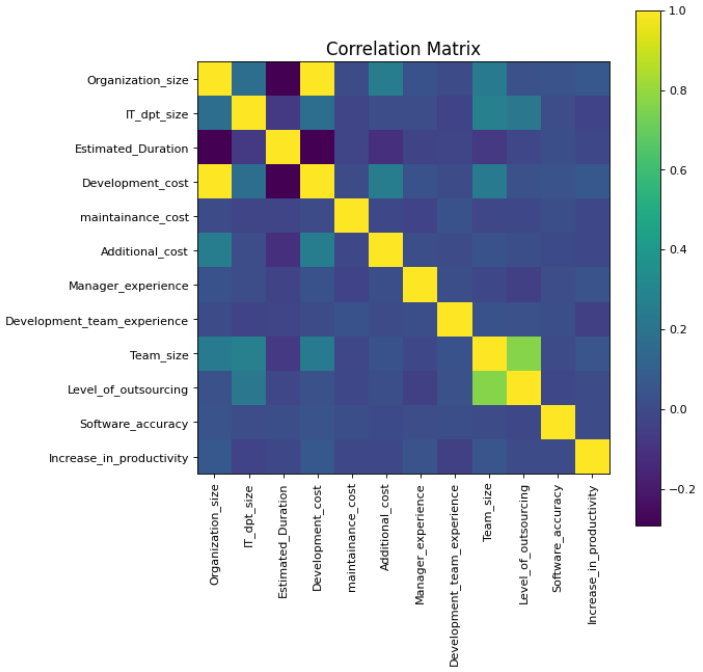

Figure 4: Correlation matrix of correlated feature

3.4 Calculating costs

The total cost of the chatbot project is calculated using the below formula.

$$C_t = 12 \times \sum C_{1 \to n} \times Y \times e$$

Here, $\sum C_{1 \to n}$ is the sum of n monthly cost components. Y is the number of years calculated by

$$Y = \frac{Estimated\ duration\ in\ days}{365}$$

and e is the error correction coefficient.

3.5 Calculating Benefits

Total Benefits is by implementing and adopting a chatbot application can be calculated by quantifying the qualitative indicators of the chatbot project. This can be thought as the amount of money an organization saves by adopting the application. The monetary sum the organization saves is determined by the quality attributes. Calculated using the below formula

$$Benefits = (\frac{2}{n})(\sum_{q=1}^{n} S_q) \times (1 + \triangle R) \times Y \times C_o$$

S_q are scaled project quality attributes between 0 and 1, $\triangle R$ is the percent change in revenue, Y is the Project Duration in years and C_o is the Annual outsourcing cost.

3.6 Calculating ROI

$$ROI = \frac{Estimated\ total\ Benefits\ -\ Estimated\ total\ Costs}{Estimated\ total\ Costs} \times 100$$

3.7 Machine Learning for Estimation

1. Decision Tree -

   Decision trees are tree-structured algorithms for classification and regression tasks.

   Used Decison Tree Regressor from scikit learn package in python for training and predicting ROI values.

2. Random Forest Classifier-

   Random forest is a Supervised Machine Learning Algorithm also used in Classification and Regression problems. Takes different subset of sample data and create decision trees and takes their majority vote for classification and average in case of regression. This method method is called ensemble method.

   Used Random forest Regressor from scikit learn package in python for training and predicting ROI values.

3. Support Vector Regressor/Machine (SVR/M)-

   Same as support vector machine Support vector regressor uses the same logic for regression task. SVR essentially can be considered as a single layer neural network.

4 Results

The below Illustration show the probability distribution of the calculated Return on Investment after the same amount of time taken to complete the project. The ROI curve shows promising result ranging from 0.5 to 8 times return.

