

Analysing relevant metrics and improvement methods to address data quality issues in the field of Oil and gas.

Rutuja Shah

A dissertation submitted in partial fulfilment
of the requirements for the degree of
Bachelor of Science
of the
University of Aberdeen.



Department of Computing Science

May 03, 2019

Declaration

No portion of the work contained in this document has been submitted in support of an application for a degree or qualification of this or any other university or other institution of learning. All verbatim extracts have been distinguished by quotation marks, and all sources of information have been specifically acknowledged.

Signed: *Rutuja Shah*

Date: May 03, 2019

Abstract

The purpose of this research is to analyze the data quality process which comprises of a set of dimensions, metrics and improvement methods used to enhance the quality of big data generated in oil and gas industries. From the past two decades, oil and gas companies have spent lots of money on data because they have come to realize its importance and the potential it possesses in improving their operational efficiency. But in order to get meaningful inferences from data analysis, the data needs to be of good quality. Oil and gas companies use numerical time series data (Log ASCII standard) for well logging. The quality issues faced by this kind of data are noise (random and systematic), gaps and outliers. It is important to transform the quality of data by improving the existence of these key issues. In order to achieve the best quality of data, it is important to choose the best improvement technique for each of the quality issues faced. The assumption is that the choice of best technique for a quality issue depends on the context in which it is being examined. This means that it depends on an organization's priorities, requirements, and the dataset chosen. In order to prove this, a set of relevant improvement methods have been devised for each quality issue and the results obtained have been evaluated. The purpose of doing this is to provide a set of meaningful recommendations to help those in the field of oil and gas pick the best improvement technique given the context. No research has been conducted in the past to provide a thorough understanding of the different factors (context) that contribute towards the decision-making process when choosing the best IM. Therefore, the exploration done by this research paper aims to contribute to an understanding of that. This research has also devised a set of metrics to assess each quality dimension along with a thorough walk through the components of the data quality process.

Acknowledgements

I would firstly like to thank my parents for giving me the opportunity to study at this university. I thank them for their constant support, love and encouragement. Secondly I would like to thank my supervisor Dr. Ernesto Compatangelo for his continuous support. He has always been there for me whenever I needed help and advice. He encouraged me at every step of the way which lead to a successful completion of this project. Thirdly I would like to thank Jacob Gelling for his collaboration and contribution to this project. Lastly I would like to thank Ishani Jayasinghe and Erika Litvin for being great friends and for always supporting me throughout my time here in Aberdeen.

Contents

1	Introduction	11
1.1	Problem identification and motivations	11
1.2	Overview	11
1.3	Objectives	13
1.4	Outline	14
2	Literature review	15
2.1	Data quality	15
2.2	Data quality framework and its components	16
2.2.1	Data quality dimensions	17
2.2.2	Data quality metrics	18
2.2.3	Data quality improvement methods	19
2.3	Relevance of this research	21
3	Design	22
3.1	Requirements specification	22
3.2	Data	23
3.3	Methodology	25
3.3.1	Dimensions	25
3.3.2	Metrics	26
3.3.3	Improvement	26
3.3.3.1	Noise	27
3.3.3.2	Gaps	28
3.3.3.3	Outliers	28
3.4	Software considerations	29
4	Implementation	30
4.1	Metrics	30
4.1.1	Completeness	30
4.1.1.1	Gap ratio	30
4.1.1.2	Simple gap percentage metric	30
4.1.1.3	Weighted gap percentage metric	30
4.1.2	Accuracy	31
4.1.2.1	Average absolute distance metric	32

4.1.2.2	Median absolute distance metric	32
4.1.2.3	Mode absolute distance metric	33
4.1.2.4	Relative error metric	33
4.1.3	Precision	33
4.1.3.1	Standard deviation	33
4.1.3.2	Relative uncertainty	33
4.1.3.3	Moving precision checker	34
4.1.4	Consistency	35
4.1.4.1	Consistency checker	35
4.1.4.2	Systematic consistency checker	35
4.1.5	Metrics relevant to each quality issue	36
4.2	Improvement methods	36
4.2.1	Random noise	36
4.2.1.1	Linear regression	37
4.2.1.2	Cubic parabolas	37
4.2.1.3	Cubic splines	38
4.2.1.4	Binning	39
4.2.1.5	Moving average filter	39
4.2.1.6	Savitzky-Golay filter	40
4.2.1.7	Fast Fourier Transform	40
4.2.1.8	Kalman filter	41
4.2.2	Gaps	42
4.2.2.1	Linear regression, cubic parabolas and cubic splines	42
4.2.2.2	Mean imputation	43
4.2.2.3	Autoregressive moving average model	43
4.2.2.4	K nearest neighbours model	44
4.2.2.5	Interpolation method	44
4.2.3	Outliers	45
4.2.3.1	K-means clustering	46
4.2.3.2	Two-sided median method	46
4.2.3.3	Interquartile range	47
4.2.3.4	Histograms	47
4.2.3.5	Mean intervals method	48
4.2.3.6	Moving precision checker	48
4.3	Software testing	48
5	Results and evaluation	49
5.1	Noise	49
5.1.1	Deciding between preserving the accuracy of the original points and maximum noise improvement	49
5.1.2	Deciding between saving storage costs and effective noise removal	50

5.1.3	Deciding between wanting a rough trend of the data points and observing data patterns.	50
5.1.4	Choosing a technique based on the natural behaviour of the data points	51
5.1.5	Choosing a technique based on noise considerations	51
5.1.6	Recommendations on systematic noise	52
5.2	Outliers	52
5.3	Gaps	53
5.3.1	Choosing a technique based on the visual appearance of data	54
5.3.2	Deciding between fast and effective techniques	55
5.3.3	Choosing a method based on the behaviour of the data points	55
5.3.4	Choosing a method based on data expectations	56
5.4	Recommended order in which these quality issues should be treated	57
6	Discussion and Future work	58
6.1	Discussion	58
6.2	Future work	59
7	Conclusion	60
A	User Manual	68
B	Maintenance manual	70
B.1	Installations	70
B.2	How to run the program	71
B.3	Organisation of files	71
B.4	Important files and functions	72
B.5	The dataset	72
B.6	Space and memory requirements	72
B.7	Extensibility and future use	73
B.8	Further information on weighted gaps	73

List of Figures

1.1	Big data analytics	12
2.1	Data quality framework relevant to the O&G field.	16
2.2	A categorical approach to data quality dimensions	18
2.3	Metric ratios for different dimensions	19
2.4	A slightly different type of metric ratio	19
3.1	LAS File containing temperature readings taken at different depths from a single well at a particular time	24
3.2	Heat map of the dataset	25
3.3	Zoomed-in noisy areas and gaps in the dataset.	27
3.4	Changes in uncertainty when noise is being reduced	28
4.1	Percentage completeness computed from lists containing the same number no. of gaps but different gap sizes.	31
4.2	Comparing the results obtained from the WGPM and SGPM.	32
4.3	AADM for measuring accuracy of a dataset.	32
4.4	Relative error metric for measuring accuracy of a dataset.	33
4.5	Relative uncertainty metric for measuring precision of a dataset.	34
4.6	Moving precision checker metric explanation.	34
4.7	Table containing metrics relevant to each quality issue.	36
4.8	Linear regression for reducing noise.	37
4.9	Cubic parabola for reducing noise.	38
4.10	Cubic splines for reducing noise.	38
4.11	Mean binning for reducing noise.	39
4.12	Moving average filter for reducing noise.	40
4.13	Savitzky-Golay filter for reducing noise.	40
4.14	Fast Fourier transform for reducing noise. The first image shows the original signal. The second shows the frequencies of the original signal. The third shows the original vs the output signal generated after passing the low pass filter.	41
4.15	The kalman filter for reducing noise.	42
4.16	Using linear regression, cubic parabola and cubic splines to predict missing values.	43
4.17	Using mean imputation to fill gaps	43
4.18	Using the ARMA model to fill gaps	44
4.19	Using the K nearest neighbour model to fill gaps	45

4.20	Using the interpolation method to predict missing points.	45
4.21	K-means clustering for detecting and removing outliers.	46
4.22	Two-sided median method for detecting and removing outliers.	47
4.23	Using Histograms to spot and remove outliers.	47
5.1	Results for noise improvement methods	49
5.2	Technique showing a rough trend vs technique preserving some noise	51
5.3	Bands showing what is considered as non-noisy.	52
5.4	Table containing results obtained from different outlier techniques.	53
5.5	Table containing results obtained from different gaps techniques.	54
5.6	Choosing a technique based on the visual appearance of data	54
5.7	Choosing a method based on the behaviour of the data points	56
A.1	Snapshot of the output generated from the main.py file.	68
A.2	Snapshot of the output generated from an improvement method.	69
B.1	Snapshot of how to run the program	71
B.2	Snapshot of the height and path variables available in each code file.	72
B.3	Variables set for the WGPM	73
B.4	The WGPM algorithm	74

List of Abbreviations

DQF	Data quality framework
O&G	Oil and gas
IM	Improvement method(s) ('method' or 'technique' used interchangeably)
SGPM	Simple gap percentage metric
WGPM	Weighted gap percentage metric
AADM	Average absolute distance metric
MADM	Median absolute distance metric
S.D	Standard deviation
MPCM	Moving precision checker metric
MPC	Moving precision checker
LR	Linear regression
FFT	Fast Fourier transform
KF	Kalman filter
MI	Mean imputation
KNN	K nearest neighbours
KMC	K-means clustering
SN	Systematic noise

Chapter 1

Introduction

1.1 Problem identification and motivations

The focus of this research is on the quality of numerical time-series data used in the oil and gas (O&G) industry. From the past two decades, O&G companies have spent a lot of money on data because they have come to realize its importance and the potential it possesses in improving their operational efficiency [64]. For example, by integrating a range of digital sensors into offshore equipment, elements that can affect production such as temperature, humidity and wave heights can be tracked and monitored. Using this obtained data, the offshore platform can be maintained effectively through predictive maintenance and by detecting potential equipment breakdown [41]. However, these effective insights can only be gained if the obtained data is of good quality. Data obtained can contain issues such as noise, outliers and gaps. Addressing these quality issues would improve the quality of data leading to a better operational efficiency.

With this, it is clear that data quality is vital in the O&G industry. Hence, to improve the quality of the data, the first step would be to define a set of quality dimensions based on company requirements. Based on this, the next step would be to formulate metrics to assess the quality of each dimension. The results from these metrics would determine whether or not the data requires improvement. If it does require improvement then the final step would be to apply improvement methods (IM) to address quality issues. All of these steps are components of the data quality framework (DQF). These frameworks are cyclic processes used by companies to assess and improve the quality of data (Further discussed in chapter 2). A lot of research has been conducted on these frameworks and its components but how can an O&G company, or any company in general, know what the best IM is to address each of the quality issues mentioned? In order to answer that question, it is important to understand that the choice of best IM depends on the context in which it is being examined. No research has been conducted in the past to provide a thorough understanding of the different factors (context) that contribute towards the decision-making process when choosing the best IM. Therefore, the exploration done by this research paper aims to contribute to an understanding of that.

1.2 Overview

The terminology 'Big data' has been ever-growing and many businesses have recognised its importance. Popular companies such as Amazon and big financial firms like Capital one along with industrial areas such as O&G, healthcare and transportation facilities use big data [92]. But what is big data and why is it so significant? In simple terms, big data is basically referred to as huge

volumes of data, such as data in Terabytes, that can't be searched, stored, analyzed or processed by using the traditional approaches (I.e.: relational databases) within a given time frame [67, 45]. It is an information asset characterized by high volume, variety and velocity which requires specialized technology and analytical methods for its transformation into a valuable form [16]. It is linkable information with complex data structures such as data obtained from various social media platforms or call records for example [54]. Big data requires a lot of processing power for efficient storage, analysis and manipulation therefore specialized technology such as Hadoop is used to handle it.

