Mathematical Theory and Modeling
ISSN 2224-5804 (Paper) ISSN 2225-0522 (Online)
Vol.10, No.3, 2020

Data Analytics for Numeric Modeling and Its Application to an Offsite Concrete Block Production Operation

Ronald Ekyalimpa[1*] Isaiah Tugumenawe[2]

1. College of Engineering, Art, Design, and Technology, Makerere University, PO box 7062, Kampala, Uganda
2. China National Complete Plant Import & Export Corporation Ltd (COMPLANT), PO box 23159, Lugogo House, Plot 42, Logogo By-Pass, Kampala, Uganda

* E-mail of the corresponding author: rekyalimpa@gmail.com

Abstract

The majority of real-world systems within the Engineering domain and particularly the construction sector, generate enormous amounts of data every instance of time that they are in operation. This data can be collected from these systems real-time or otherwise using traditional methods or using contemporary techniques such as those that facilitate the implementation of concepts such as the Internet Of Things (IOT). Once gathered into a repository, this data can be utilized for planning, predictive, diagnostic, and other purposes. For this data to be put to such meaningful uses, there are analytics that need to be performed. This paper showcases typical examples of such analytics that generate information that can serve as decision support in a practical setting. First, background information that is necessary to support simple to complex data analytics is presented. This is followed by a case study used to demonstrate how analytics can be performed on data from an offsite concrete block production operation to gain insights into the operation and for diagnostic purposes. To achieve this, probability distributions fit to collected data for each state variable are utilized in a setup Monte Carlo simulation experiment configured to predict concrete production cycle lengths.

Keywords: Data, Analytics, Offsite, Concrete block, Production, Cycle Length, Monte Carlo Simulation

1. Introduction

Practitioners recommend that before the utilization of data in any data-driven analytics study such as simulation modeling, machine learning algorithms, or artificial intelligence algorithms, it is absolutely critical that you understand the data. Otherwise, one runs the risk of violating mathematical assumptions and drawing wrong conclusions when they throw the data into a black box and make deductions (Angela 2014). The best place to start gaining insights into the dataset is with descriptive statistics which can reveal a lot of interesting things without the need to perform complex calculations. Once this notion of performing frontend analytics is embraced by analysts, it then becomes necessary for them to also adopt a strategy that entails exploring each variable individually.

In situations that regression modeling is the main focus in performing analytics on a given dataset, it is mandatory to perform preliminary analytics that give insights into which variables could be redundant and need to be dropped moving forward into regression analysis. Correlation analysis is done on all possible variable pairs. Those that return high positive correlations (for the linear or non-linear, i.e., Spearman's and Pearson's respectively), reveal a potential for redundancy hence causing a double effect moving forward into regression analysis. If such a case arises, it is recommended to only carry forward one of the highly correlated variables. This is one of the benefits of including frontend descriptive analytics in any comprehensive, data-driven study.

Analytics performed at another level can be utilized to fit models that represent the dataset and the patterns that include within it. These could be fuzzy membership functions or probability distributions. In this paper, the focus was on the use of probability distributions for this purpose. It was demonstrated how probability distributions are fitted to data and how the appropriate distribution was picked, i.e. using different goodness of fit criteria. To demonstrate this, a case study on concrete block production was studied. This operation was abstracted, data collected about it and data models fitted to each of its state variables. Subsequently, the operation was emulated on a computer through Monte Carlo simulation for purposes of predicting the cycle length associated with producing four concrete block units.

2. Descriptive Statistics

Descriptive statistics are some of the front-end statistics that are computed in any analytics study. This is because

they allow the data modelers a chance to gain insights into underlying trends in that data that they will be processing. In specific cases, the descriptive statistics also provide guidance on how to improve the quality of the data, e.g. in case there may exist outliers.

2.1 Sum and Count

The size of any given dataset is represented by the count statistic. The count is equal to the total number of instances, tuples, etc., that there are in a dataset. This statistic is very important because it is used to verify the completeness of the data. The count is also used in the computation of other statistics such as the mean, standard deviation, skewness, kurtosis, etc. This statistic is often denoted by the letter "n". The sum is a statistic that at times is relevant and other times it's not relevant. When relevant, the sum is indicative of the total magnitude of the data instances within a dataset for a particular state variable. In other cases, it represents the total magnitude of data instances that represent different state variables for each unique tuple. A practical example of a situation in which the sum of data instances for different state variables would make sense is the evaluation of cycle length, also often referred to as cycle time. The sum statistic can give indirect insights into whether the magnitudes of the individual data instances are reasonable.

2.2 Measures of Central Tendency

Measures of central tendency are statistical parameters that are indicative of the distribution of data relative to the mean value of the instances within the dataset.

2.2.1 Mean

The mean of a dataset is the same as the average value which represents the value that is central to all instances within a dataset considering their magnitudes as their weights. As such, the mean value tends to lie closer to the values with a higher magnitude. The mean is different from the median, which finds the central value in the dataset when its instances are ranked in ascending order. The median does not consider the magnitude of the data instances but rather, their rank. The mathematical Equation used to obtain the mean value of a given data set is presented in Equation 1.

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{1}$$

2.2.2 Variance and Standard Deviation

Standard deviation indicates the extent to which instances within a given dataset are close to or dispersed from the mean value. Standard deviation is obtained through the variance which takes utilizes the square of the difference between instances and their mean value so as to avoid the negative deviations canceling out the positive deviations in cases that they are equal in magnitude. The standard deviation is obtained as the square root of variance. The mathematical equation used for this is presented in Equation 2.

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(x_i - \bar{x}\right)^2} \tag{2}$$

2.2.3 Median

Median is an interesting statistical measure because it not only falls under the category of measures of central tendency but also is within the category of quartiles. A description of this statistic will be presented in this sub-section, however, the mathematical formula utilized in its computation will be presented in a general form within the sub-section on quartiles. A median is that value that slices the dataset exactly into half when the instances in that dataset are arranged in ascending order. This measure does not make use of the magnitude of the data instances like the mean does, but rather utilizes the rank of the ordered instances. As such, there are rare instances in which median values are the same as mean values but they are often different. When used together, the mean and median values can provide valuable insights into the presence of outliers. It has been written in literature, that if the mean value and median value of a given dataset are significantly different, there are likely to be outliers within that dataset (Angela 2014).

2.2.4 Skewness

Skewness is an important measure in data science that informs about the distribution of data about the mean. Formal literature defines skewness as a measure of symmetry about the mean (Provide citations). Data may be said to have one of three types of skewness – no skew, positive skew, or negative skew. There exists a simple rule to tell the category of skewness for data that has been plotted graphically. Data is said to have a negative skew if its tail that is left of the mean is longer than the tail that is right of the mean. Data is said to have a positive skew if the tail to the right of the mean is longer than the tail to the left of the mean. Data that has no skew has the length of the left tail approximately equal to the tail that is right of the mean. This is shown in Figure 1.

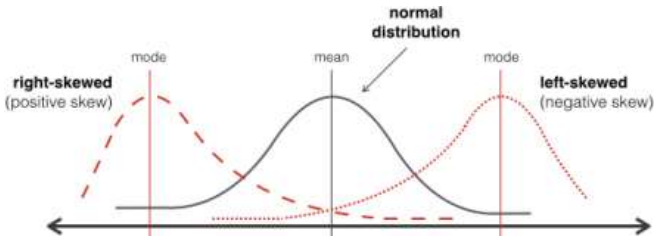


Figure 1. Graphical Representation (Shape) of Positive, No, and Negative Skew (Source: Angela 2014)

Skewness indicates where the bulk of the distribution/data is and a possible presence of outliers within the data. If present, particularly in skewed data, outliers are often in the opposite direction to where the bulk of the data is. The Mathematica formula for computing skewness is given in Equation 3.

$$Skewness\left(\gamma_3\right)=\frac{n}{(n-1)(n-2)}\sum_{i=1}^{n}\frac{\left(x_i-\bar{x}\right)^3}{s^3} \tag{3}$$

Values computed from this skewness equation can be zero, negative, or positive. A value of zero would imply that there is no skewness, i.e. the data is symmetrically distributed about its mean value. On the other hand, negative and positive values would imply that the data is skewed to the left and right respectively.