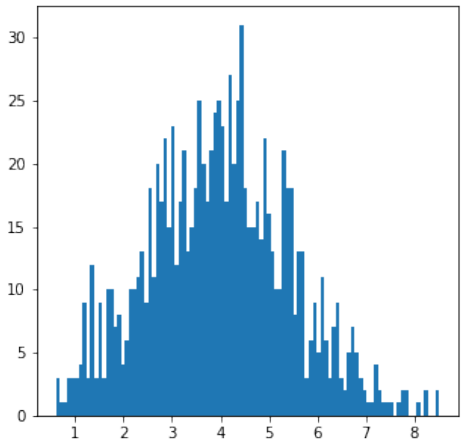

Figure 5: ROI probability distribution

The curve shows great return on investment that peak at

4.1 Mean Absolute Error

Mean Absolute Error denote the Absolute difference between the predicted and the actual value. Denoted by the formula

$$MAE = \sum_{i=1}^{D} |x_i - y_i|$$

here, x_i, and y_i are the predicted and actual value.

4.2 Root Mean Squared Error

- To compute the RMSE, we calculate the residual i.e. difference between prediction and actual for each data point, compute the normal of residual for each data point, compute the mean of residuals and take the square root of it. Denoted by the formula

$$RMSE = \sqrt{\sum_{i=1}^{D} (x_i - y_i)^2}$$

No.	Algorithm Name	RMSE	MAE
1	Decison Tree	1.4336	1.1205
2	Random Forest	0.9875	0.7908
3	SVM/R	1.0035	0.7851

The Results obtained by the supervised machine Learning models given above.

References

1. (3.3), G. and Weckesser, W., 2022. Generate correlated data in Python (3.3). [online] Stack Overflow. Available at: ¡https://stackoverflow.com/questions/16024677/generate-correlated-data-in-python-3-3¿ [Accessed 15 February 2022].

2. scikit-learn. 2022. 1.10. Decision Trees. [online] Available at: ¡https://scikit-learn.org/stable/modules/tree.html¿ [Accessed 15 February 2022].

3. Engati. 2022. 6 types of chatbots - Which is best for your business? — Engati. [online] Available at: ¡https://www.engati.com/blog/types-of-chatbots-and-their-applications?utmcontent=types-of-chatbots-and-their-applications¿ [Accessed 15 February 2022].

4. Malathi, S. and Sridhar, S., 2012. Performance Evaluation of Software Effort Estimation using Fuzzy Analogy based on Complexity. International Journal of Computer Applications, 40(3), pp.32-37.

5. Latex.codecogs.com. 2022. Online LaTeX Equation Editor - create, integrate and download. [online] Available at: ¡https://latex.codecogs.com/eqneditor/editor.php¿ [Accessed 15 February 2022].

6. Pandas.pydata.org. 2022. pandas - Python Data Analysis Library. [online] Available at: ¡https://pandas.pydata.org/¿ [Accessed 15 February 2022].

7. Matplotlib.org. 2022. Pyplot tutorial — Matplotlib 3.5.1 documentation. [online] Available at: ¡https://matplotlib.org/stable/tutorials/introductory/pyplot.html¿ [Accessed 15 February 2022].

8. Resende, C., Grace Heckmann, C. and Michalek, J., 2012. Robust Design for Profit Maximization With Aversion to Downside Risk From Parametric Uncertainty in Consumer Choice Models. Journal of Mechanical Design, 134(10).

9. scikit-learn. 2022. sklearn.ensemble.RandomForestRegressor. [online] Available at: ¡https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html?highlight=random

10. scikit-learn. 2022. sklearn.svm.SVR. [online] Available at: ¡https://scikitlearn.org/stable/modules/generated/sklearn
[Accessed 15 February 2022].

11. GitHub. 2022. youtubechannel/montecarlo1.ipynb at main · lukepolson/youtubechannel. [online] Available at: ¡https://github.com/lukepolson/youtubechannel/blob/main/Python

12. Khalifelu, Z. and Gharehchopogh, F., 2012. Comparison and evaluation of data mining techniques with algorithmic models in software cost estimation. Procedia Technology, 1, pp.65-71.