Big data on its own is relatively useless unless it can be used to drive a company's decision making process. At a seminar by Steven Rossiter on big data, he stated that companies like Amazon use big data for machine learning to sell people adverts effectively [75]. Big data can be used for data analytics in which data sets are analysed to benefit companies by providing them with meaningful insights that could help them enhance their business operations, develop new products and services along with creating new businesses. This can create benefits such as saving company costs, drive a better decision making and help them produce a higher quality of services as well as products [50]. For example, a company can use the data coming out of aircraft engines and use that to improve its fuel efficiency and performance [30]. So ideally, data analytics can be used to unlock meaningful insights by analyzing trends as well as patterns indicating how an organisation is currently performing and how they can do better. This data can also be used to make future predictions about processes or machinery that could potentially break and the information acquired can be used to prevent it from happening [30]. Figure 1.1 briefly highlights the data analytics process.

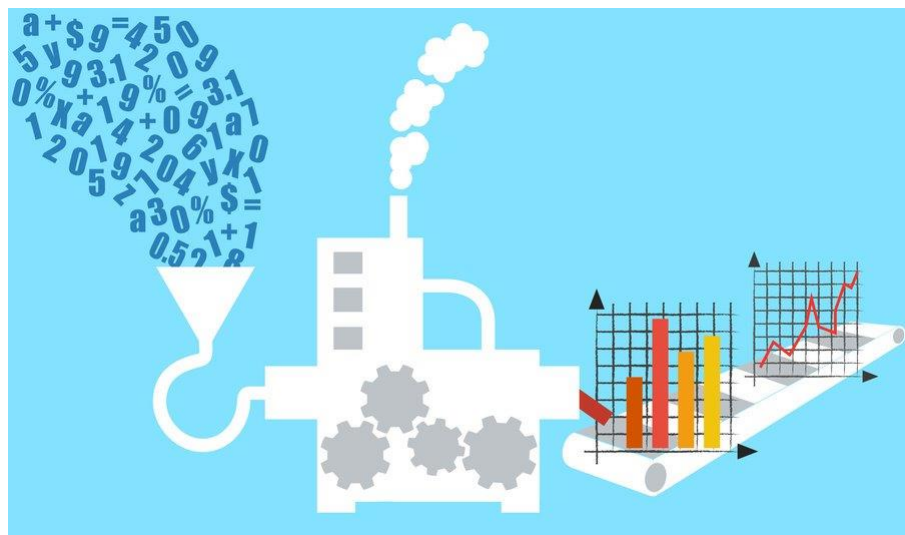


Figure 1.1: Big data analytics
[113]

Upon understanding the relevance and importance of big data in the various industries, it is important to note that data quality is also equally as important. The traditional definition of quality is based on the viewpoint that the data must meet the requirements of those who use it and is hence defined as 'Fitness for use' [80, 56, 86]. The reason this statement says 'to those who use it' is because data quality goals are organisations and context dependent. What may be classed as high

quality data for one organization may be classed as low quality data for the another. The specific quality goals depend on the objectives set by organisation which outline the requirements that need to be fulfilled in order to class the data as good quality. A precondition for analyzing and using big data is to use high-quality data in order to guarantee the value generated from it [12]. Big data can have biases, ambiguities, gaps and inaccuracies (noise and outliers) which need to be identified and either removed, accepted or improved in order to reduce inference errors when analysing data [80]. Bad quality of data would lead to inaccurate inferences being drawn from the data analytics process which can be fatal to organisations. Research suggests that losses due to poor data quality are estimated to exceed billions of dollars per annum and hence, it is important to improve the quality of data [35, 55]. Quality improvement can be achieved through the data quality process (DQF) consisting of dimensions, metrics and IM (Further discussed in chapter 2).

1.3 Objectives

Understanding the DQF is very important in order to improve the quality of the data in a systematic manner. One of the objectives of this research is to walk the reader through the steps of the DQF. This is done by thoroughly defining and explaining the significance of each component of the framework. An in-depth analysis is then conducted on the two main components of the DQF in order to achieve the second and third objectives of this research which are:

1. To explore different metrics used to assess the quality of data.
2. (main) To critically evaluate the IM for each data quality issue in order to understand which ones are *best* depending on the context in which they are being examined.

To meet the first objective, several metrics for assessing each aspect of data quality have been implemented in order to present the different kinds of metrics that exist. There is no research question to evaluate these metrics because there exists no standard for comparing the results obtained from the metrics against one another. This is because they are simply different ways of assessing the aspect of data quality in question. Assessing the quality of data is a very important part of the data quality process so it is important to acknowledge the different kinds of metrics that exist to assess it. It is important to understand how they function as the result obtained from each metric would be different from the other. Depending on how they function and the type of data being assessed, an expert would then be able to decide which metric to choose for assessing the quality of their data. This exploration would hence be meaningful and of interest to those who are in charge of assessing data quality.

As with the second objective on data quality improvement, several IM have been implemented for each data quality issue. The results obtained from these IM are critically evaluated to demonstrate which method is 'best' given the context. Data quality improvement is the end goal of the DQF and hence, it is essential that the best IM is chosen in order to provide the best possible quality of data. It is important to understand which method to choose under certain requirements, priorities and the type of dataset being dealt with. Information on which method to use amongst the others has been missing in data quality research so this paper aims to deliver the knowledge needed to make this decision. This in-depth analysis would be useful and of great interest to those in charge of improving the quality of data within the field.

1.4 Outline

This paper has been organized as follows. The next chapter describes the related work on data quality, DQF, dimensions, metrics and IM followed by a section explaining the relevance of this paper. Chapter 3 presents the research question along with the requirements specification, dataset used and the design of this research. Chapter 4 describes all the metrics developed for each dimension and the IM developed for each quality issue. Chapter 5 presents the results obtained from each quality issue along with a set of meaningful recommendations. Chapter 6 presents the discussion and finally, chapter 7 presents the conclusion. The user and maintenance manuals have been provided in the appendices.

Chapter 2

Literature review

2.1 Data quality

Between the 1950's and mid 1990's, data grew at a slow rate because computers, storage and data networks were very expensive. The arrival of the World wide web in the 90's led to the sudden growth of data and the emergence of data analytics. Data analytics has proven to be of great value to organisations and therefore, the money spent on it has been estimated to grow from \$1.6 billion in 2015 to \$5.4 billion by the year 2020 at an annual growth rate of 27.6% [50]. But to gain meaningful insights with the use of analytics, the quality of data needs to be up to the mark. This is because data quality has been a very important part of company deliverables such as products and services. However, the awareness of its value and the emergence of methods to measure as well as improve it has been an evolutionary development [65]. In 1996, [96] began research on data quality by identifying quality dimensions (discussed in the next section). Ever since then, data quality has been addressed from a wide range of perspectives.

Studies have confirmed that when dealing with data quality, companies must focus on two very important aspects: the objective aspect and the subjective aspect [96, 47, 6, 40]. A book by [65] defines data quality as 'inversely proportional to variability'. This means that if the data values are highly varying (high S.D) their quality would be low. Having said that, high quality data could be defined as data with an absence of noise. In other words, absence of noise would mean that there is no difference between the 'True' value and the recorded values of the data set [80]. In reality, it is almost impossible to achieve a data set with recorded values that are identical to the 'True' value but what [65] definition says is that the smaller the difference between the true and recorded values, the higher the quality would be. This definition only defines data quality from the objective aspect (data perspective). As seen in the introductory paragraph, data quality has been defined as 'fitness for use' by many researchers [80, 56, 86]. This means that quality of data depends on the data quality goals set by individual organisations. These goals are company specific requirements that the data needs to meet in order to be classed as acceptable quality. This is the subjective aspect of data quality. These quality goals reflect the needs and data requirements set by individuals involved with the data such as stakeholders. When data quality is considered, both the objective as well as subjective aspects need to be considered as they are equally as important.

Most of the data quality research has been conducted on databases, linked data, images and

a variety of other domains [87, 4, 18]. The type of data used for this research is numerical time-series. Very little research has been conducted on the quality considerations for this type of data. Numerical data quality can be defined as the measure of deviation of a given data set from a pre-defined standard. Numerical data sets in O&G industries either come from measurements (I.e: temperature) obtained from a variety of sensors, mathematical calculations or other algorithmic processes. The quality of the recorded values in these data sets can be compromised by a number of issues such as noise, outliers or gaps [60]. These issues can be critical and need to be addressed. In order to improve data quality, it is important to first assess its quality. This would inform the user which aspects of the data require improvement. In order to efficiently assess and improve the quality of data, a well-defined data quality framework needs to exist, a discussion is presented next.

2.2 Data quality framework and its components

One of the first frameworks for data quality research was presented by [57] in 1992. The researchers suggested that data quality should be characterized using a total data quality management cycle which calls for defining, measuring, analyzing and improving data quality as a continuous process [35, 57]. Thenceforth, several efforts to summarize, classify and develop frameworks for the data quality research have been made by different researchers [77].

In 2007, [83] took an approach which focused on the assessment, management and contextual aspects of data quality. The data quality assessment category was then further broken down to problem identification, data quality dimensions and assessment methodologies. Similarly, a recent paper by [77] in 2017 introduced a methodology for the DQF which consisted of three stages: data collection, data preparation and analysis. These two frameworks along with the one introduced by [57] in 1992 are similar to the kind of framework used for this research. The framework used in this research is a simple framework developed by [60]. It is specific to the needs of the O&G industries in terms of measuring, analyzing and improving the quality of numerical time-series data. Figure 2.1 shows this framework.

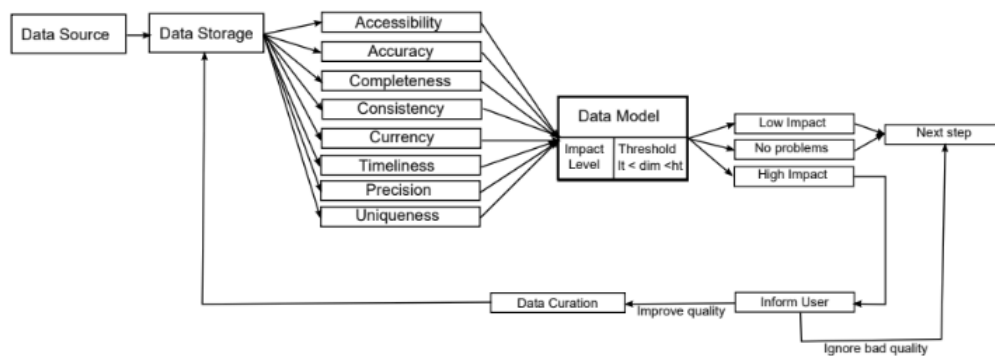


Figure 2.1: Data quality framework relevant to the O&G field.

[60]

The first step is **data collection** in which data generated from information sources such as sensors provide the initial input to the workflow process. The data collected is then stored until it is

ready to be processed for the next step. The second step is to **define** a set of data quality dimensions which represent aspects of data quality. These dimensions are specific to the requirements and quality goals set by the company. The third step is to **measure** each of the dimensions defined in the previous step with the help of metrics. Each metric computes a score with the use of thresholds in order to class the data as either low impact (good quality) or high impact (bad quality). These thresholds are pre-defined by the company. The fourth step is to **analyze** the results obtained from the metrics in order to determine whether quality decisions need to be made on the data or if it meets the requirements to proceed to the next level of the process (I.e: analytics). If the data is not satisfactory, quality decision makers may choose to either discard, keep or improve the faulty values. If the decision is to improve the quality of data, then the data goes through the fifth step of the process which **improves** it with respect to the dimensions that did not meet quality requirements. This is a cyclic process because the data keeps going through all the stages until it qualifies as good quality data and meets the requirements to proceed to the next level of the process.

So, it is important to first define/choose relevant dimensions in order to make choices on the corresponding metrics which then determine the areas of improvement in the data. It is essential to fully understand these three key components (dimensions, metrics and IM) of the DQF [60, 96, 57]. Therefore, the following three subsections cover the definitions along with the related work on these three key components.

2.2.1 Data quality dimensions

After data storage, the second step is to *define* a set of data quality dimensions. [96, 57]. These are a set of data quality attributes that represent a single aspect of quality [12]. For example, a quality dimension 'completeness' represents the completeness aspect of the data set. It indicates whether or not the dataset is complete. These dimensions help check the quality of collected data in relation to specific data quality goals [12]. There are many such dimensions and each allows the user to measure as well as manage the quality of data. Hence, research suggests that data quality is a multi-dimensional concept [96, 47, 6, 40]. These dimensions are context dependent which means that they are chosen based on the kind of data being examined and the data quality goals set by organisation [63]. Because of the context-dependency of dimensions, there is no mutual agreement on which dimensions to use when [95].