2.2.5 Kurtosis

Kurtosis is a Greek word that means “curved arching”. In probability and statistics, kurtosis is used as a measure of “tailedness” of a probability distribution. It describes the shape of the distribution’s tails in relation to its overall shape. It is expressed as the combined weight of a distribution’s tails relative to the weight of the center of the distribution (shown in the following mathematical expression).

$$Kurtosis=\frac{Weight\ of\ \ left\ tail+Weight\ of\ \ right\ tail}{Weight\ of\ \ the\ rest\ of\ \ the\ distribution} \tag{4}$$

It is common knowledge amongst data scientists that normal probability distributions are bell-shaped with tails/extremes that extend up to approximately three standard deviations away from the mean (above and below the mean). Kurtosis is a statistical measure whose magnitude strives to compare the distribution of data to the shape of a typical normal distribution. For example, high values of kurtosis are interpreted as data that is distributed in such a way that its tails are longer (also often termed as “heavier”) than those of a normal distribution. In other words, high kurtosis values are indicative of the presence of several extreme values that make the tails heavy. This translates into an overall unique shape of the distribution, i.e., extreme values can be thought of as stretching a distribution along its horizontal axis making the bulk of the data appear within a skinny, tall vertically narrow range. The weight of the tails is said to be heavier than that of the rest of the distribution and as such, the kurtosis is referred to as *leptokurtic*. On the other hand, small kurtosis values imply tails that are shorter than those of a comparable normal distribution. The fact that there are fewer values within the tails implies that the distribution is not stretched along its horizontal hence there are more values in the rest of the distribution over a wider range other than the tails resulting in a fatter shorter distribution shape. In this case, the weight of the tails is less than the weight of the rest of the distribution and as such the kurtosis is referred to as *platykurtic*. The last case is one in which the tails and rest of the distribution are identical to the bell-shaped normal probability distribution. Such as kurtosis is referred to as *mesokurtic*. Figure 2 is a visual representation of these types of kurtosis.

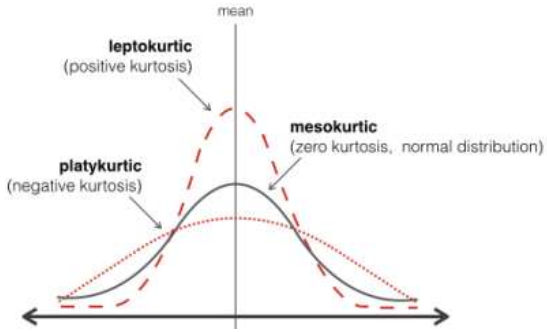


Figure 2. Graphical Representation (Shape) of Negative, Zero, and Positive Kurtosis (Source: Angela, 2014)

When assessed in isolation, values for this measure are referred to as kurtosis but when compared to the kurtosis of a normal distribution (which is a value of 3.0), the measure is referred to as excess kurtosis. The excess kurtosis value for a leptokurtic case is positive, that for a mesokurtic is zero while that for platykurtic is negative. The Mathematica Equation for excess kurtosis is presented in Equation 5.

$$Kurtosis\left(\gamma_4\right)=\left\{\frac{n(n+1)}{(n-1)(n-2)(n-3)}\sum_{i=1}^{n}\frac{\left(x_i-\bar{x}\right)^4}{s^4}\right\}-\frac{3(n-1)^2}{(n-2)(n-3)} \tag{5}$$

2.3 Partitioning Statistical Measures

There are a number of statistics that slice dataset into parts and are referred to here as "Partitioning Statistical Measures". Some of the measures that also belong to this category define the limits/boundaries for the data, i.e. minimum value and maximum value. The minimum value represents the cut-off at the lower boundary and is equal to the least value in the dataset. The maximum value represents the cut-off at the upper boundary and is equal to the greatest value in the dataset. The other partitioning measures have general names describing them, i.e. percentiles, deciles, and quartiles. Underneath those can be different instances of percentiles, deciles, and quartiles.

2.3.1 Percentiles, Deciles, and Quartiles

Percentiles, deciles, and quartiles are all statistics that were created to facilitate and equip data scientists with mechanisms of examining the fashion in which instances within a particular dataset are distributed. These statistics don't tell the analyst the number of instances that are present within a sub-domain of the dataset but rather report the magnitude of an instance at a section where the dataset is sliced. Such information may be extremely useful in certain types of applications. The three statistics exist to provide different resolution options at which datasets can be scanned. Percentiles provide a $1/100^{th}$ resolution while deciles provide a $1/10^{th}$ resolution. Quartiles, on the other hand, provide ¼ resolutions. The type of application domain, together with the preferences of data scientist, influence the choice of resolution. However, quartiles seem to have been the most commonly utilized in both academia and practice. According to Mendenhall & Sincich (1995), the general formula used to compute the position index (i) for the value sought is presented in Equation 6.

$$i=\frac{j(n+1)}{m} \tag{6}$$

In this formula, n represents the total number of data instances, also referred to as the count for the dataset. Then j represents the statistic-based index that is of interest to the analyst. The parameter m represents the resolution at which the data is to be split and directly relates to whether you are finding quartiles, deciles, or percentiles. The result of this formula is an index i that corresponds to the ranking of the value that is being sought from the dataset. The final value is obtained from interpolation or extrapolation using the rank index i that was computed.

2.3.2 Box-Whisker Plot

A box-whisker plot is a graphical tool used to visually display the distribution of the instances within a given dataset. It utilizes the three quartile values (i.e. first quartile, second quartile, and third quartile) to split the data into four equal portions (Statistics Canada, 2017). The whiskers go from each quartile to the minimum or maximum (See Figure 3).

Figure 3. Typical Schematic Layout of a Box-Whisker Plot (Statistics Canada 2017)

In this Figure, the "lowest observation" and "highest observation" labels correspond to the minimum and maximum values in the dataset respectively. The difference between the upper quartile (Q_3) and lower quartile (Q_1) is referred to as the inter-quartile range. The inter-quartile range value is utilized in certain criteria for examining the presence of outliers in a given dataset. Box-whisker plots give insights into the extent to which a given dataset varies. The larger the box, the higher the variability in the data instances. It is also indicative of the magnitude of the values.

2.3.2 Histogram/Probability Distribution Function and Cumulative Distribution Function

Probability density functions and cumulative density functions can be expressed both graphically and analytically as an equation. However, preference is given to graphical representation because it makes it easy to make inferences from visual inspection of their shapes. In fact, for empirical distributions, graphical methods may be the only way to represent PDFs and CDFs. PDFs serve different purposes from CDFs. PDFs are mainly used to gain insights into the distribution of the data, i.e. the extent of skewness. CDFs, on the other hand, can be a fast, and easy way to determine the probability with which certain values or range of values are likely to occur.