13. Arslan, F., 2019. A Review of Machine Learning Models for Software Cost Estimation. Review of Computer Engineering Research, 6(2), pp.64-75.

14. Suwanjang, H. and Prompoon, N., 2012. Framework for Developing a Software Cost Estimation Model for Software Modification Based on a Relational Matrix of Project Profile and Software Cost Using an Analogy Estimation Method. International Journal of Computer and Communication Engineering, pp.129-134.

15. 2022. [online] Available at: ¡https://www.researchgate.net/publication/346734027SEERA A software cost estimation dataset for constrained environments¿ [Accessed 15 February 2022].

16. B. Boehm, L. Huang, A. Jain and R. Madachy, "The ROI of software dependability: The iDAVE model," in IEEE Software, vol. 21, no. 3, pp. 54-61, May-June 2004, doi: 10.1109/MS.2004.1293073.

17. G. Bockle, P. Clements, J. D. McGregor, D. Muthig and K. Schmid, "Calculating ROI for software product lines," in IEEE Software, vol. 21, no. 3, pp. 23-31, May-June 2004, doi: 10.1109/MS.2004.1293069.

18. C. Jones, "Economics of software reuse," in Computer, vol. 27, no. 7, pp. 106-107, July 1994, doi: 10.1109/2.299437.

19. R. van Solingen, "A Follow-Up Reflection on Software Process Improvement ROI," in IEEE Software, vol. 26, no. 5, pp. 77-79, Sept.-Oct. 2009, doi: 10.1109/MS.2009.120.

20. En.wikipedia.org. 2022. Normal distribution - Wikipedia. [online] Available at: ¡https://en.wikipedia.org/wiki/Norm
distribution¿ [Accessed 15 February 2022].

21. En.wikipedia.org. 2022. Gamma distribution - Wikipedia. [online] Available at: ¡https://en.wikipedia.org/wiki/Gam
distribution¿ [Accessed 15 February 2022].

22. En.wikipedia.org. 2022. Beta distribution - Wikipedia. [online] Available at: ¡https://en.wikipedia.org/wiki/Beta
distribution¿ [Accessed 15 February 2022].

Appendices

Below are the given attributes recorded based on the SEERA dataset used for software cost estimation along with the description and relevance to the calculation of the ROI.

1. Year of project
2. Organization id Organization type
3. Role in organization
4. Size of organization
5. Size of IT department
6. Customer organization type
7. Estimated duration
8. Actual duration

9. project (gain/loss)
10. Development type
11. Application Domain
12. Contract Date
13. Contract software delivery date
14. Actual software development
15. start date
16. Year of project (Main)
17. Contract price
18. Actual incurred costs
19. project gain (loss) (Main)

Below is the description and relevance of each and every variable used.

1. Project ID - .
2. Organization -
3. Project start date - This is along with project end date is a very crucial feature that can help analyse in kind of season changes in project initiation and delivery.
4. Project end date - Same as Project start date used to identify seasonality component if any
5. Organization size - This feature denotes the scale of the organization implementing the chatbot. It is very crucial feature that has correlations with many other features of the dataset that we will see in the next items.
6. IT dpt size - This feature denotes the size of the Information Technology services department in the company.
7. Customer type - Categorical dummy feature that gauge the kind customer being serviced by the project.
8. Estimated Duration - This is estimated time frame to complete the project calculated prior project start
9. Actual Duration - This is Actual project completion duration. The combination can be used to model the error in estimation for much closer and accurate estimates.
10. Project Profit Loss percent - Shows the Profit or loss percent of project.
11. Project type another dummy categorical variable that is used to gauge any specific style of software development like scrum or agile.
12. Development type project management model like waterfall, COCOMO etc
13. Estimated effort One of the very important features used for estimating cost, benefits and Overall ROI. Shows the total number of estimated hours of required to complete the project. This feature is full of uncertainty.
14. Actual effort The Actual amount of efforts in hours taken to complete the project. Used to model the errors for more accurate estimate of efforts.
15. Management clarity Categorical data showing Level of Clarity from management and client.
16. Training effort Number of hours taken for upskilling and training purposes.