Previous work conducted on dimensions has been devoted to non-numerical data (I.e: database fields). In 1996, [94] determined four categories for classifying data quality. These are intrinsic, accessibility, contextual and representational. These categories were used by several researchers as shown in figure 2.2 with each category containing dimensions relevant to that domain of research.

The categorical approach was not taken by all researchers when defining relevant dimensions. In 1999, [2] described six dimensions relevant to web data which are: accuracy, authority, objectivity, currency, transaction features and intended audience. Research on dimensions in the domain of society and culture had been carried out in 1999 by [78] who developed an emotic-based framework for addressing data quality with a total of 11 dimensions [12]. In 2000, [112] proposed six dimensions relevant to information retrieval which are: currency, availability, authority, cohesiveness, info-to-noise ratio and popularity. Furthermore, [24] proposed four quality

	Intrinsic IQ	Contextual IQ	Representational IQ	Accessibility IQ
Wang and Strong [39]	Accuracy, believability, reputation, objectivity	Value-added, relevance, completeness, timeliness, appropriate amount	Understandability, interpretability, concise representation, consistent representation	Accessibility, ease of operations, security
Zmud [41]	Accurate, factual	Quantity, reliable/timely	Arrangement, readable, reasonable	
Jarke and Vassiliou [16]	Believability, accuracy, credibility, consistency, completeness	Relevance, usage, timeliness, source currency, data warehouse currency, non-volatility	Interpretability, syntax, version control, semantics, aliases, origin	Accessibility, system availability, transaction availability, privileges
Delone and McLean [11]	Accuracy, precision, reliability, freedom from bias	Importance, relevance, usefulness, informativeness, content, sufficiency, completeness, currency, timeliness	Understandability, readability, clarity, format, appearance, conciseness, uniqueness, comparability	Usableness, quantitateness, convenience of access ^a
Goodhue [14]	Accuracy, reliability	Currency, level of detail	Compatibility, meaning, presentation, lack of confusion	Accessibility, assistance, ease of use (of h/w, s/w), locatability
Ballou and Pazer [4]	Accuracy, consistency	Completeness, timeliness		
Wand and Wang [37]	Correctness, unambiguous	Completeness	Meaningfulness	

Figure 2.2: A categorical approach to data quality dimensions [51]

dimensions for databases which are: accuracy, correctness, completeness and consistency. These four dimensions have been widely used in many other researches [94, 74, 66].

Evidently, a defined set of data quality dimensions does not exist. It all heavily depends on the kind of dataset being examined and requirements set by organisations. The dimensions viewed above come from different domains such as web data, databases, society and culture and many more. It is clear that most of the papers discussed address dimensions for non-numeric data. However, a paper by [60] defines eight dimensions relevant to numerical datasets which are: accessibility, accuracy, completeness, consistency, currency, timeliness, precision, and uniqueness. These are further discussed in the following chapter.

2.2.2 Data quality metrics

After defining quality dimensions, the third step is to *measure* the quality of data using a set of metrics [58, 57]. With each selected dimension, metrics are formulated to quantify/measure that dimension. For example, the dimension 'Precision' can be measured using S.D. These metrics compute an assessment score for the dimension in question which is then used to determine whether the data is suitable to proceed to the next level of the process (I.e: analytics). It is worth to note that several metrics can be used to measure a particular dimension but each metric only measures one dimension [109].

In most research on quality assessment, metrics take the form of a ratio. This ratio measures the frequency of observed outcomes out of the frequency of the desired outcomes [109]. Most researchers [83, 91, 9, 71, 45, 4] have not developed metrics specific to each dimension but have rather used the same kind of ratio for all dimensions as seen in figure 2.3.

In 2002, [71] suggested the following kinds of metrics: Simple ratio, min or max operation, and weighted average. The simple ratio is the same as one described in figure 2.3 but with a slight variation. In this, the number of undesirable outcomes are divided by the total number of outcomes subtracted by 1 (figure 2.4). This computes a scores between 0 and 1 with 1 represented as the most

DQ Dimensions		Metric functions
<i>Accuracy</i>		$Acc = (Ncv / N)$
<i>Completeness</i>		$Comp = (Nmv / N)$
<i>Consistency</i>		$Cons = (Nvrc / N)$
<i>Ncv</i>	<i>Number of correct values</i>	
<i>Nmv</i>	<i>Number of missing values</i>	
<i>Nvrc</i>	<i>Number of values that respects the constraints</i>	
<i>N</i>	<i>Total number of values (rows) of the sample Dataset</i>	

Figure 2.3: Metric ratios for different dimensions
[83]

desirable and 0 as the least desirable score. This ratio has also been used by other researchers such as [45]. The min and max operations are used for handling dimensions that require aggregations of numerous quality indicators/variables after the individual variables have been measured using a simple ratio. The mix or max values are computed among the normalized values of individual data quality indicators. The min operator assigns a dimension with an aggregate value no higher than the value of its weakest quality indicator while the max operator is used if a liberal interpretation is needed. Finally, the weighted average is an alternative to the min or max operation and is used for a multivariate case. It is used when an organization has a good understanding of the importance of each variable and wants to make an overall evaluation of a dimension [71]. Further discussion on the literature concerning metrics has been provided in chapter 4.

$$D = 1 - (Ni/Nt)$$

Figure 2.4: A slightly different type of metric ratio
[45]

2.2.3 Data quality improvement methods

If in the fourth step of the DQF, a decision has been made to improve the quality of data then the data enters the fifth step of the process. In this, data is transformed from an unsatisfactory quality to a quality that meets the requirements of the organisation. This needs to be done prior to analyzing data with an unknown quality [82]. There are two strategies for improving data quality. The first one is data-driven in which techniques are applied to clean and hence improve the quality of data. Process-driven on the other hand identifies the root causes of poor data quality and redesigns the process which creates or records this data. This report focuses on the data-driven strategy which improves the quality of data by addressing the dimensions that don't meet quality requirements [82]. These are identified with the use of metrics as seen in the previous section. These should not be ignored as they can lead to incorrect conclusions when analyzing data [53]. Data improvement, most often referred to as data preprocessing, can be done in many ways in which data is either cleaned, filtered/reduced, ordered, integrated or discretized [21, 108]. The quality issues in the O&G industries that are subject to improvement are: Noise, gaps and outliers. It is important to

first understand what each of these mean before discussing their corresponding IM.

Corrupted or distorted data that provide meaningless information is known as noise [102]. This could be due to reasons such as sensor failure, improper data entry, failure in data transmission etc [21]. Two types of noise are relevant to this research:

1. Random noise: Also known as white noise, often contributes to a large portion of noise in the data. It is data that a system cannot understand or interpret correctly because it is random in nature [102]. This type of noise can be defined as values that fluctuate within close radius to the true value. This randomness makes it difficult to identify. This terminology is most often used for acoustic data but can also be used for numerical data sets as well.
2. Systematic noise: These are errors that repeat throughout the dataset and the main source of problem is the system. In the O&G industry, this type of noise can occur because of heat or pressure. For example: if a well is being drilled, the drill could heat up because of friction and this could cause the drilling pipe to heat up too. This could affect the temperature values recorded by the sensors at that particular time. These errors are usually easy to detect as they have a pattern that repeats throughout the dataset. They can be spotted out with the help of visual heat maps of the dataset [60].

Commonly used IM for addressing noise in general are regression, fast fourier transform, convolution filters (I.e: the moving average, Savitzky and Golay filter, 4253H twice filter, mean-value iteration filter, ARMD3-ARMAS filter), binning methods, gaussian function filtering, kernel density estimation, kalman filters, and partitioning algorithms (derived from the kalman filter) [82, 21, 108, 49, 38, 85, 106, 13].

Gaps, as the name suggests, are values that are missing from a dataset. A dataset is said to have gaps if it does not have all the expected values. These gaps could occur due to a temporary interruption in the sensors that record values. Quality of data is improved by filling these gaps. The most common gap IM are imputation techniques such as mean, hot-deck, cold deck, distribution-based, statistical (eg: regression), and multiple imputation. There are many popular machine learning techniques such as autoregressive moving average (ARMA), k nearest neighbours (KNN), Multi-layer perceptron, Self-organisation maps and the ANN prognosis model. Other techniques for filling gaps include global constant, moving average and inference-based models [11, 43, 107, 34, 42, 29].

And lastly, an outlier is an observation point that lies an abnormal distance from the rest of the values. This could be because of variability in the measurements recorded or may indicate experimental error. Outliers are most commonly detected using supervised classification, semi-supervised recognition and unsupervised clustering. Common approaches include density based (I.e: local outlier factor), clustering (I.e: k-means), distance based (I.e: KNN), classification, distribution based, and graph based models. Statistical techniques include control charts, interquartile ranges (IQR), histograms and the median absolute distance (MAD) [108, 61, 107, 34, 1, 44, 73, 11, 65].

This provides a brief overview of the existing IM used to address each quality issue. Most of these IM are further discussed in chapter 4 since it is not possible to fully elaborate on each here. Although it is clear that there has been a lot of research on the different kinds of IM that exist

for improving data quality, no research has been conducted in the past to provide a thorough understanding of the different factors (context) that contribute towards the decision-making process when choosing the best IM.

2.3 Relevance of this research

This research is focused on the quality of numerical time-series data generated in the O&G industry from devices such as sensors and other mathematical operations. The use of big data and analytics has proven to be very beneficial to the O&G industry in a number of different areas. In the area of *drilling*, data monitoring can be used to alert any anomalies based on conditions or predict the possibility of drilling success. In the area of *production operations*, analytics can be applied to seismic, production and drilling big data to help engineers map changes in the reservoirs and provide the knowledge needed to make changes to lifting methods. And finally, it could also be beneficial in the area of *maintenance*. If pressure and temperature can be collected and analyzed, past history on a compressor (for example) can be compared making it possible to automate alerts. This could be beneficial when critical assets are involved and failure to detect could have a significant impact on health, safety and environment [22].

As seen above, data analytics is very beneficial to the O&G industry. But, in order to get meaningful insights from these analytics, data needs to be of good quality. The quality of data is compromised due to the data quality issues discussed. Hence, it is important to measure the quality of data to address the existence of these issues and to then improve the quality by resolving them. Choosing the best IM is not easy because there are many factors that need to be considered beforehand and there is not even a single research that focuses on a thorough understanding of this. Therefore, several IM have been implemented for each quality issue and the results obtained have been evaluated. The purpose of doing this is to provide knowledgeable information to those in the field of O&G with the intention to help them make the best choice when improving the quality of their data. The following chapter presents the research question and describes how this research will be executed in order to meet its goal.

Chapter 3

Design

Acknowledging the significance of data analytics in the O&G industry and how data quality impacts the results obtained from the analytical process, the centrality of data quality triggers the following research question:

Which is the best improvement method for addressing data quality issues: noise, gaps and outliers?

Hypothesis: This decision depends on the context in which the numerical data is being examined.

The word 'context' means a variety of factors (I.e: requirements and priorities) that need to be considered when making a decision on the most appropriate IM. What type of numerical data is it (trendy or random)? What is its source? What are the intended requirements of the data set by organisations? What conditions should the data meet in order to be classed as good quality? What are the priorities? The answers to these questions depend on requirements set by individual organisations and hence, it is important to understand that data quality is a context-dependent concept. This purpose of proving this hypothesis is to provide meaningful knowledge that would help organisations pick the best IM given a context. The following sections provide details about the conditions as well as considerations on which this research is based upon and how it will be conducted.

3.1 Requirements specification

This paper, in accordance with the research purpose, is not focused on producing a substantial codebase but rather on the meaningful results drawn from the concept-proofing code implemented. Hence, not a lot of emphasis has been put upon the requirements specification as it is not an essential part of this research. However, the software is useful and would be of interest to those who would like to use the metrics and IM developed. Therefore, the requirements are as follows:

Functional requirements:

Must have:

1. The system must be able to output results for any dataset as long as it is in the same format/standard as the one currently used.
2. The system must be able to output results for any depth as long as that depth exists in the dataset being used.