PDFs can be represented in one of two forms based on the fashion in which the likelihood is computed. The first of these types makes use of the relative frequency of the data as its likelihood value. The other PDF type utilizes a ratio of the relative frequency to the bin width. In the first PDF type, the sum of the relative frequency values is always equal to one. In the second type of PDF, the total area for the histogram sums to a value of one. When drawing histograms, one critical variable that often affects the shape and values generated, is the bin width. Several mathematical models have been proposed for calculating the bin width. They include – Sturge's rule (Sturge 1926), Doane's rule (Doane 1976), Rice's, and Freedman-Diaconis's rule (Hyndman 1995; Legg et al 2013; Doane 1976). These Equations are each summarized next in their respective order.

$$1+3.322\log n \tag{7}$$

$$\log_2 n+1+\log_2\left(1+\frac{\sum_{i=1}^{n}\left(x_i-\bar{x}\right)^3}{\left[\sum_{i=1}^{n}\left(x_i-\bar{x}\right)^2\right]^{3/2}}\sqrt{\frac{(n+1)(n+3)}{6(n-2)}}\right) \tag{8}$$

$$3.49\sigma n^{-1/3} \tag{9}$$

$$2*\sqrt[3]{n} \tag{10}$$

$$2(IQR)*n^{-1/3} \tag{11}$$

Once the number of bins has been computed, the bin width can be calculated using Equation 12. Once this has been done, the bins across the range of the data can be determined along with the number of data instances that fall within each bin, i.e., the frequency. Then one of the types of histograms can be constructed.

$$Bin\ width=\frac{Maximum\ value-Minimum\ value}{Number\ of\ bins} \tag{12}$$

A cumulative distribution function is an accumulation of a probability density function. It represents the probability with which a specific value or more for a state variable is likely to be achieved. When represented graphically, the y-axis always has a minimum value of zero and a maximum value of one. The process of generating an empirical CDF from data requires first ranking the data instances in ascending order, then finding the ratio of each instance's rank to the count. For theoretical probability distributions, an analytical equation can be obtained for the CDF by integrating the PDF equation.

2.4 Fitting Methods for Probability Distributions

Probability distributions can be fitted in one of two ways: (1) using empirical data, and (2) using expert knowledge. The process of fitting probability distributions, whether continuous or discrete in nature, entails determining the values of the parameters for that probability distribution. When making use of empirical data as input into the distribution fitting process, several fitting methods can be utilized. The most popular of these include: (1) moment matching method, (2) maximum likelihood method, and (3) least-squares method. Of these three methods, the least-squares method is the most accurate and makes use of the output of the moment matching and maximum likelihood methods as its input. Most probability distribution fitting software provide for the three distribution fitting methods, for example, easy fit, @Risk, Crystal Ball, etc. Distribution fitting follows data collection and cleaning processes in cases in which empirical data is used in the fitting process.

2.5 Testing a Probability Distribution Fit

The Jargon used to refer to testing how well a fitted distribution is, is referred to as a "Goodness of Fit Test". The goodness of fitting process just follows any probability distribution fitting process that is based on empirical data. It is the criteria by which the distributions are ranked. It should be noted that any distribution can be fitted to a given dataset because it is a matter of determining its parameter values using the dataset. However, if several distributions are fitted to a dataset, there will be some that fit better than others. The fitted distribution represents the population model to which the domain of all possible values for that state variable belong or are drawn. Different likelihoods can be computed that the empirical data sets came from the various probability distribution population models. These can then be used to rank the quality of fit for all the fitted distributions. Examples of the goodness of fit ranking methods include, (1) visual inspection of fitted distribution pdf to the empirical data histogram, (2) Chi-squared method, and (3) Kolmogorov-Smirnov method. The ultimate preference of the distribution is based on the modeler's intended use of the distribution and how well the distribution ranked based on the goodness of fit criteria.

2.6 Data Usage

Practitioners, researchers, and other analysts within the modeling domain, regard data as the lifeblood of numeric computing and modeling. This is attributed to the fact that it is crucial for all the stages involved in the modeling process (See Figure 4). Prior to its use within models, data is transformed into different forms depending on the type of modeling that is to be performed. For example, probability distributions may be fitted, fuzzy membership functions defined, fuzzy rules formulated. All this can only commence based on the data collected directly from the field/lab or domain experts. Such data should be collected in the right quantity and should undergo pre-processing (e.g. outlier identification and jack-knifing) to ensure that the quality is desirable. It is also necessary to gain insights into the nature of the data – its distribution and any obvious patterns that could exist. Acquiring such an understanding is extremely vital, particularly when preparing data for purposes of performing Artificial Intelligence/Machine Learning (AI/ML) analytics where patterns – simple or complex, matter a lot. This is summarized schematically in Figure 4.

This paper places an emphasis on pre-processing operations that need to be taken prior to transforming data into forms that can be utilized in computer-based numeric modeling. There is a significant amount of computational work that has to be done at the tail-end of any classical modeling study. This is often technically referred to as "output modeling".

3. Case Study – Concrete Block Production

3.1 Prefabrication

Prefabrication is a term used within the construction domain to refer to materials resources utilized in construction production processes that typically take place onsite. However, this category of materials, referred to as "prefabricated material", are produced offsite and transported to the construction site for utilization.

Offsite production is often performed either within a closed facility such as a structural steel or pipe spool fabrication shop or within a protected yard. In an offsite setting, the environment is not as constrained from a space and layout perspective as construction sites are. This gives the construction engineers vital control and leverage that they often utilize to boost their quality, safety, and productivity. As such, the offsite production of construction components has been widely adopted by the industry and has blossomed. This has translated into

modularization of facility components through all project phases, i.e., conceptualization, design, construction (fabrication, and erection), operation, and maintenance. The modularization concepts and practice have matured over the years transitioning from first through to fourth generation modularization. Modularization and offsite construction has not been limited to specific construction materials but rather has been embraced across the spectrum of materials used in the production of facility or infrastructure components such as steel, timber, asphalt concrete, cement concrete, etc.

In this paper, a cement concrete application is chosen and discussed for purposes of showcasing the application of the data science concepts presented. The operation selected was one that involves the production of plain light cement concrete blocks. The choice of this operation was guided by the fact that it is cyclic in nature and multi-tasked but not in a convoluted fashion.

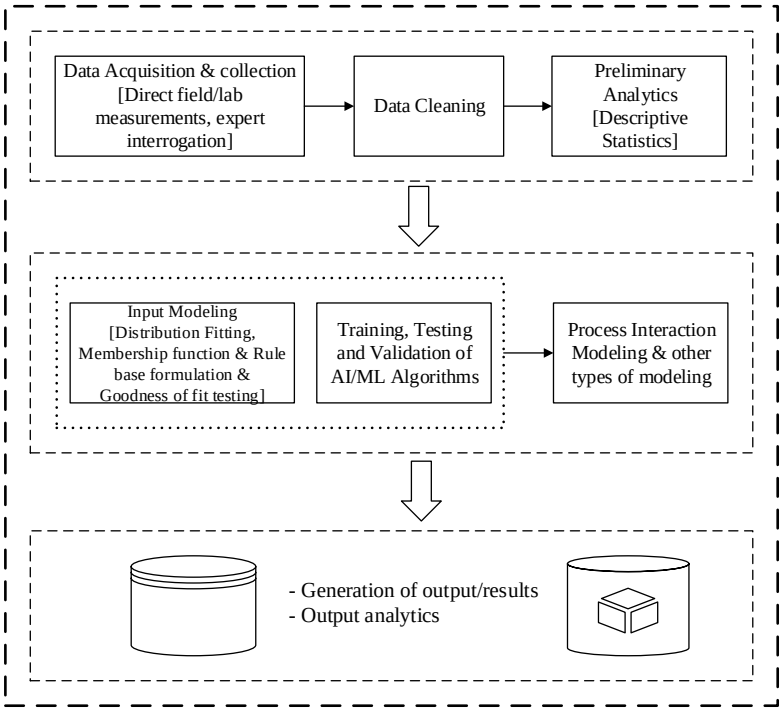


Figure 4. Schematic layout showing the workflow for fitting data models and their use in numeric models

3.2 Methodology

3.2.1 Process Abstraction and Data Collection

In order to successfully create the formalism and precise abstraction of a real-world operation and implement that successfully on a computer in a numeric fashion, there is a need to abstract the logical sequence in which the operation works. Also, there is a need to define data models that emulate the passing of time when processes associated with the operation are being executed. This systematic process of formalizing real-world operations was ratified and applied in the abstraction of the concrete brick/block making operation. The logical sequence in which processes are executed was established through observations made within the fabrication yard. Details abstracted were summarized in a schematic layout. These schematic layouts were validated by domain experts prior to finalization and subsequent use. The resources utilized within the production process were also documented after careful observations from the perspective of each activity.