No.	Feature Name	Description
1	Project ID	Index value ID number
2	Organization	Categorical nominal
3	Project start	Categorical Date time value
4	Project end	Categorical Date time value
5	Organization size	Discrete Ordinal Value
6	IT dpt size	Discrete Ordinal Value
7	Customer type	Categorical nominal
8	Estimated duration	numerical Discrete value
9	Profit/loss percent	numeric continuous
10	Project type	Categorical Nominal
11	Development type	Categorical Nominal
12	Estimated effort	numeric continuous
13	Management clarity	numerical Discrete
14	Training effort	numeric continuous
15	User resistance	numerical ordinal Discrete
16	Software licence cost	numeric continuous
17	Hardware cost	numeric continuous
18	Development cost	numeric continuous
19	Maintenance cost	numeric continuous
20	Additional cost	numeric continuous
21	Total cost	numeric continuous
22	Requirement stability	numeric continuous
23	Manager Experience	numeric discrete
24	Development team cost	numeric continuous
25	Maintenance team cost	numeric continuous
26	ML Team cost	numeric continuous
27	Data analysis team cost	numeric continuous
28	Team size	numerical Discrete
29	Daily work hours	numerical Discrete
30	Level of outsourcing	numerical Discrete
31	Software availability	numerical Discrete
32	Software Stability	numerical Discrete
33	Software accuracy	numerical Discrete
34	Cloud Platform type	Categorical Nominal
35	Computing Tier	Categorical Ordinal
36	Programming Lang used	Categorical Nominal
37	DB used	Categorical Nominal
38	Open source tool used	Categorical Nominal
39	Increase in Productivity	numeric continuous
40	Change in Average wait time	numeric continuous
41	Change in Average call duration	numeric continuous
42	Change in Revenue	numeric continuous
43	Annual outsourcing cost	numeric continuous

17. User resistance This feature is crucial to gauge and estimate response customer, correlated to customer type feature.

18. Software licence cost Cost of any proprietary software cost.

19. hardware cost Cost of Hardware like Servers, laptops, desktops, office setup etc.

20. Development cost Cost of Development including developers salary.

21. maintenance cost Cost of maintenance including Support Engineers salary.

22. Additional cost Other costs and overheads.

23. Estimated total cost - Total estimated cost denoted by sum of the above 5 values.
24. Actual total cost - Total cost of project after delivery correlated to duration of the project and estimated total cost.
25. Requirement stability Categorical Data gauging the how stable client requirements are for the project.
26. Manager experience Manager Experience in Years
27. Development team experience Development team Experience in Years
28. Maintenance support team experience In Years
29. Machine learning team experience ML and NLP data engineer and Data scientist experience in years.
30. Data Analysis team experience in Years
31. Team size Count of Team members required to complete the project.
32. Daily work hours Hours of work per Day
33. Level of outsourcing Very important feature for calculation of ROI as heavily correlated to ROI as higher level of outsourcing shows reduction of operations and outsourcing cost.
34. Software availability Quantified benefit of solution. Higher the value lesser the down time.
35. Software accuracy Accuracy measured by software evaluation metrics in resolving queries.
36. Software stability Shows level of stability, higher the stability lesser the glitches and bugs in Solution.
37. Programming language used Categorical data to analyse trends of programming languages used.
38. DB used Categorical data to analyse trends of Databases used.
39. Open source tool used Categorical data to analyse trends of any other licensed software used.
40. Degree of software reuse Very important feature that potentially exponentially increase the benefits and give higher ROI.
41. Cloud Platform used Categorical data denoting cloud platform used like AWS, Azure, GCP etc.
42. Cloud Tier - Tier of computing and storage used correlated to cost of Project.
43. Annual Outsourcing Cost Very important feature that shows the annual cost of outsourcing operations.
44. Increase in productivity Indicator of extent to which productivity increased that indirectly result in higher ROI.
45. Change in waiting time Indicator of extent to which waiting time decreased that indirectly result in higher ROI
46. Change avg wait time Indicator of extent to which average waiting time decreased that indirectly result in higher ROI
47. Change avg call duration Denote average increase or decrease in call duration that is correlated to productivity.
48. Change in revenue Overall change in revenue after time after project completion date
49. Overall ROI Return on Investment of the Project calculated using all the relevant features.