3. The user must be able to select a dimension of their choice and obtain results from all the corresponding metrics on the console
4. The user must be able to select an IM of their choice and obtain results on the console as well as in a graphical form.
5. The user must be able to view a visual heat map of the dataset in question and for any other dataset as long as it is in the same format/standard as the one currently used.
6. For each improvement method, the user must be able to *view* a new set of improved time (x (if changed)) and temperature (y) values for the depth in question.

Should have (additional) :

1. For each IM, the user should be able to *save* a new set of improved time (x) and temperature (y) values for the chosen depth in a csv file.
2. For each improved set of values, the system should be able to replace the old values in the dataset with the new improved ones.

Non-functional requirements:

1. The system must be able to interact with the user on the console efficiently.
2. The system must contain comments and cite relevant bits of code.
3. When initially run, the main file could take about 5 seconds to run but once it has been compiled, the rest of the operations must respond under a second.
4. Metrics and improvement methods can be extended easily.
5. The system is portable and must run on any operating system as long as python and required libraries are installed.

3.2 Data

The dataset used is in log ASCII standard (LAS) file-format. This format is commonly used in the O&G industries to store well log information. The purpose of well logging is to investigate and characterize the subsurface stratigraphy in a well [101]. The dataset has been acquired from an O&G company operating in the UK continental shelf. The dataset contains a few folders with each containing readings taken from one well on a particular day. All the files in each folder are of LAS file-format and contain readings taken from the optical sensors in distributed temperature sensing (DTS) devices. The accuracy of these optical sensors is 0.01°C (+1%). Each file in the a folder is a time slice containing temperatures recordings taken from depths around 0m (top of well) to 2000m (bottom of well). Figure 3.1 shows a snapshot of a LAS file.

Since time, depth and temperature values are recorded, this dataset is of 3D nature. Figure 3.2 displays a snapshot of the heatmap created from this data. The independent variables are time (x) and depth (y). The dependent variable is the temperature recorded (z in colour). For this research, values from depth (horizontal) slices have been used for metric and improvement methods. Which

```

~Version Information
VERS.                2.0: CWLS Log ASCII Standard
VERSION 2.0
WRAP.                NO: One line per depth step
~Well Information Block
STRT.m  0.5994:
STOP.m  2823.6865:
STEP.m  1.0150
NULL.    -999.2500:
TIMESTAMP: 2014/07/17 19:19:59
~Curve Information Block
DEPTH.m
TEMP_DTS.degC
~A
0.5994      17.9640
1.6144      18.5110
2.6284      18.7560
3.6424      18.3600
4.6573      17.7600
5.6713      16.9190
6.6863      16.6840
7.7003      16.9880
8.7153      17.3130
9.7293      17.1350
10.7432     16.7010
11.7582     16.5380
12.7722     16.5530
13.7872     16.6280
14.8012     16.6370
15.8162     16.5210

```

Figure 3.1: LAS File containing temperature readings taken at different depths from a single well at a particular time

means that for the chosen depth, its corresponding time and temperature values are used. Depth slices have been chosen because temperature values are supposed to stay more or less the same at a particular depth no matter what the time is. So, a fluctuation in values would indicate the presence of noise at that depth. Another reason why depth slices have been chosen is because similar surrounding temperature values need to be considered when cleaning noise and filling gaps. And since temperature values are more or less the same at a particular depth, filling gaps or removing noise would be more reliable if done using depth slices. It is not ideal to work with depth and temperature values obtained from time (vertical) slices because temperature values vary at each depth. Therefore, these would appear to be very noisy although they are not and smoothing these out would not make sense as temperature values are unique at each depth. In order to take depth (horizontal) slices, a matrix of values has been created so that it is easy to obtain temperature and time values for any depth.

Two different folders have been chosen to work with. One contains gaps and the other does not (reason discussed in section 4.2). As seen in figure 3.2, the time has been converted into minutes for each folder. The timestamp of the acquired dataset was in DATETIMEw.d format (I.e:2014/07/17 19:19:59) but has been converted into minutes using TIMEw.d. This has been done to make working with time (x) values easier. These time readings are not strictly at regular intervals but are roughly taken every minute. This is a constraint because some IM require regular intervals for best results. Another constraint is that the dataset provided is not very noisy. This makes it difficult to detect and improve noise efficiently.

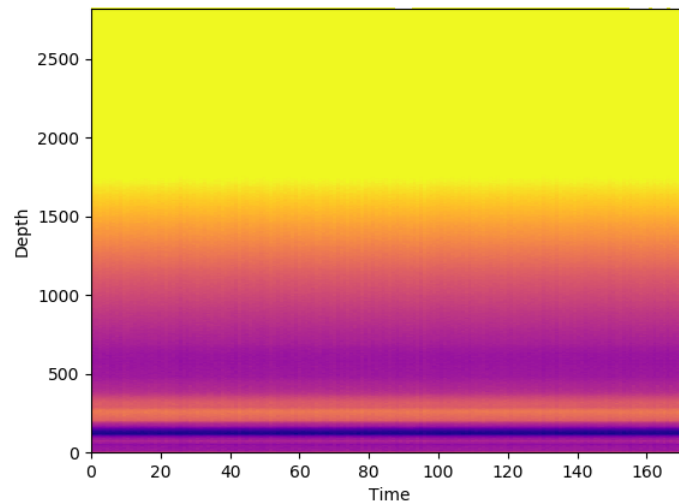


Figure 3.2: Heat map of the dataset

3.3 Methodology

3.3.1 Dimensions

Eight dimensions for addressing numerical data quality had been identified in chapter 2 but not all are these are used to measure the specific quality issues faced in the O&G industries. In his paper [60] suggests completeness, accuracy, timeliness, currency, precision and consistency as relevant dimensions used for measuring these quality issues encountered in numerical time-series data. However, it is not possible to measure the timeliness and currency of the dataset acquired because knowledge about dataset creation has not been provided and also because the research is not carried out in a real-world environment. Hence, the four dimensions most relevant to the research in this context are: Accuracy, consistency, completeness and precision. Each of these will now be defined for thorough understanding.

Accuracy can be defined in terms of syntactic accuracy and/or semantic accuracy. Semantic accuracy is described as the degree to which the recorded values are equivalent (close) to the 'True' value(s) [60, 23, 35]. The closer the values are, the more accurate the recorded data is. In cases where the 'True' value is unknown, syntactic accuracy (sometimes classed under the correctness dimension) can be measured instead. In this case, it is checked whether the recorded value belongs to a corresponding domain D . Domain D can be assumed to be a range of acceptable values that are classed as accurate [23, 109]. So, if the recorded data points fall within these range of acceptable values then they are considered accurate.

Precision on the other hand is completely independent of the 'True' value [93]. This dimension refers to the repeatability of a set of numerical values. It addresses the closeness between two or more successive values recorded under the same conditions. Precision is thus inversely proportional to the dispersion of a set of values. Therefore, the lower the S.D of a set of values, the higher the precision [60].

Consistency can be defined from two aspects. One being the degree to which data follows a set of predefined rules such as format, type and structure [60]. And the second is similar to

precision which addresses the closeness between two or more successive values recorded under the same conditions. Values are considered consistent if they repeat which in other words means the lower the dispersion of the data, the higher the consistency. The second definition is used to measure the consistency of datasets for this research.

Finally, *completeness* refers to the verification that a dataset contains all the expected elements [60]. A dataset is considered to be complete if it contains recorded values for every meaningful state of the system (optical sensors in this case) [81]. In other words, a dataset is considered complete if it does not have any gaps or missing values.

3.3.2 Metrics

The dimensions relevant to this research are accuracy, consistency, completeness and precision. As defined in section 2.2.2, metrics formalise a way to measure these dimensions. A set of metrics have been developed to measure each of these four dimensions. A few of these metrics are straightforward ratios obtained from relevant literature and are applied directly to the data given. While others, devised by this research, are in the form of algorithms that use ratios after computing results from the algorithmic process. Consequently, both these types of metrics do use ratios, nonetheless, the results obtained would be different depending on the process through which they have been computed. Both of these are classed as metrics because both are forms of measuring the aspect of quality in question.

These metrics highlight areas for improvement. Most of the metrics developed in this research produce scores based on predefined thresholds. These results are then used to determine whether the aspect of data quality measured requires improvement or not. For example, a completeness metric would measure whether the data has any gaps. If results obtained exceed a threshold limit, then improvement considerations need to be made. Not all dimensions and metrics developed are specific to a quality issue. In some cases, the same dimension and metric can be used to assess two different quality issues. For example, the accuracy dimension can be used to measure the presence of both random noise and outliers. A table of quality issues and its corresponding metrics are presented in chapter 4.

3.3.3 Improvement

It would be very time consuming and infeasible to measure as well as improve the quality of data at each of the 2000+ depths. Thus, the ideal option would be to first visualize the dataset so that any depths with noise or gaps can be identified. Metrics and IM then only need to be applied to these identified depths. Although visualizations help identify quality issues, metrics still need to be applied to the identified depths to check whether the data does indeed require improvement by evaluating the scores returned. This is because what may appear noisy may not necessarily be noisy as per company requirements. Figure 3.3 shows a zoomed-in snapshot of the visualization of the dataset.

From the visualization in figure 3.3, the yellow patches are potential noisy values between depths 860 to 900m. There is also a gap between the 85th to 90th minute. If a decision has been made to improve these, then noise, outlier and gap techniques need to be applied. The following sections describe the basis on which different IM have been compared for each quality issue. The results obtained from these are then contributed to chapter 5.

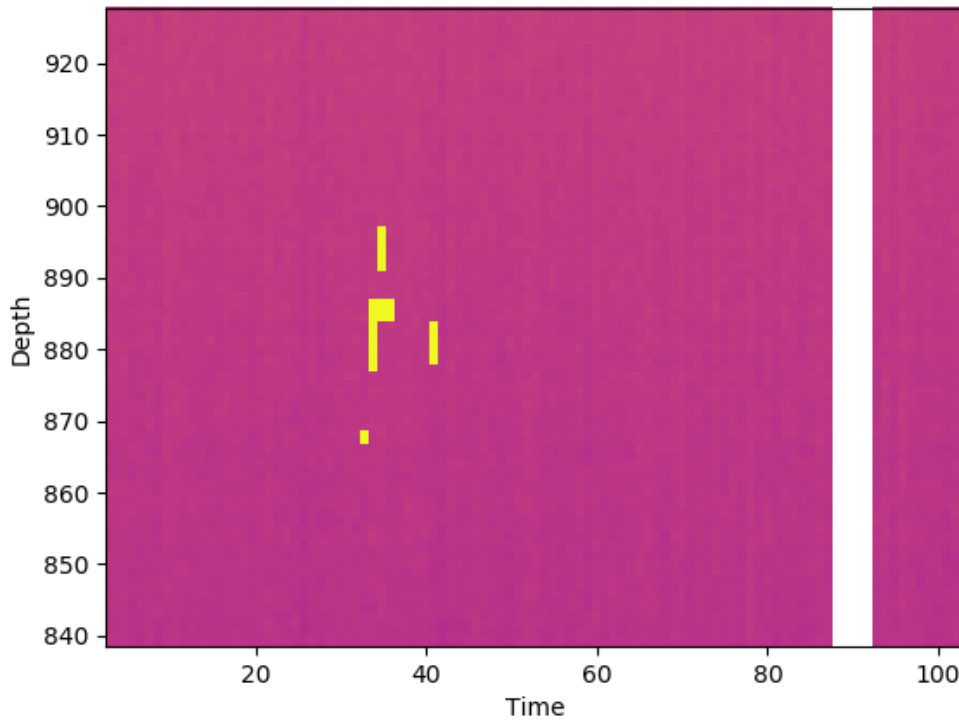


Figure 3.3: Zoomed-in noisy areas and gaps in the dataset.

3.3.3.1 Noise

It is important to understand what happens when noise is being reduced (improved) when IM are applied to the data. Every temperature value has an uncertainty of 0.01°C (I.e 16.52 ± 0.01). This means that the real value varies between 16.51°C and 16.53°C inclusive. The 0.01°C is the level of uncertainty in the temperature values recorded by the optical sensors and this has been verified by an expert who provided this dataset.