Strategic sampling was utilized when identifying locations for studying concrete brick/block production operations. The data tracked in this production process was mainly the duration required to complete each activity. The duration data for the majority of the tasks were obtained through direct measurement. However, those that involved travel over variable distances, the duration variable was replaced with travel speed. The duration was measured using a stop clock built within a mobile phone and values recorded within a journal. For tasks that involved travel over variable distances, travel times and travel distances were measured and their corresponding travel speed values computed. Data was collected at different times of day and days of the week.

All this data was subsequently transferred from the journal into an MS. Excel file. Strategies often utilized in performing effective time studies were also applied in this data collection process. The following sub-sections present an overview of the abstracted operation and data on the duration and speeds with which certain critical tasks in this operation are executed.

3.2.2 Concrete Block Production Operation

Cement concrete blocks are often produced offsite because they are easy to transport and assemble into walling systems once delivered onsite. These blocks are produced from light concrete, a concrete that is made using fine aggregate, cement, and water. Often a block/brick making machines are used in the production process so as to expedite production. The majority of these machines are pneumatic and powered by electricity. Their power source varies from conventional hydro-electricity to diesel/gasoline-driven power generators. These machines come in different sizes and setups. There is a type that has a mixing chamber attached to the pneumatic component. Others don't come with an attached mixing chamber. Mixing is done separately and the mixture is loaded into the machine, which is predominately a pneumatic compacting chamber. The case study operation presented in this study made use of a block producing machine without an attached mixing component. The process utilized in producing concrete blocks on that offsite production facility is described next.

Raw ingredients are mixed outside the block making machine. This mixing is typically done either by hand or by the use of a mixing machine. In the operation that was being studied, blending was done by machine. After ingredients (cement and fine aggregate were fixed), water was added to make the mixture plastic. Ingredients are blended to generate a coherent and consistent near-dry mix. This plastic mixture was then loaded into the block making machine. The near-dry mix that is loaded into the block making machine contains a lot of air and moisture that are not good for the final block product and at this point also has no shape. The block making machine has mould components that house the mix that is loaded into the machine. The mixture that was placed into the moulds was then pneumatically compacted so that air in the voids is expelled and the contents of the mould achieve the desired size and shape of the final block product. Compaction also translates into better aggregate interlock hence superior strength and durability. The compacted material is referred to as a concrete block from this point onwards. Each concrete block that was produced had the following dimensions: *200mmX200mmX400mm*. The described processes are summarized in the process box labeled "A" in Figure 5.

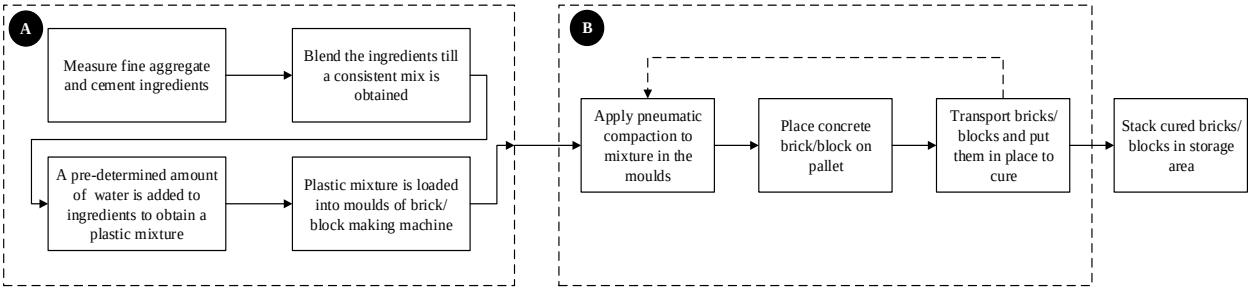


Figure 5: Schematic Diagram Summarizing the Concrete Block Making Process

The quantity of material feed into a mixing machine was sufficient to produce 24 concrete blocks. However, the concrete block making machine was restricted to a capacity of 4 blocks per production cycle because it contains only 4 models. As such, subsequent cycles for mixing of raw ingredients were started only after the block making machine has completed its 6 cycles required to empty a fully loaded mixing machine. Concrete blocks are then placed onto a pallet that is transported by a buggy to a designated laydown area where they are placed individually onto the ground surface that is covered with a polythene sheet. The average distance between the block making machine and the curing laydown area was approximately 30m. The blocks are left out in the open to cure, i.e. dry and develop their design strength. This is shown in the process box labeled "B" in Figure 5. However, overhead protection from rain is provided whenever necessary because uncontrolled exposure to excessive moisture can adversely affect the curing process. After curing is completed and the desired strength is achieved, blocks are moved to a storage area where they are stacked over each other in order to optimize space. When the time comes, the blocks are loaded onto a truck and transported to a construction site where they are utilized in the building process.

4. Analysis, Results, and Discussion

4.1 Data Export to Mathematica

Data collected on field activities was stored in an MS. Excel file with values for each state variable placed in a column and state variable columns arranged side-by-side. This data is exported to another environment, in this case, Mathematica, for purposes of performing analytics on it. This is done to utilize the extensive and robust mathematics library built into Mathematica when performing computations hence automating the entire analysis process. Data from the excel file was imported using the following Mathematica code snippet and put in unique lists for each state variable. Note that an escape sequence was used in the file path definition when calling the "Import" function. After the data was successfully exported into Mathematica, other operations were carried out on it.

```
Clear[Data, MachineLoadingTime, MachineProcessingTime, PalletLoadingTime, HaulTime, ReturnTime, row];
MachineLoadingTime = {};
MachineProcessingTime = {};
PalletLoadingTime = {};
HaulTime = {};
ReturnTime = {};
row = 2;
Data =
  Import[
   "C:\\Users\\ekyalimp\\Desktop\\[1]. Journal Paper - Input Simulation and Output Analysis Fall
     2019\\Concrete Block Prefabrication Data.xlsx"];
For[
  row = 2,
  row ≤ Length[Data[[1]]],
  row++,
  MachineLoadingTime = AppendTo[MachineLoadingTime, Data[[1]][[row]][[3]]];
  MachineProcessingTime = AppendTo[MachineProcessingTime, Data[[1]][[row]][[4]]];
  PalletLoadingTime = AppendTo[PalletLoadingTime, Data[[1]][[row]][[5]]];
  HaulTime = AppendTo[HaulTime, Data[[1]][[row]][[6]]];
  ReturnTime = AppendTo[ReturnTime, Data[[1]][[row]][[7]]];
 ];
```