Project_ID	Organization	Project_start_date	Project_end_date	Organization_size	IT_dpt_size	Customer_type	Estimated_Duration	Actual_Duration	Project_Profit_Loss_percent	Project_type	Development_type
0	Company_2	11/02/2019	24/03/2019	3	6.2	2	40.1	159	71.49389552	Linguistic Based	Extreme Programming
1	Company_7	18/06/2021	4/12/2023	4	4.6	7	897.8	303	10.74828301	Machine Learnin	Waterfall
2	Company_5	20/12/2020	9/12/2022	5	7	5	717.5	6	45.65882675	Machine Learnin	Lean
3	Company_7	17/06/2019	1/12/2019	4	4.6	7	165.8	124	32.99933018	Linguistic Based	Scrum
4	Company_2	21/12/2022	10/12/2026	3	6.2	2	1449.1	188	16.71560674	Machine Learnin	Extreme Programming
5	Company_1	15/08/2020	30/03/2022	4	3.6	1	590.8	627	38.27109261	button-based	Waterfall
6	Company_2	10/01/2020	18/01/2021	5	7	2	372.5	281	68.23503406	Voice Bots	Extreme Programming
7	Company_5	9/08/2021	17/03/2024	4	6.6	5	949.8	130	35.24027571	Keyword recogn	Agile
8	Company_4	13/02/2022	28/03/2025	4	3.6	4	1137.8	82	74.77067959	Hybrid model	Lean
9	Company_6	1/09/2021	2/05/2024	2	4.8	6	973.4	197	17.58566254	Machine Learnin	Waterfall
10	Company_4	19/02/2021	10/04/2023	4	3.6	4	778.8	221	33.78724082	Hybrid model	Scrum
11	Company_2	14/05/2019	24/09/2019	4	6.6	2	131.8	730	93.59733455	Linguistic Based	Agile
12	Company_4	1/05/2020	30/08/2021	6	4.4	4	484.2	504	57.43200828	Voice Bots	Extreme Programming
13	Company_3	21/02/2020	12/04/2021	3	5.2	3	415.1	400	32.59540785	Hybrid model	Scrum
14	Company_3	11/03/2022	19/05/2025	6	6.4	3	1163.2	168	33.27342354	button-based	Feature-Driven
15	Company_4	13/03/2022	23/05/2025	3	3.2	4	1166.1	646	29.60674104	Linguistic Based	Agile
16	Company_7	1/08/2021	1/03/2024	3	4.2	7	942.1	250	49.613405	Keyword recogn	Extreme Programming
17	Company_1	18/10/2019	3/08/2020	3	3.2	1	289.1	92	31.52005915	Hybrid model	Lean
18	Company_5	27/06/2020	22/12/2021	4	6.6	5	541.8	510	62.85870723	Linguistic Based	Scrum
19	Company_5	5/12/2021	8/11/2024	4	6.6	5	1067.8	160	48.91247747	Voice Bots	Extreme Programming
20	Company_2	28/03/2019	22/06/2019	4	6.6	2	84.8	349	53.79341259	button-based	Feature-Driven
21	Company_3	10/11/2022	19/09/2026	4	5.6	3	1407.8	261	43.92948694	Keyword recogn	Agile
22	Company_2	3/07/2020	3/01/2022	5	7	2	547.5	219	42.80376983	Voice Bots	Extreme Programming
23	Company_6	3/11/2020	6/09/2022	4	5.6	6	670.8	134	40.86803572	button-based	Scrum
24	Company_5	17/08/2021	2/04/2024	4	6.6	5	957.8	195	46.18226205	Voice Bots	Feature-Driven
25	Company_1	21/04/2020	10/08/2021	6	4.4	1	474.2	380	34.88506371	button-based	Scrum