When methods to reduce noise are applied to the data, the uncertainty associated with each temperature value increases. For example, if an original point 16.52°C with an uncertainty of $\pm 0.01^{\circ}\text{C}$ is improved to 16.75°C (when reducing noise) then the uncertainty of the original point changes to $\pm 0.23^{\circ}\text{C}$ ($16.75 - 16.52$) so that it can cover/fit the improved point. In this case, uncertainty of the original value has increased by 0.22°C ($0.23 - 0.01$). To understand this better, consider point N1 in figure 3.4. The uncertainty of the original point (blue band) must increase (red band) in order to fit/cover the improved point on the regression line. It is important to understand that reducing noise increases uncertainty which reduces the accuracy of the original values. Accuracy of the original points is important because as previously defined 'Quality is inversely proportional to variability' so by increasing the uncertainty of the original points, you are actually increasing the variability of the data and hence reducing its quality. So while the quality of data is being improved by reducing noise, the quality of the original points is being lost. These original points may unlock some interesting phenomenon in gas wells which may be missed out when IM are applied. Therefore, accuracy of the original points is important and not a lot of research has been conducted in the past to address this.

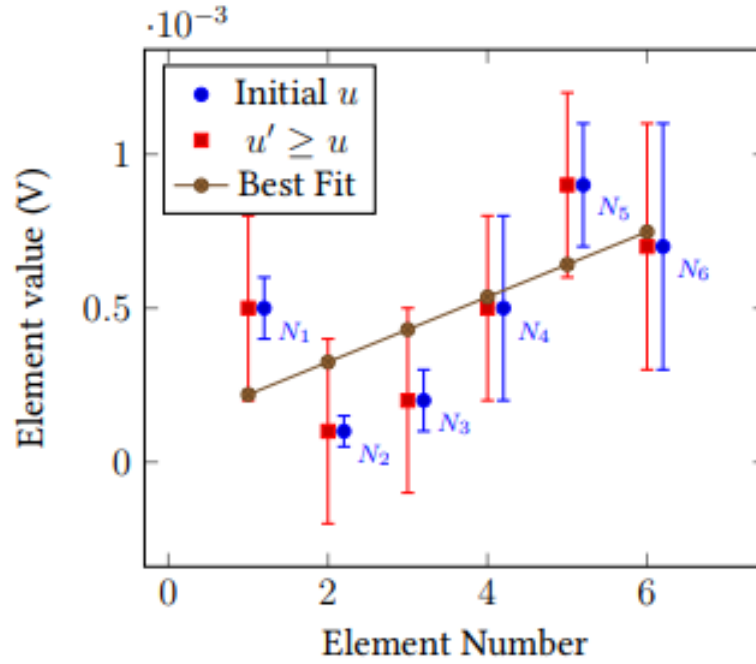


Figure 3.4: Changes in uncertainty when noise is being reduced

This average uncertainty increase is recorded upon the execution of each noise IM. This is done by first calculating the absolute uncertainty change by taking the difference between each of the recorded (original) and improved points. Each of these differences are then subtracted by 0.01 to get the uncertainty increases. An average of all these uncertainty increases is then taken to output an average uncertainty increase score for the IM in question. These calculations only address random noise and will be evaluated in chapter 5. For systematic noise, context-dependent suggestions are provided in the same chapter.

3.3.3.2 Gaps

Gap IM are used to fill missing data points in a dataset to improve its quality. A complete dataset has been used and original values have been extracted in places to create gaps. This is done so that a comparison can be made between the original values and the values obtained using gap IM. An absolute distance is calculated to measure the difference between the original values and the values obtained using a gap IM. An average of all these differences is then computed to output an average absolute distance score for the gap IM in question. A small score would indicate a similarity between the original values and the values obtained using the gap IM. These scores are then used to compare and evaluate which gap IM are better than the others.

3.3.3.3 Outliers

In order to evaluate the performance of outlier IM, similarities between results obtained from each IM have been compared in order to conclude which IM are better than the others (Further explained in section 5.2). This has been done because there is nothing to compare the result of each outlier IM with as knowledge of outliers has not been provided (such as the range beyond which values are considered as outliers). Each technique outputs outliers based on a predefined threshold value so that results from all IM can be compared on similar grounds.

3.4 Software considerations

The objected oriented approach had been considered but was not taken because each metric and IM is completely unique and independent from each other. Besides taking x and y lists as input, methods do not share any common functionalities and hence can not be re-used anywhere else. Therefore, functions have been created instead and reused where necessary. The development of this project has been executed by an in-depth research on the relevant techniques within the area and a specification of requirements was finalized before the commencement of the software. Hence, no agile development was required. This system does not have an architecture of any sort. The results from each IM are simply outputted on the command line. Further implementation details have been provided in appendix B.

Chapter 4

Implementation

The following chapter will run through all the metrics and IM developed. The programming language used for development was Python version 3.6. The python libraries, modules and functions used for implementation are provided in appendix B.

4.1 Metrics

The following subsections present metrics developed for each dimension relevant for assessing data quality for this research. As mentioned in section 1.3, the purpose of presenting different kinds of metrics is to provide meaningful knowledge to those who are in charge of assessing data quality. All of these metrics use data from DTS_G1_BLEED_1 folder. Time (x) and temperature (y) values used for each metric are obtained from default depth 189.2785m (the word data or dataset used in the following sections mean these x and y values at the default depth). Most of these metrics only consider temperature values (y) to do all the calculations. Each of these metrics compute a different score since they operate differently. This is why it has been previously mentioned that it not possible to compare them against one another.

4.1.1 Completeness

4.1.1.1 Gap ratio

The metric shown in figure 2.4 returns the completeness of the dataset as a ratio between 0 and 1 [45]. N_i is the number of missing items and N_t is the number of expected items. In this case, a score of 1 would mean the dataset is complete and a score of 0 would mean otherwise.

4.1.1.2 Simple gap percentage metric

This percentage returns the completeness of the dataset and has been adapted from a conference paper by [84] in 2016. It is calculated by dividing the number of recorded y values by the number of expected y values multiplied by 100. The higher the percentage, the more complete the dataset is said to be.

4.1.1.3 Weighted gap percentage metric

If a gap occurs between the 86th and 88th minute, it is rather easy to predict what the missing value for y could be. It could be as simple as taking an average of the y values at the 86th and 88th minute. But if a gap occurs between the 80th and 90th minute, it is rather difficult to predict what the 10 missing values could have been because there are so many possibilities. Figure 4.1 displays two lists, one with 10 gaps of size 1 and the other with 1 gap of size 10. Using the SGPM, the percentage completeness computed for both will be the exact same but this should not be the

case. Even though both the lists contain the same number of gaps, the gap size should matter. The bigger the gap, the poorer the quality of data because it is difficult to predict the values that could have been in the gap as there are so many possibilities. Gap size matters because a bulk of missing points can be hard to fill and it could be difficult to draw meaningful insights in the data analysis phase as an interesting phenomena could have been missed out. The SGPM gives the same result in both cases without considering the impact of gap size on the quality of data. Therefore, this research has devised a new metric which addresses this issue.

$$\begin{aligned} \text{list1} &= [1,3,5,7,9,11,13,15,17,19,21] & (10/21)*100 \\ \text{list2} &= [1,2,3,4,5,6,7,8,9,10,11,21] & (10/21)*100 \end{aligned}$$

Figure 4.1: Percentage completeness computed from lists containing the same number no. of gaps but different gap sizes.

The WGPM operates by giving weights to a gap depending on its size. The greater the gap size, the heavier/bigger the weights (explained in appendix B) and hence, the lower the percentage completeness computed. This metric uses the SGPM but multiplies the result from that metric with a weight score which impacts on the final result. For example, look at figure 4.2. The first three lists (x1, x2 and x3) have 10 gaps in total but of different sizes. The percentage completeness computed from the SGPM remains the same for all three cases (61.54%) but the results computed from the WGPM are different in each case. The first list contains the biggest gaps amongst the others list. It contains gaps of size 6 and 4 and hence gets the lowest % completeness. List 2 gets a higher % completeness than list 1 because it contains 3 gaps of size 2 and 1 gap of size 4. 3 gaps of size 2 have less impact on the quality of data than a single gap of size 6 which is why list 2 gets a higher % completeness. And finally, list 3 gets the highest % completeness because it contains 5 small gaps of size 2. So, list 3 has the smallest gap size and hence, gets the highest % completeness. The results computed for each of the 3 lists using the WGPM are more or less the same because of the weights used. If big weights are used then the difference in results would be big but if small weights are used (as in this case) then the difference in results would be small. And finally, both metrics (SGPM and WGPM) output the same result for a complete list as shown for list 4 in figure 4.2.

It is important to note that this metric is specifically designed to address gaps in time series data because it looks for missing values in time (x). For example, if time has a regular interval of one minute and if a gap between the 80th and 84th minute is suspected, then the metric considers this to be a gap of size 3. Implementation details on how this metric operates are explained in appendix B

4.1.2 Accuracy

Each of the following metrics output results in the form of a percentage. The higher the percentage, the better the accuracy of the dataset.

	Weighted gap % metric	Simple gap % metric
x1 = [1,2,3,4,5,6,7,8,15,20,21,22,23,24,25,26]	49.82	61.54
x2 = [1,2,3,4,5,6,7,8,11,14,17,22,23,24,25,26]	50.09	61.54
x3 = [1,2,3,4,5,6,7,8,9,10,13,16,19,22,25,26]	50.18	61.54
x4 = [1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18]	100%	100%

Figure 4.2: Comparing the results obtained from the WGPM and SGPM.

$$Proportion(m) = \frac{\sum_{i=0}^n A_{n,m}}{n}$$

Figure 4.3: AADM for measuring accuracy of a dataset.
[60]

4.1.2.1 Average absolute distance metric

The following metric has been adapted from a paper by [60] and is most commonly used for measuring the semantic accuracy of a data set. An absolute distance is the difference between the 'true' value and recorded y value in a dataset. The metric calculates an average in which the sum of all absolute distances is divided by the total number of recorded y points as shown in figure 4.3. The smaller the average calculated from the equation above, the better the accuracy of the dataset. However, in the dataset acquired, this has been done a bit differently. Because the 'True' value of y at the depth used is unknown, a mean of values has been taken at intervals of 5. Each mean is then used as the 'True' value for the interval from which it was calculated. Absolute distances between this 'True' value and each value in that interval are taken. This is repeated for all intervals and lastly, an average of all these differences is taken to compute the result as a percentage.

4.1.2.2 Median absolute distance metric

The use of median is a common way of estimating the center of a population. It can be assumed that y values in the middle are more likely to be accurate in comparison to the values at the far end. Hence, median can be used to estimate the accuracy of the dataset. The idea behind this metric has been taken from a book by [73] but has been adapted by this research for making it possible to measure syntactic accuracy. The metric works by firstly calculating the median of the recorded y values. It then adds a range of acceptable values around that median value. So for example, if median is 19.0 and a threshold of 0.2 is chosen, then the range of acceptable values would be all values from 18.8 to 19.2. Each recorded y value is then checked to see if it falls within or outside this range. If it falls outside this range then it is considered to be inaccurate. All the inaccurate values are then divided by the total number of values to compute a percentage of inaccurate values. This percentage is then subtracted by 100 to compute the percentage of accurate values in the dataset. It is important to note that the results obtained from this metric depend on the threshold chosen for the range of acceptable values.

4.1.2.3 Mode absolute distance metric

This is a new metric devised by this research. It works in the exact same way as the MADM but in this case, the mode is used instead of the median. It can be assumed that y values that repeat more often than the others are 'true' values of the data being examined and these true values are more likely to be accurate than the others. Hence, mode can be used to estimate the accuracy of the dataset. This metric measures the syntactic accuracy as well. Because the y values in the dataset are in 3 significant figures, it is very unlikely to find a recurring value (mode). So, values have first been rounded down to 1 significant figures before computing the mode. The rest of the metric functions the same way as the MADM but by using a mode instead of a median. It is important to note that the results obtained from this metric depend on the threshold chosen for the range of acceptable values.