Figure 6. Code Snippet for Importing Data into Mathematica

4.2 Histograms

Histograms are useful statistical graphics that give analysts insights into the distribution of the data. There is an in-built "Histogram" function within Mathematica that was utilized in plotting the histograms for each of the five state variables within our concrete block production case study. The following Mathematica code snippet was used to automate this process.

```
Histogram[MachineLoadingTime, ChartElementFunction → "Rectangle", ChartStyle → Yellow,
 AxesLabel → {"Machine Loading Time [Mins]", "Likelihood"}, PlotLabel → "Machine Loading Time"]
Histogram[MachineProcessingTime, ChartElementFunction → "Rectangle", ChartStyle → Gray,
 AxesLabel → {"Machine Processing Time [Mins]", "Likelihood"}, PlotLabel → "Machine Processing Time"]
Histogram[PalletLoadingTime, ChartElementFunction → "Rectangle",
 AxesLabel → {"Pallet Loading Time [Mins]", "Likelihood"}, PlotLabel → "Pallet Loading Time"]
Histogram[HaulTime, ChartElementFunction → "Rectangle", ChartStyle → Blue,
 AxesLabel → {"Haul Time [Mins]", "Likelihood"}, PlotLabel → "Haul Time"]
Histogram[ReturnTime, ChartElementFunction → "Rectangle", ChartStyle → Red,
 AxesLabel → {"Return Time [Mins]", "Likelihood"}, PlotLabel → "Return Time"]
```

Figure 7. Code Snippet for Plotting Histograms for Each State Variable

The following histograms (in Figure 8) were generated when the code snippet was executed. The "Likelihood" type histogram was generated in each case. There is evidence of skewness in some of the histograms that were generated for each state variable in the concrete block fabrication process.

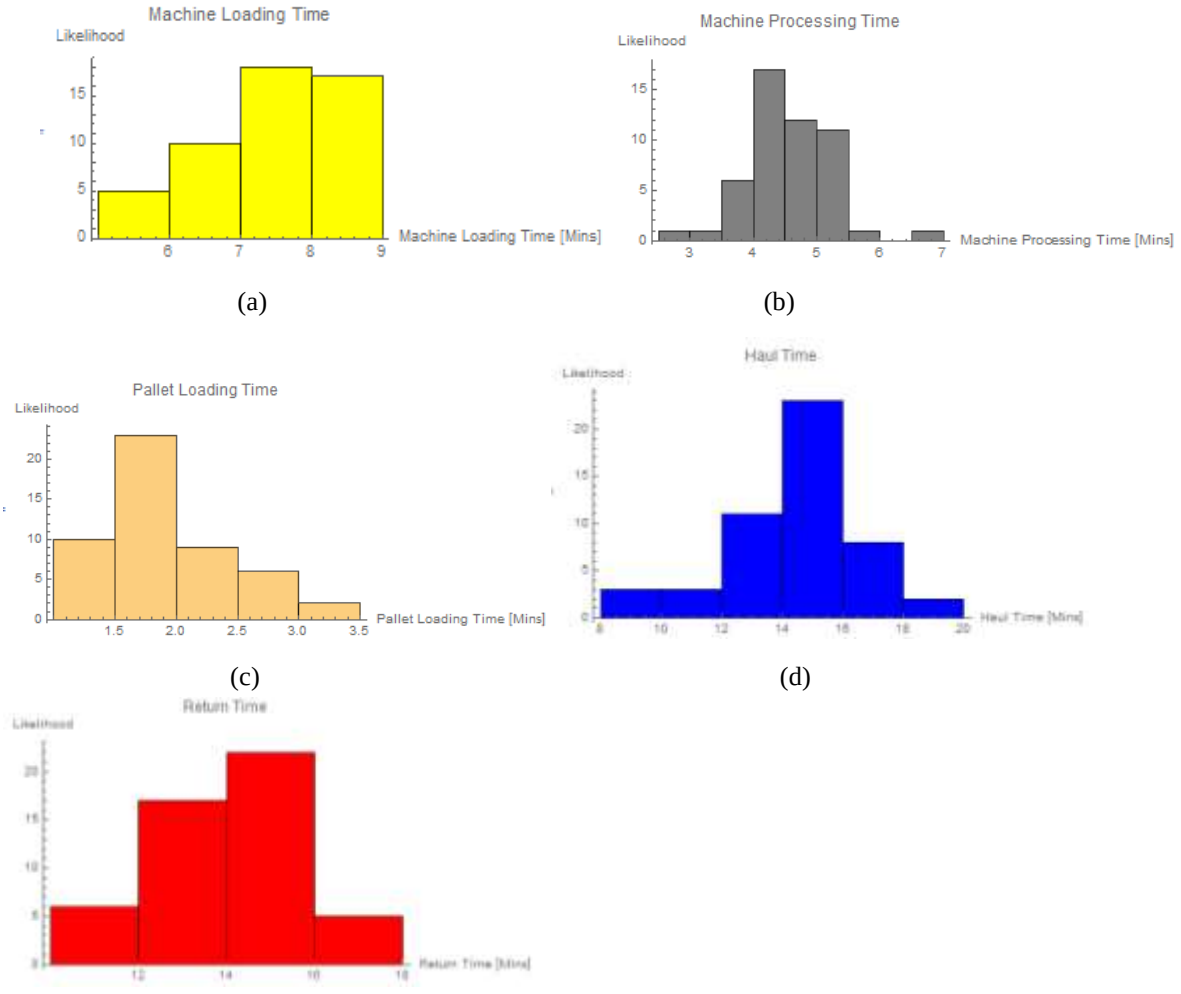


Figure 8. Histograms Generated from (a) MachineLoading, (b) MachineProcessing, (c) PalletLoading, (d) Haul, and (e) Return Datasets

The histograms for the "MachineProcessing" and "Haul" state variables have tails that are fairly long and heavy. This implies that the data points collected in the field had a high presence of extreme values, i.e. very small and very large values. The histograms for the "MachineLoading", "PalletLoading", and "Return" don't seem to have tails which imply that there were no extreme data points collected from the field.

The histograms for "MachineProcessing", and "Return" exhibit near-symmetry, i.e., minimal skewness. Histograms for "MachineLoading", and "Haul" are skewed to the left, i.e. the bulk of their data points are to the right. The histogram for "PalletLoading" seems to be skewed to the right, i.e. the majority of its data points are to the left side.

4.3 Box-Whisker Plot(s)

Box-Whisker plots are important visual ways of representing the dispersion of data points for a given state variable. For this study, these were generated in Mathematica. The charts obtained are shown in Figure 9. The "BoxWhiskerChart" in-built function in Mathematica was used to generate this chart.

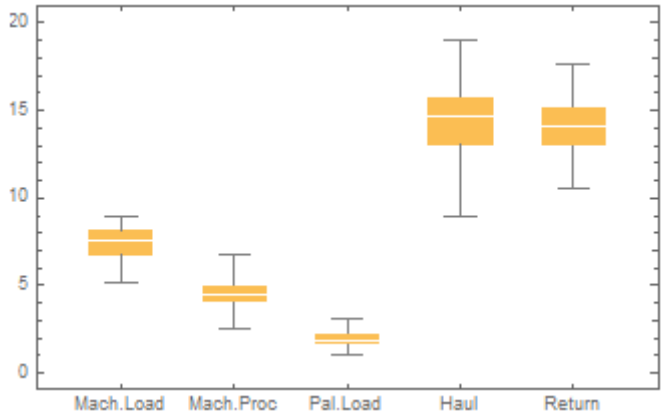


Figure 9. Box-Whisker Plots for Data on each of the State Variables

The “Machine Loading”, “Haul” and “Return” state variables show the greatest variation, with the “Haul” and Return” variables having the largest variation. It is important to note that the state variable referred to here as “Machine Loading” involves the activities for preparing the mix for 4 blocks and loading that mix into the 4 moulds of the block making machine. The “Palette Loading” state variable has the least variation and has the least values compared to all the other state variables. The “Machine Processing” state variable exhibits moderate variability.