Figure 6: Illustration of the dataset created

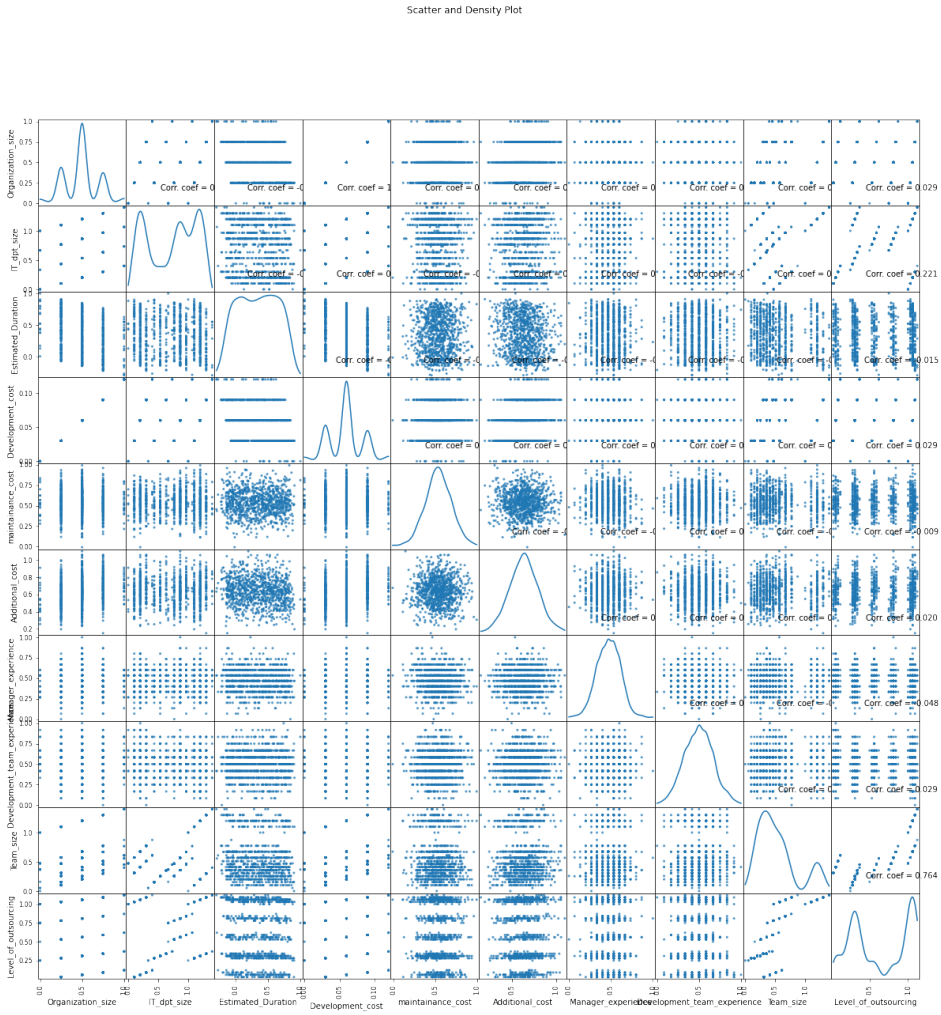

Figure 7: Scatter plot showing of correlation between variables