4.1.2.4 Relative error metric

This metric is a measure of uncertainty in the recorded value compared to the size of the recorded value and is often used to report the accuracy of a dataset [88]. Figure 4.4 shows how this is calculated.

$$\text{Relative Error} = \frac{\text{measured value} - \text{expected value}}{\text{expected value}}$$

Figure 4.4: Relative error metric for measuring accuracy of a dataset.
[88]

This equation is usually used to find the relative error of each recorded value but it has been adapted for this research to return the relative error of the whole dataset. Because the expected (True) value is not given, mean values are used instead. The value on the numerator used here is the same as the output of the AADM. The value for denominator used here is the mean of all the recorded y values. The relative error of the dataset is then represented as a percentage.

4.1.3 Precision

The following metrics output results as a percentage where appropriate. The higher the percentage, the higher the precision of the dataset.

4.1.3.1 Standard deviation

S.D is often used to measure the amount of dispersion in a set of values. It is commonly used to measure the precision of a dataset. The higher the S.D, the lower the precision and vice versa [3]. This metric returns the mean and the S.D for the recorded y points. Unlike the other metrics, this one does not provide a precision percentage of the dataset but looking at the S.D value should be enough for data quality experts to know how precise their dataset is.

4.1.3.2 Relative uncertainty

Also known as fractional uncertainty, is a metric often used to calculate the precision of a data set [88]. This equation is usually used to find the relative error of each recorded value but it has been adapted for this research to return the relative uncertainty of the whole dataset. The S.D calculated from the metric above is used as the numerator and the mean of all values is used as the denominator in equation 4.5. The relative uncertainty of the dataset is then represented as a percentage.

$$\text{Relative Uncertainty} = \left| \frac{\text{uncertainty}}{\text{measured quantity}} \right|$$

Figure 4.5: Relative uncertainty metric for measuring precision of a dataset.
[88]

4.1.3.3 Moving precision checker

This is a new metric devised by this research. It has been developed to measure the precision of a dataset by taking a set of surrounding values into consideration when measuring the precision of each data point. A good way of measuring the precision of a dataset would be to consider a collective set of points and see how precise they are instead of considering all the points of a dataset. This is because values of the dataset could change over time so measuring the precision of a small set of points could give more accurate results than measuring the precision of all the points in the dataset altogether. In this metric, for each y value in the dataset, 5 values surrounding it are considered. Each of the surrounding values are given a range decided by a predefined threshold. This is done because the values are recorded in 3 significant figures so it is rare to find completely identical values. Consider a point y. Each time this point falls outside the range of a surrounding value, a counter is incremented. A limit is set to determine whether or not point y is imprecise based on the number held inside the counter. If point y falls outside the range for more than the limit set (3 surrounding values in this case) then it is considered to be imprecise.

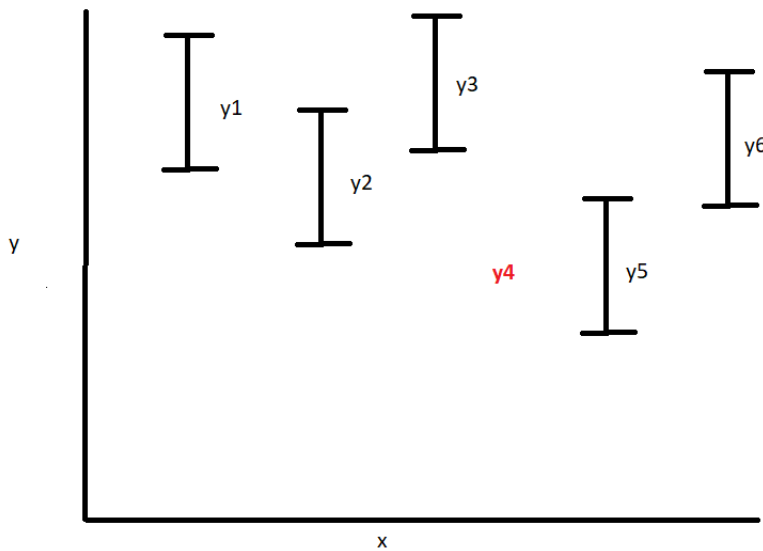


Figure 4.6: Moving precision checker metric explanation.

For better understanding, look at figure 4.6. The value in question is y4 and the surrounding values considered are y1, y2, y3, y5 and y6. Each of the surrounding values are given a range within which if y4 falls, then it is considered 'safe'. Now in this case, y4 falls outside the range for surrounding values y1, y2, y3, and y6. This exceeds the limit and therefore, y4 is considered

to be imprecise. This procedure is repeated for each and every y value in the dataset. To begin with, each value is compared with a set of five consecutive values to the right but in cases where there are not five values to the right, (some or all) values to the left are considered. For example if there are 150 values in total and you are examining the 148th value, then the 148th value will be compared with the 145th, 146th, 147th, 149th, and 150th value. The precision of the dataset is then expressed as a percentage. The user is free to choose the threshold, number of surrounding values considered and the limit.

4.1.4 Consistency

4.1.4.1 Consistency checker

This is a new metric devised by this research. It calculates the consistency of the dataset. It is similar to precision metrics but works slightly differently. A set of points are taken and a local consistency is calculated by computing the absolute differences between successive points. A mean of these absolute differences is then taken and compared to a threshold value. If it exceeds the threshold value then those set of points are considered to be inconsistent. For example consider 3 points, 19.0, 22.0 and 27.0. The calculations would be: $((19-22)+(27-22))/2$. If the resulting value is greater than the threshold, then these set of values are considered as inconsistent. The same procedure is then repeated for the next set of points up until all points of the dataset have been considered. And finally, the consistency of the dataset is displayed as a percentage. The higher the percentage, the more consistent the dataset.

The idea behind this metric is that similar values would be consistent while values that are varying (big differences) would be inconsistent. Predefined thresholds determine which differences are considered consistent which are not. As mentioned in the MPCM, values of the dataset could change over time so measuring the consistency of a small set of points could give more accurate results than measuring the consistency of all the points in the dataset altogether.

4.1.4.2 Systematic consistency checker

This is a new metric devised by this research and its purpose is to look for systematic noise. The checker starts off by collecting potential noisy values from the depth in question (h_1). It does this by calculating the absolute distance for all the points as done in the AADM and by using a threshold value to collect points that lie outside non-noisy bounds. Each potential noisy point at h_1 is compared with points occurring at the same index in the height above (h_2) and below (h_0). When point x_1 is being compared to point x_2 (from h_0) and x_3 (from h_2) at the same index, points x_2 and x_3 are given a range using a threshold value as small as 0.1. This is done because as previously mentioned, values are recorded in 3 significant figures so finding completely identical values is rare. If x_1 falls within the range of x_2 and x_3 , then it is considered to be consistent. This means that points x_1 , x_2 and x_3 are systematic noise because they are repeating at different depths. The metric outputs points that are systematic noise along with a percentage of systematic noise at the depth in question. It is not necessary for systematic noise to appear at all 2000+ depths. Sometimes it may just appear in a set of consecutive depths which is why this metric only outputs the systematic noise occurring at the depth in question.

4.1.5 Metrics relevant to each quality issue

More than one dimension can be used to address a quality issue. For example outliers violate the precision and accuracy of a dataset. To measure the presence of outliers, metrics addressing both these dimensions can be used. The table in figure 4.7 shows which metrics can be used to address each quality issue. Outliers and random noise can be measured using some of the same metrics because it is the threshold value that distinguishes one from the other. For these metrics, a high threshold value can be used to detect outliers alone and a low threshold value can be used to detect noise (and outliers).

Gaps	Outliers	Random noise	Systematic noise
Weighted gaps	Moving precision checker	Consistency checker	Systematic noise checker
Gap ratio	Median and mode absolute distance metric	Average absolute distance metric	
Gap percentage	Standard deviation	Median and mode absolute distance metric	
	Relative uncertainty	Standard deviation	
		Relative error metric	
		Relative uncertainty	

Figure 4.7: Table containing metrics relevant to each quality issue.

4.2 Improvement methods

The following sections outline the different IM implemented for each quality issue. IM for random noise and outliers use data from the DTS_G1_BLEED_1 folder and IM for gaps used data from DTS_G1_BLEED_2 folder. The reason two different folders were used is because bleed 1 contains twice as many data points as bleed 2. Therefore, applying noise and outlier IM to a dataset with more data points will yield more accurate results. Bleed 2 was chosen for gaps because it does not contain any gaps unlike bleed 1. Reason behind choosing a complete dataset for gaps as been discussed in section 3.3.3.2. Time (x) and temperature (y) values used for each IM are obtained from default depth 189.2785m (the word data or dataset used in the following sections mean these x and y values at the default depth). Most of these IM only consider y values when computing results.

4.2.1 Random noise

For each of the following methods, an average uncertainty increase (discussed in section 3.3.3.1) is calculated and a score is outputted onto the console. A visualization showing the improvement

of points is also outputted for each method along with a new list of improved points.

4.2.1.1 Linear regression

Regression analysis is very commonly used for reducing noise in the dataset [85, 19, 32]. A common type of regression is LR. For fitting a regression line, a method of least squares is used. The method calculates the equation of a line that best fits the recorded points by minimising the sum of squares of the vertical deviations from each recorded point to the line. The equation calculated is of form $Y = mx + c$ where x is the independent variable and y is the dependent variable [100]. Using LR for this research, improved points are considered to be the ones that fall on the line and these are obtained by using the equation of the line generated. Each x value is imputed into the equation and the corresponding y value is obtained. Figure 4.8 shows a visualization of the line and improved points.

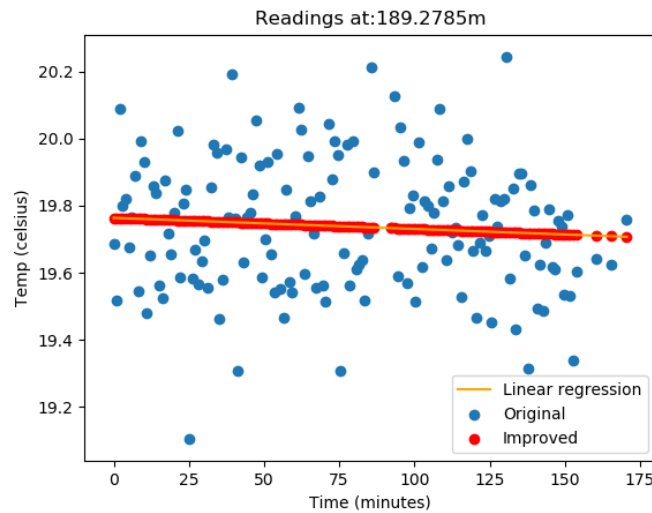


Figure 4.8: Linear regression for reducing noise.

4.2.1.2 Cubic parabolas

A cubic parabola is a form a regression analysis in which the relationship between the independent (x) and dependent (y) variable is modelled as a 3rd degree polynomial [103]. As done with LR, a parabola that best fits the recorded points generates an equation in the form $y = ax^3 + bx^2 + cx + d$. Improved points are considered to be ones that fall on the curve and these can be obtained using the equation of the curve generated. Each x value is imputed into the equation and the corresponding y value is obtained. Figure 4.9 shows a visualization of the curve and improved points.

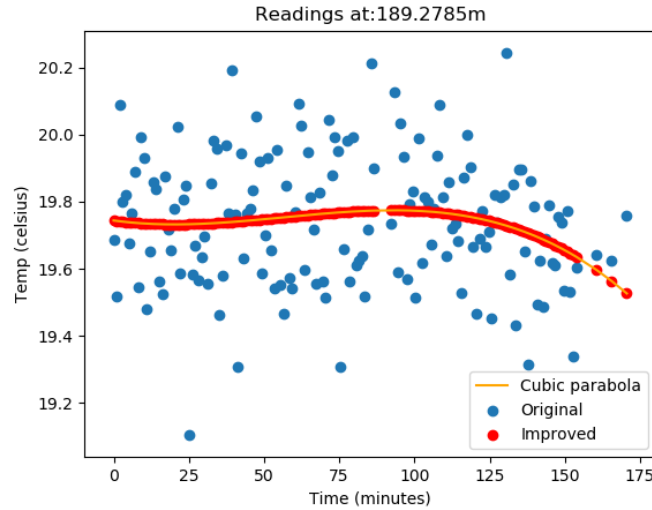


Figure 4.9: Cubic parabola for reducing noise.