The high uncertainties in the “Haul” and “Return” state variables are inherent to the fact that it is a travel activity and one that involves movement that is manually powered through a significant distance, i.e. 30m. The effort expended by the workers pulling or pushing the loaded or unloaded cart varies depending on how their fatigue levels, time of day, etc. The fact that the duration for these two activities are almost twice as high as those of other activities implies that they could result in waiting and delay hence bottlenecking the production cycle. Consequently, these two activities should be the first to be looked at if any improvements to the production of the operation are to be considered. Addition of resources, i.e. another cart with workers could lessen the delay of waiting by the block producing machine hence shortening the overall cycle length for the block production.

The “MachineProcessing” state variable is a tricky one in this case study. Although the Box-Whisker plot shows that it has moderate variability and relatively short durations compared to most of the other activities (except pallet loading), it is a critical activity for the entire process because it is equipment intensive. Consequently, decision makers managing and supervising such an operation need to pay attention to it and dedicate the best equipment that they possibly can in order to maximize the uptime and avoid bottlenecking the operation. It is for such scenarios that data analysts are advised and cautioned to interpret their results in relation to the contexts they are studying.

4.4 Basic and Other Statistics

Results generated by running the Mathematica code snippet in Figure 10 are summarized in Table 1. It is a combination of basic statistics and more advanced ones. These statistics give valuable conclusive insights into the nature and quality of the data that was collected for each state variable.

The “MachineLoading”, “Haul”, and “Return” datasets generated negative skewness values. The values of skewness for these state variables are relatively small which implies that their data was slightly skewed to the left. The “Haul” time had the highest magnitude while the “Return” time had the least magnitude of the three hence the “Haul” state variable has the highest skew to the left while the “Return” state variable has the least skewed to the left of the three state variables.

The “MachineProcessing” and “PalletLoading” state variables generated positive skewness values. Their values are both relatively small but with that of the “PalletLoading” being the larger of the two. This means that both these state variables are skewed to the right but with the “PalletLoading” exhibiting more of that than the “MachineProcessing”.


```
Print["Count = ", Count[ReturnTime, Except[0.0]]]
Print["Min = ", Min[ReturnTime]]
Print["Max = ", Max[ReturnTime]]
Print["Mean = ", Mean[ReturnTime]]
Print["Variance = ", Variance[ReturnTime]]
Print["Skewness = ", Skewness[ReturnTime]]
Print["Kurtosis = ", Kurtosis[ReturnTime]]
Print["Quartile 1 = ", Quartiles[ReturnTime][[1]]]
Print["Quartile 2 = ", Quartiles[ReturnTime][[2]]]
Print["Quartile 3 = ", Quartiles[ReturnTime][[3]]]
Print["Inter-Quartile Range = ", Quartiles[ReturnTime][[3]] - Quartiles[ReturnTime][[1]]]
HOLCutoff = Max[ReturnTime] + (1.5 * (Quartiles[ReturnTime][[3]] - Quartiles[ReturnTime][[1]]))
LOLCutoff = Min[ReturnTime] - (1.5 * (Quartiles[ReturnTime][[3]] - Quartiles[ReturnTime][[1]]))
```

Figure 10. Mathematica code snippet for generating basic statistics

Table 1. Basic statistics for the state variables in the concrete block production operation

Parameter	State Variable Time [Minutes]				
	MachineLoading	MachineProcessing	PalletLoading	Haul	Return
Count	50	50	50	50	50
Minimum	5.21	2.57	1.07	8.89	10.52
Maximum	8.90	6.81	3.14	19.01	17.64
Mean	7.45	4.53	1.95	14.28	14.13
Variance	0.88	0.46	0.29	4.98	2.85
Skewness	-0.49	0.22	0.47	-0.62	-0.18
Kurtosis	2.56	5.04	2.67	3.20	2.74
Quartile 1	6.76	4.06	1.64	13.03	13.03
Quartile 2	7.62	4.50	1.88	14.67	14.12
Quartile 3	8.17	5.01	2.26	15.78	15.23
Inter-quartile range	1.41	0.95	0.62	2.75	2.2
Lower Outlier Threshold	3.10	1.15	0.14	4.86	7.22
Upper Outlier Threshold	11.02	8.24	4.07	23.14	20.94
# of Outliers	0	0	0	0	0

Only two state variables, “MachineProcessing” and “Haul” have kurtosis values greater than 3.0 hence leptokurtic. This implies that their tails are heavier than that of a normal distribution and also indicate a high presence of extreme values in the data points collected from the field. This was evident in the shape of the histograms that were plotted. Consequently, there is a higher probability of sampling high and small values from the probability distributions of these state variables.

The “MachineLoading”, “PalletLoading”, and “Return” state variables have kurtosis values less than 3.0 meaning that they are platykurtic and have tails that are lighter relative to a normal distribution of same mean and standard deviation. It also implies that there was no presence of extreme values in the data points collected in the field. The probability of sampling small or large values from their probability distributions is also extremely low.

The quality of the datasets for each respective state variable is largely determined by the presence or lack of outliers. In this case study, near outlier cut-off thresholds were used. Results obtained indicate no presence of outliers in any of the state variables hence implying that the quality of the datasets was good and could be used in the subsequent analysis without a need for improvement.

5. Input Modeling

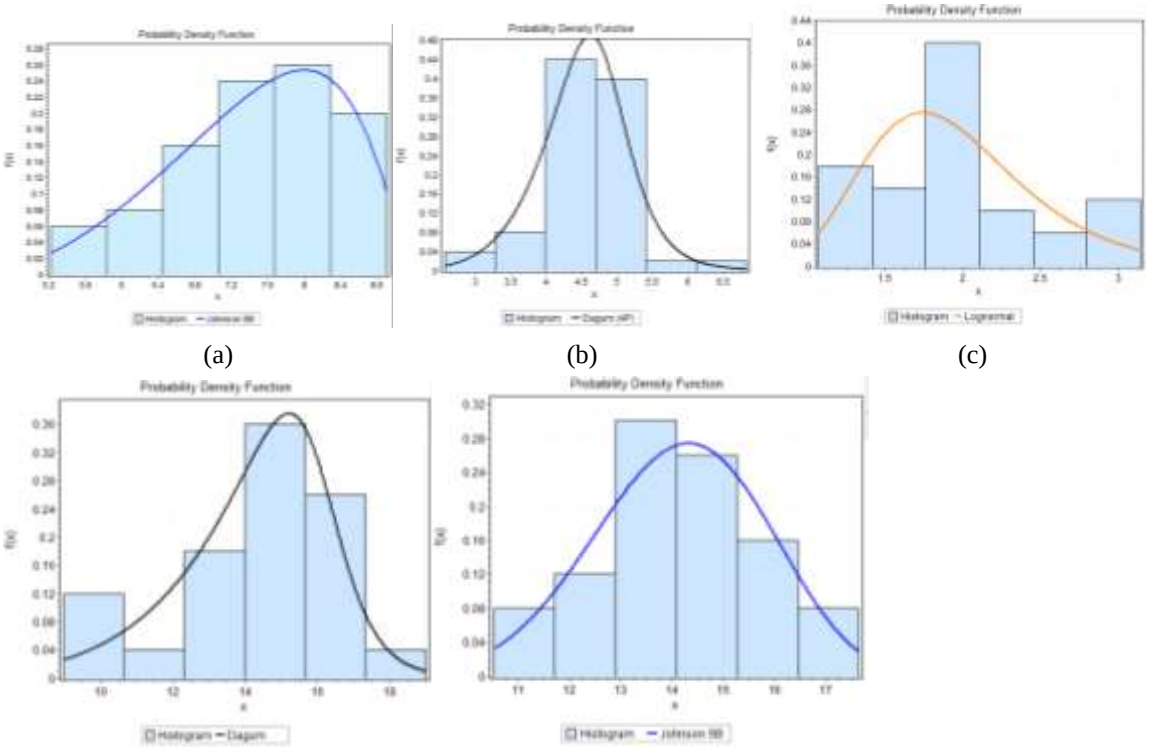
Input modeling may be expert-driven or data-driven and can be used to transform data into a model format that can subsequently be entered as input into a higher-level model. These low-level data models could be probability distributions, fuzzy membership functions, fuzzy rules, etc. In this study, we will be demonstrating the process of fitting probability distributions which can subsequently be utilized in simulation experiments.