4.2.1.3 Cubic splines

Splines are a form of polynomial interpolation. They have been widely used for smoothing data and are often known as smoothing splines [15, 19]. A cubic spline is a spline constructed of piecewise third-order polynomials which pass through a set of N control points [105]. Because the splines are cubic, the equation generated by each spline is the same as the one generated by cubic parabolas. Each spline has an equation of its own which is used to obtain improved y points that fall on it. This is done by imputing x points into the equation and obtaining the corresponding y points. The same is done for all the splines to get a complete list of improved points. The improved points obtained depend on the smoothness and x step size settings chosen for the spline construction. Smoothness of 5 and x step size of 0.1 had been chosen in this case. Figure 4.10 shows a visualization of the splines and improved points.

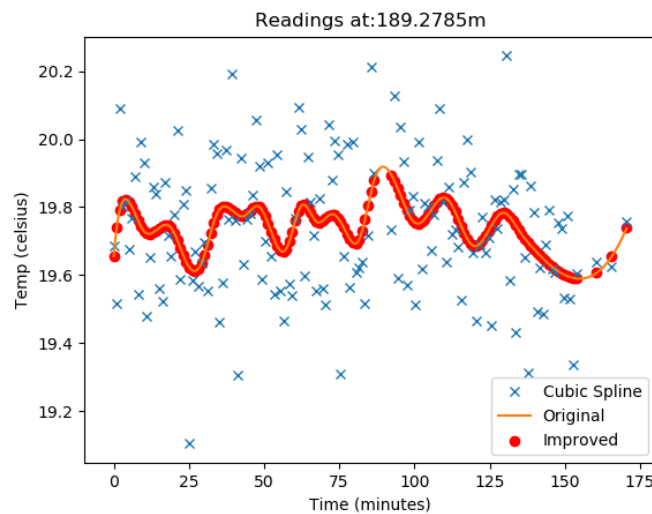


Figure 4.10: Cubic splines for reducing noise.

4.2.1.4 Binning

Binning has also been used for smoothing data [32, 108]. The idea behind binning is to partition data into bins in which each bin contains the same number of data points. Each bin is then smoothed by mean, median or boundary techniques. The approach chosen for this dataset is mean binning with bins of size 5. A mean is calculated from the values in a bin and this mean value is then imputed in place of the original values in that bin. These mean imputed values are now the improved values. The same procedure is done for all the bins to get a complete list of improved values. Figure 4.10 shows a visualization of mean binning and improved points.

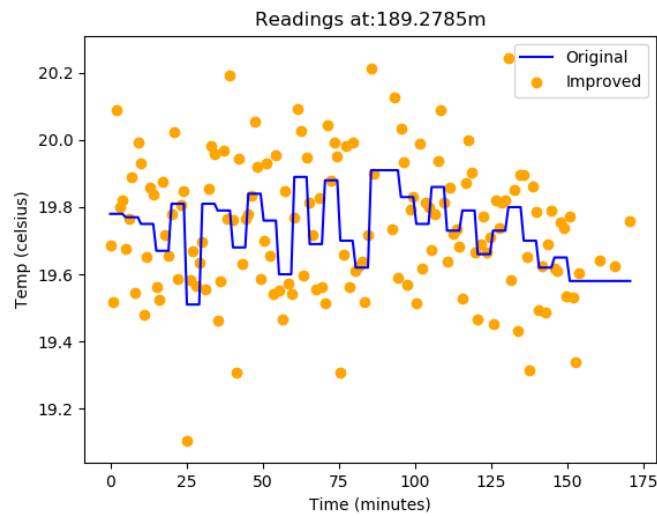


Figure 4.11: Mean binning for reducing noise.

4.2.1.5 Moving average filter

A very popular way of smoothing noise is by applying filters to it using convolution. Convolution is a mathematical operation on two functions (noisy points and filter) that produces an output function (improved points) expressing how the shape of one has been modified by the other [98]. In other words, the purpose of convolution is to smooth a set of noisy data points (fixed array) in time depending on the shape of the filter (moving array) used. One of the filters widely used for this process is the MAF. It operates by averaging a set of points in the input to produce a less noisy output. The filter can be thought as a rectangular window that passes through a set of points, smoothing them [31, 5]. The filter only considers y values when doing the calculations and the output is a set of improved y values. The improved points obtained depend on the window size chosen. In this case, a window of size 10 (points considered for averaging) is chosen. Figure 4.12 shows a visualization of the filter and improved points.

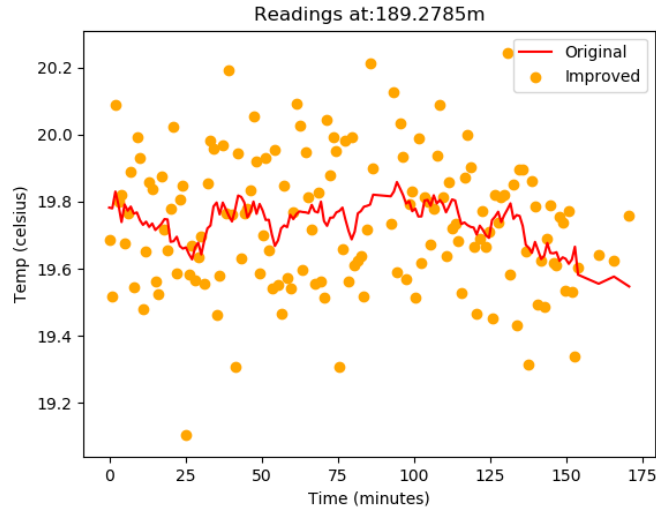


Figure 4.12: Moving average filter for reducing noise.

4.2.1.6 Savitzky-Golay filter

This is another very popular filter used for smoothing a set of data points with the help of convolution [5, 31, 76, 13]. It smoothes a set of points using a low degree polynomial by the method of linear least squares [104]. Like the MAF, this filter also takes a set of noisy y points and returns improved y points. The improved points obtained depend on the window size and order of polynomial chosen. The window size chosen for computing the results in this case is 11 and the order of polynomial chosen is 3. Figure 4.13 shows a visualization of the filter and improved points.

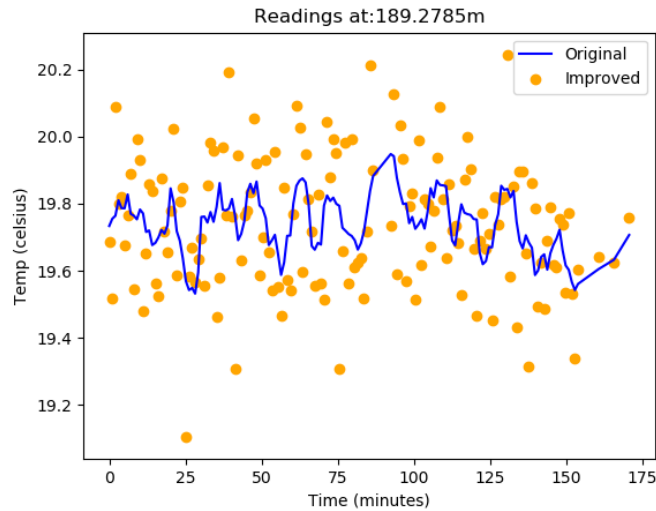


Figure 4.13: Savitzky-Golay filter for reducing noise.

4.2.1.7 Fast Fourier Transform

FFT has been a very popular technique for reducing noise in data with its direct applications being convolution, devolution, optimal filtering, power spectrum estimation and many more [72, 37, 46, 108, 21]. This technique works by expressing a signal in terms of the frequencies that make it up. It shows you what frequencies are present in the signal and in what proportions [70]. The

original signal in the time domain is converted to frequencies in the frequency domain. At this point, a filter is usually applied (convoluted) to remove unwanted frequencies that contribute to noise. Once this is done, the noise free frequencies in the frequency domain are converted back to the time domain resulting into a smooth signal. A limitation with FFT is that it requires the input data to be sampled at regular intervals [72] but the time values in this dataset are not at regular intervals. The FFT function only considers y points but when calculating the frequencies, x needs to be at regular intervals. This is important because it affects the judgement made while removing unwanted frequencies.

Although the dataset used contains a set of points and not a signal, it is still worthy to note how effective FFT is for reducing noise and how it functions. The data points could have been converted into a spline with regular intervals but this would have not been ideal because making a spline out of the data would tamper it and result into a lot of false frequencies making results unreliable. For this dataset, the original set of y points are inputted to the FFT function and the difference between the first two x points are used to calculate the frequency. When a set of y values are inputted, the FFT function creates a signal by interpolating all the points. The FFT function generates FFT values for the positive as well as negative frequencies in which both are mirror images of one another. A graph displaying FFT values and frequencies is then plotted to see which frequencies need to be removed. Upon looking at the graph, an expert from the company suggested that the frequencies between -0.02 to 0.02hz represent the non-noisy/acceptable data points and the rest is just noise. So, a low pass filter was then used to reduce FFT values in a linear fashion after the cut off point (0.02hz). After that, inverse FFT function was used to convert the resulting frequencies back to the time domain resulting into a smooth signal. Figure 4.14 shows the visualizations of this process.

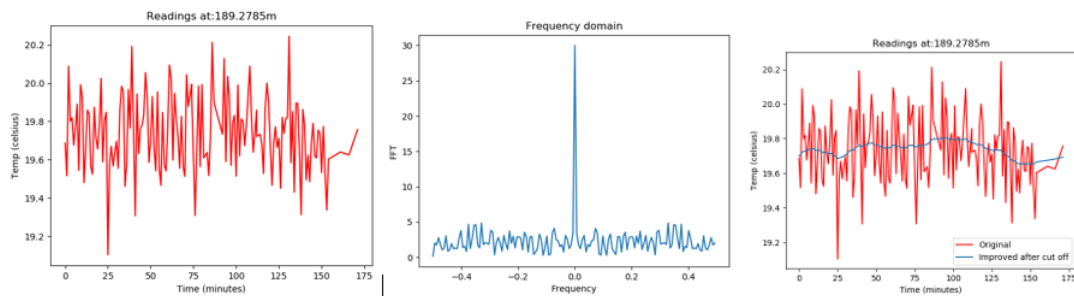


Figure 4.14: Fast Fourier transform for reducing noise. The first image shows the original signal. The second shows the frequencies of the original signal. The third shows the original vs the output signal generated after passing the low pass filter.

4.2.1.8 Kalman filter

Another common technique for noise reduction is the KF [85, 21, 20, 48]. It is a recursive filter that uses a set of equations to estimate the state of a dynamic system from a series of noisy points [85, 89]. It produces this estimate using an average of the system's predicted state and of the new measurement using a weighted average. The result of the weighted average is an estimate of a new state which has a better uncertainty than the predicted and measured state. This process is

then repeated at every step calculating a new estimate at each step. Eventually, the KF learns what the 'True' y value for that depth could be [89, 99]. To use the KF for smoothing data, either the true value can be used as improved values (by imputing it in place of all the original points) or the filtered estimates calculated along the way can be used as the improved values. The second approach has been used in this case because it is difficult to find the point at which the true value was estimated. Figure 4.15 shows a visualization of the filter and improved points. The code for this filter has been taken from SciPy Cookbook [14].

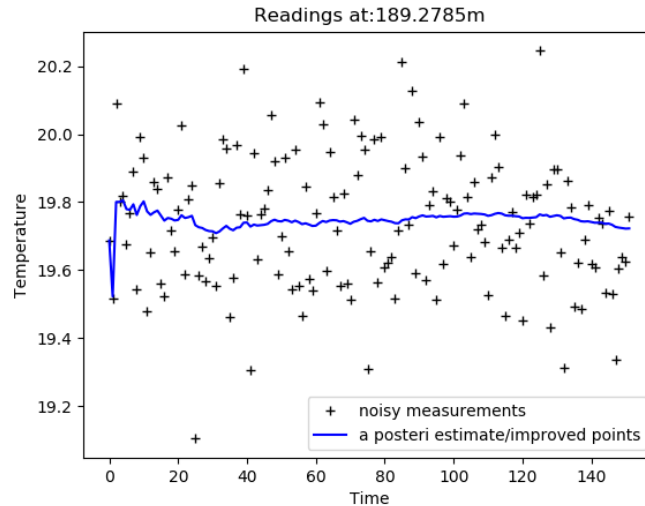


Figure 4.15: The kalman filter for reducing noise.