EasyFit software was used to fit probability distributions to data for each respective variable. The least-squares distribution fitting method was used to obtain the parameters for each distribution for the respective state variables. The same goodness of fit methods were used to obtain the best-fitted distributions for each state variable. "1" stands for the K-S test, "2" for the Anderson-Darling test, and "3" for the Chi-Square test.

Table 2. Input data models for the state variables in the concrete block production operation

State Variable	1 [KS]	2 [AD]	3 [Chi]
MachineLoading	JohnsonSB[-0.82,1.14, 3.96, 5.38]	Triangular[4.92,8.17,9.17]	Weibull[10.98,8.78,-0.93]
MachineProcessing	Dagum[0.46,14.02,3.97]	Gamma[150.40,0.05,-3.74]	Beta[790.93,1185.7,-19.81,41.03]
PalletLoading	Lognormal[0.63,0.28]	Log-Logistic[5.93,1.85]	Cauchy[0.26,1.88]
Haul	Dagum[0.26,25.22,16.19]	Triangular[8.12,15.42,19.34]	Beta[4050.50,15.06,-2270.8,22.78]
Return	JohnsonSB[-1.11,2.81,20.27,2.08]	Weibull[4.18,6.78,7.97]	Lognormal[2.64,0.12]

The graphical plot of the fitted probability distributions that ranked first for each of the state variables were plotted in Figure 11.



(a) (b) (c)

(d) (e)

Figure 11. Fitted probability distributions for (a) MachineLoading, (b) MachineProcessing, (c) PalletLoading, (d) Haul, and (e) Return state variables

6. Monte-Carlo Analytics

In order to generate an authentic result on which output analysis could be show-cased, a Monte Carlo simulation

was performed using the five state variables to obtain the cycle length for producing several concrete blocks off-site. A code snippet was written in Mathematica to automate the simulation process (See Figure 12).

```
Clear[TotalSimulationIterations, CycleLengths, SeedValue, CycleLength];
TotalSimulationIterations = 5000;
CycleLengths = {};
CycleLength = 0.0;
SeedValue = 7808849308;
SeedRandom[SeedValue];
Clear[MachineLoadingDistribution, MachineProcessingDistribution, PalletLoadingDistribution, HaulDistribution,
  ReturnDistribution, MachineLoadingRandomVariate, MachineProcessingRandomVariate, PalletLoadingRandomVariate,
  HaulRandomVariate, ReturnRandomVariate];
MachineLoadingDistribution = JohnsonDistribution["SB", -0.82, 1.14, 3.96, 5.38];
MachineProcessingDistribution = DagumDistribution[0.46, 14.02, 3.97];
PalletLoadingDistribution = LogNormalDistribution[0.63, 0.28];
HaulDistribution = DagumDistribution[0.26, 25.22, 16.19];
ReturnDistribution = JohnsonDistribution["SB", -1.11, 2.81, 2.08, 20.27];
MachineLoadingRandomVariate = 0.0;
MachineProcessingRandomVariate = 0.0;
PalletLoadingRandomVariate = 0.0;
HaulRandomVariate = 0.0;
ReturnRandomVariate = 0.0;
For[
  i = 1,
  i ≤ TotalSimulationIterations,
  i++,
  MachineLoadingRandomVariate = RandomVariate[MachineLoadingDistribution];
  MachineProcessingRandomVariate = RandomVariate[MachineProcessingDistribution];
  PalletLoadingRandomVariate = RandomVariate[PalletLoadingDistribution];
  HaulRandomVariate = RandomVariate[HaulDistribution];
  ReturnRandomVariate = RandomVariate[ReturnDistribution];
  CycleLength = MachineLoadingRandomVariate + MachineProcessingRandomVariate + PalletLoadingRandomVariate +
    HaulRandomVariate + ReturnRandomVariate;
  CycleLengths = AppendTo[CycleLengths, CycleLength];
 ];
Print["Mean Cycle Length = ", Mean[CycleLengths]];
Print["Standard Deviation Cycle Length = ", StandardDeviation[CycleLengths]];
Plot[PDF[NormalDistribution[Mean[CycleLengths], StandardDeviation[CycleLengths]], x], {x, 0, 100},
 Filling → Axis, PlotRange → All]
```

Figure 12. Mathematica code snippet for the Monte-Carlo simulation experiment

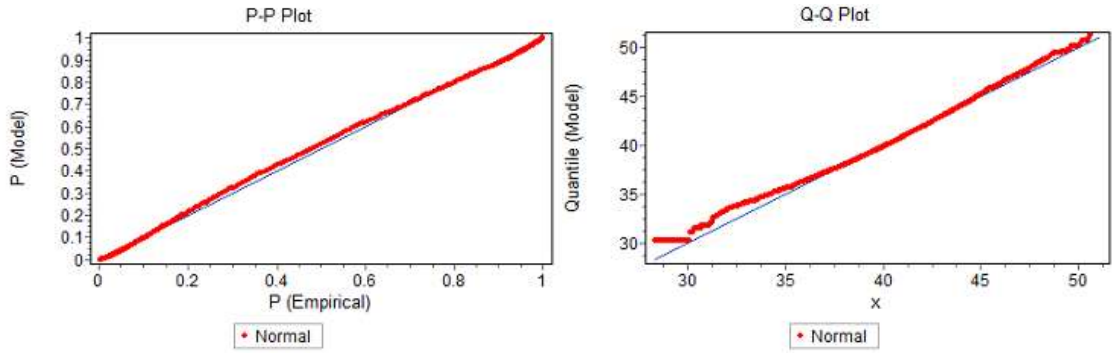
The configurations for the simulation experiment and basic statistics of the results obtained are summarized in Table 3.

Table 3. Simulation experiment configuration and results

#	Parameter	Value
1	Simulation Seed	7808849308
2	Total # of Iterations	5, 000
3	Mean Value	41.40
4	Variance	9.12
5	Standard Deviation	3.02

7. Output Analysis

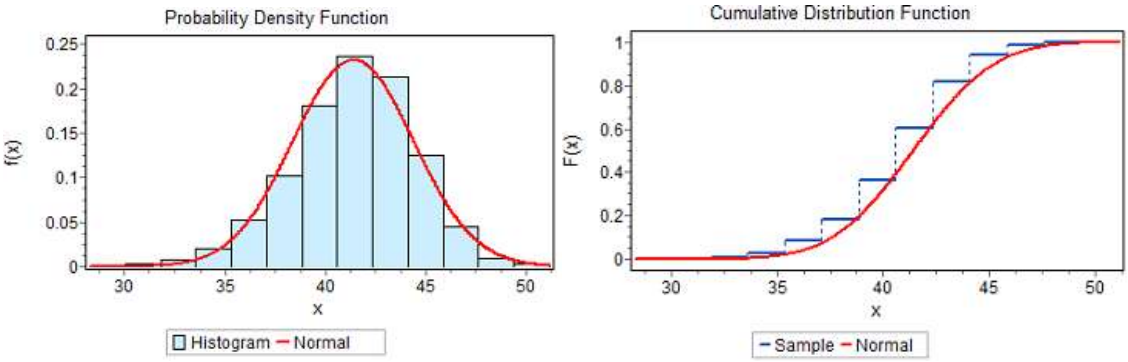
Monte Carlo simulation experiments conducted in an orderly fashion typically generate output that closely or precisely follows the normal probability distribution, i.e. is consistent with the central limit theorem. In this study, cycle length for the production of concrete blocks in an offsite location was computed using Monte Carlo simulation. The first task in any output analysis would typically involve testing for normality to confirm this. The easiest way to do this is by generating the P-P and Q-Q graphical plots. This was done using the EasyFit software and the results obtained presented in Figure 13.



(a) (b)

Figure 13. Graphical Plots of (a) P-P and (b) Q-Q for the Cycle Length in the Concrete Block Production Operation

Both plots indicate that the results generated from the simulation closely follow the normal distribution. Parameters for the normal distribution were obtained from results, i.e. the mean and standard deviation. The probability density function and cumulative density function for the fitted normal probability distribution were overlaying those of the empirical distribution for the empirically generated data. The plots generated are shown in Figure 14.



(a) (b)

Figure 14. Graphical Plot of (a) the PDF and (b) the CDF of the Theoretical Fitted Normal Probability Distribution

The probability density function primarily gives insights into the distribution of the values in a dataset and provides a visual means of assessing the appropriateness of fit for a particular theoretical probability distribution. Aside from that, it does not have much use in practice. On the other hand, the cumulative distribution function has several uses. It can be used to quickly obtain quantile values, probabilities associated with the state variable taking on certain values. For example, the probability of producing a pallet full of concrete blocks within a certain time of less can now easily be determined directly from the CDF graphs without the need for complex computations.

Other analyses that can be done include obtaining the confidence intervals for certain key statistics, quantiles, and probabilities. Examples of how this can be done for the mean and variance of the data generated in the case study are presented. A 5% significance level is assumed and the total number of observations is equated to the total number of simulation iterations performed, i.e. 5,000. The calculations for the interval for the mean value are presented next.

$$Confidence\ \text{int}erval(\mu) = \bar{X} \pm Z_{\alpha}\frac{S}{\sqrt{n}}$$

$$Confidence\ \text{int}erval(\mu) = 41.40 \pm 1.96\frac{3.02}{\sqrt{5000}}$$

$$Confidence\ \text{int}erval(\mu) = 41.40 \pm 0.0837$$

$$Confidence\ \text{int}erval(\mu) = [41.32, 41.48]$$

The computations for the confidence interval of the variance are slightly different. They are based on the Chi-Square probability distribution rather than the normal distribution. This distribution is chosen because when (the sample variance is) corrected to the population variance, the ratio of the corrected variance to the true population variance ($[n-1]s^2/\sigma^2$), is Chi-square distributed to n-1 degrees of freedom. The calculations are presented next.

$$\frac{(n-1)S^2}{\chi^2_{\alpha/2,n-1}} \le Confidence\ \text{int}erval\left(\sigma^2\right) \le \frac{(n-1)S^2}{\chi^2_{1-\alpha/2,n-1}}$$

$$\frac{(5000-1)3.02^2}{\chi^2_{0.025,4999}} \le Confidence\ \text{int}erval\left(\sigma^2\right) \le \frac{(5000-1)3.02^2}{\chi^2_{0.975,4999}}$$

$$\frac{(5000-1)3.02^2}{5196.86} \le Confidence\ \text{int}erval\left(\sigma^2\right) \le \frac{(5000-1)3.02^2}{4804.92}$$

$$8.77 \le Confidence\ \text{int}erval\left(\sigma^2\right) \le 9.49$$

$$Confidence\ \text{int}erval\left(\sigma^2\right) = [8.77, 9.49]$$

The mean and variance for which confidence intervals are computed are those of the population to which values generated in the simulation experiment are assumed to be drawn. The true values of these are not known but it can be concluded with a certain degree of confidence, 95% in this case, that they lie within a given range, i.e., the confidence interval.

In essence, output analysis utilizes values generated from a simulation experiment as inputs so that other analytics can be carried out which give to results from which inferences can be made and applied directly into practice.

8. Conclusions and Recommendations

The paper has successfully demonstrated how quantitative data can be exported into a robust environment for performing extensive analytics. Mathematica was the environment utilized in this case but other similar environments such as Matlab, Python, R, etc. could have been used. The choice of environment would depend on the analyst's skill level in writing code within the environment, access to the environment, and the availability of libraries and functions within those libraries to perform the analytics that is desired. The paper, through incorporating a practical case study, demonstrated how data analytics on observations collected from field operations can generate results that can be used to guide decision-makers on effective improvement strategies to implement.

The case study presented in this paper was on an offsite concrete block production operation. In this operation, mixtures were made such that 1 bag of cement would produce 4 concrete blocks of size *200mmX200mmX400mm*. The blending of basic ingredients was done in a machine that is different for the block making machine. A mix that can produce 24 concrete blocks was produced each cycle but only a portion of that mix that can produce 4 concrete blocks was loaded into the block making machine at a time. The paper also demonstrated how insights into each activity of a production operation can be acquired from performing basic statistics and generating standard plots. It becomes evident which activities are highly variable for example "Haul" and "Return" tasks in the case study presented, and these considered if potential improvements to the entire operation are to be made. It was also shown how data models can be derived from empirical data and utilized in more advanced analytics, such as Monte Carlo simulation to obtain useful metrics for operations such as cycle length. In the concrete block production operation presented, 4 concrete blocks were produced in each cycle. Results from the Monte Carlo simulation experiment showed that the cycle length for producing these 4 concrete blocks had a mean of 41.40 minutes and a standard deviation of 3.02 minutes. Observation of the operations and discussions held with workers also indicated that availability of pallets onto which freshly produced blocks are placed is a major constrain to how much the process can produce.

This case study serves as a basis for building more comprehensive process interaction simulation models that can be used to experiment with different scenarios, for example, whether if adding a second block making machine with its own crew, there would be a need of adding another crew for hauling loaded pallets and returning to the machine or whether one such crew would be sufficient to service both machines to realize higher production rates.

It has been shown how preliminary and more detailed analytics can be performed on data collected in a

real-world system. Strategies for scrutinizing the data for quality issues have also been presented. The authors also demonstrated how to diagnose systems for inefficiencies, how to rank and prioritize those for effective improvement strategy development and implementation.

References

Angela. (2014). How to start making sense of your data. [Online] Available: https://versionone.vc/making-sense-data/ (December, 2019).

Doane, D. P. (1976). Aesthetic frequency classification. *American Statistician*, 30, 181–183. [Online] Available: http://www.jstor.org/stable/2683757 (December, 2017).

Hyndman, R. (1995). The problem with Sturges' rule for constructing histograms. [Online] Available: https://robjhyndman.com/papers/sturges.pdf (December, 2017).

Legg (2013). Improving Accuracy and Efficiency of Mutual Information for Multi-modal Retinal Image Registration using Adaptive Probability Density Estimation. DOI: 10.1.1.678

Mendenhall, W. & Sincich, T. L. (1995). *Statistics for Engineering and the Sciences*. (4th ed.). New Jersey: Prentice-Hall.

Statistics Canada. (2017). Constructing Box and Whisker Plots. [Online] Available: https://www150.statcan.gc.ca/n1/edu/power-pouvoir/ch12/5214889-eng.htm (April, 2020).

Sturges, H. (1926). The choice of a class-interval. *Amer. Statist. Assoc.*, 21, 65–66.