4.2.2 Gaps

For the dataset with created gaps, a function first collects all the missing x points. This is done via checking the distance between two consecutive x points in the dataset. If the distance exceeds 1.2 minutes that that is considered to be a gap which needs to be filled with estimated values. For example if an x point is 85 and the successive point is 90 then that means that there is a gap between the two points. The function detects this gap and generates a list of x values to fill the gap. In this case, estimated values 86, 87, 88 and 89 will be filled in between 85 and 90 by the function. Let us call these estimated values (86,87,88 and 89) the MissingXvalues as these will be used to predict the corresponding missing y values using the IM below. For each of the following IM, an average absolute distance (discussed in section 3.3.3.2) is calculated and a score is outputted onto the console. A visualization showing the predicted y values is also outputted along with a complete list of x and y values. A complete list (I.e: y) is the original list containing predicted values imputed in places where gaps occurred.

4.2.2.1 Linear regression, cubic parabolas and cubic splines

Regression imputation falls under the umbrella of statistical techniques and is widely used for predicting missing values [85, 42, 26, 11]. In regression imputation, the imputed value is predicted from the regression equation [42]. Although the term regression imputation here refers to LR, cubic parabolas and cubic splines (smoothness: 5, x xstep size: 0.1) can also be used to predict missing values in the same way. The MissingXvalues are imputed into the equations obtained

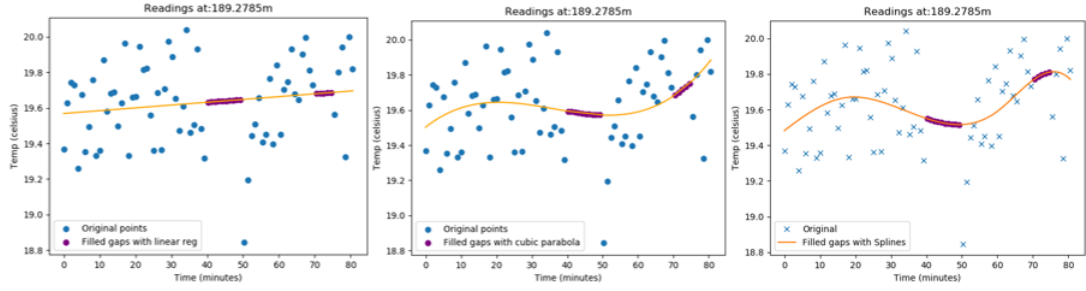


Figure 4.16: Using linear regression, cubic parabola and cubic splines to predict missing values.

from each of these regression techniques (linear, cubic and splines) to predict the corresponding missing y values that lie on the line/curve of the technique used. Figure 4.16 shows visualizations of how the three techniques can be used to predict missing y values.

4.2.2.2 Mean imputation

MI is also a statistical technique commonly used for predicting missing values [43, 108, 42, 29]. It is a very simple technique in which a mean of all the available y values (of the dataset examined) is calculated. This mean value is then imputed in place of missing values. Figure 4.17 shows a visualization of the gaps filled using this technique.

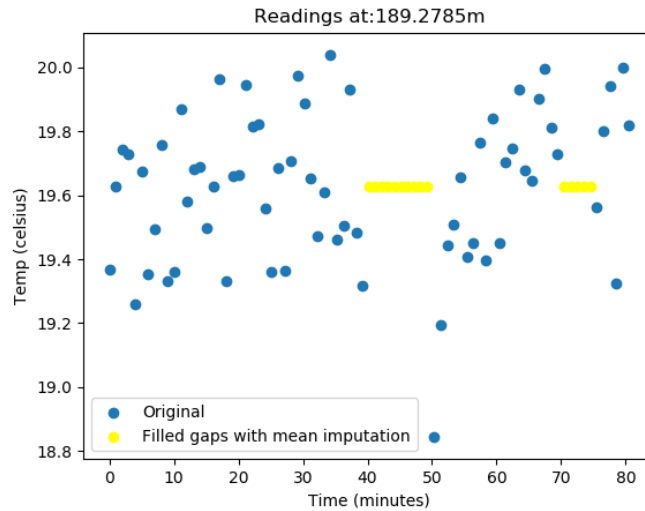


Figure 4.17: Using mean imputation to fill gaps

4.2.2.3 Autoregressive moving average model

This is also a statistical machine learning model often used for predicting missing values [25, 52, 33]. This model is made up of two sub-models: autoregressive (AR) model and the moving average (MA) model. In order to predict the value of the next time step, the AR model learns from a weighted sum of the previous values [97]. The MA model captures serial autocorrelation by expressing the conditional mean of the time series as a function of the past innovations [62]. So ideally, the ARMA models predict missing values (future values) based on the computations made on past values. So suppose there is a list with two gaps: [1,2,3,8,9,10,15]. All the values

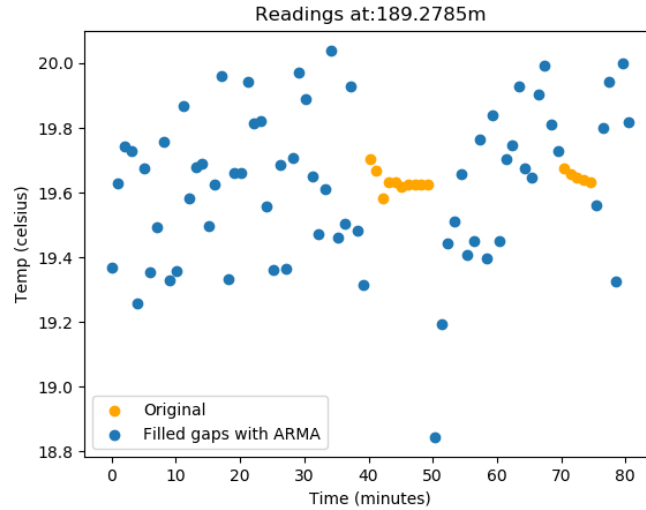


Figure 4.18: Using the ARMA model to fill gaps

before the first gap (1,2 and 3) are fed into the model to predict the missing values between 3 and 8. Then, 1,2,3,8,9 and 10 are fed into the model to predict missing values for the gap between 10 and 15. So to fill missing values for each gap, only the original past values are considered. This is because original values are more reliable than predicted missing values. Figure 4.18 shows a visualization of the filled gaps using ARMA.

4.2.2.4 K nearest neighbours model

This is a non-parametric machine learning model often used to predict missing values [43, 8, 39, 10, 68]. This method selects K cases nearest to the missing point in order to predict what its value could be. These nearest cases are found by using distances such as Euclidean and Manhattan. There are several approaches that can be used to predict the missing value such as taking the mean, median or mode of the K nearest neighbours [10, 68]. In this case, the KNN regression approach has been used. 5 is chosen as the K value which means missing values are predicted by taking the mean of their 5 nearest neighbours. The basic approach to KNN regression uses uniform weights in which of the 5 closest points contribute uniformly to the prediction of the missing point. In other words, KNN uniform assigns equal weights to 5 closest points when making a decision. KNN distance on the other hand assigns weights proportional to the inverse of the distance from the missing point [69]. Both KNN uniform and distance have been used for predicting missing values as seen in figure 4.19. The code has been adapted from SciKit learn [69].

4.2.2.5 Interpolation method

This is a new technique devised by this paper. If a gap occurs between x points 85 and 90, then the corresponding y points can be interpolated to predict the missing y values in the gap. A spline with zero smoothing and x step size of 0.06 has been used to interpolate all the y points. The MissingXvalues are then imputed into respective spline equations to predict the corresponding missing y values. The reason this technique is different to the ones mentioned above is because for example, LR predicts the missing y points by using equation of the line. This line is a least squares estimate of the original points and hence, does not touch each and every point. Whereas this technique does touch each point and uses the interpolation between points to predict missing

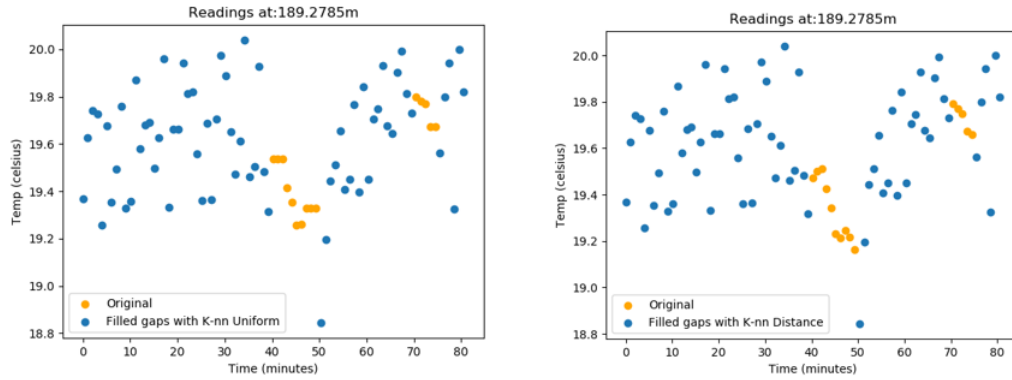


Figure 4.19: Using the K nearest neighbour model to fill gaps

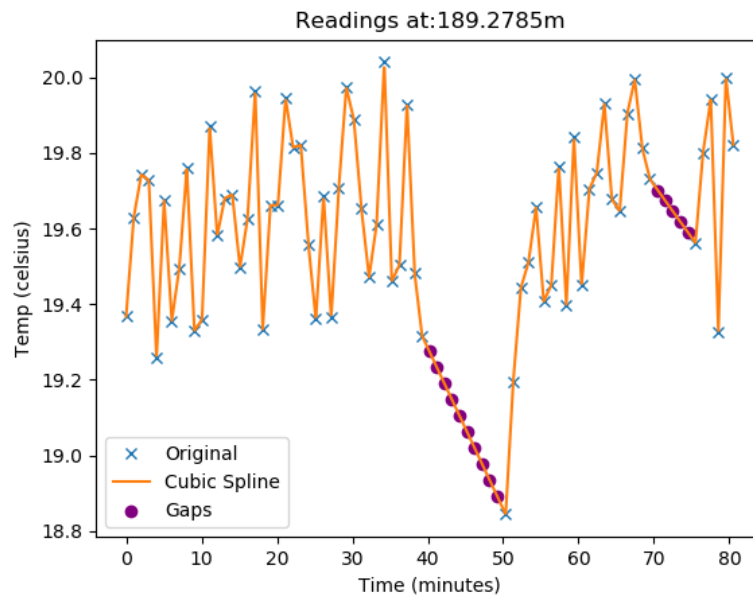


Figure 4.20: Using the interpolation method to predict missing points.

y values that lie in between. Figure 4.20 explains this better.

4.2.3 Outliers

Noise IM can also be used to treat outliers. For example, with LR, an outlier point can be improved by using the equation of the line generated as discussed in section 4.2.1.1. However, treating noise before outliers is not recommended and the reason is discussed in section 5.4. Therefore, the following IM discussed are specific to the detection and removal of outliers. Some of these use thresholds and some do not. For the ones that do, a threshold of 0.5 has been set. As done with noise IM, outlier points could have been improved but in this case, the following outlier methods do not generate equations or have computations that would determine their improved position. Therefore, outliers can only be discarded. Discarding them would make the x and y lists incomplete but gap techniques could be applied to fill the missing points. Each of the following methods output a new x and y list without outliers, a visualization of the outlier points and a list of outliers.

4.2.3.1 K-means clustering

Clusters are a set of groups that contain observations that are similar to one another. Similarity of observations is quantified using a distance measure. Different distance measures are used by different clustering techniques to form clusters. The technique used by this research is KMC which is based on pairwise euclidean distances. It is one of the most famous unsupervised machine learning techniques used to detect and remove outliers [17, 1, 82, 108, 44]. Usually, the values in the smallest cluster are classed as outliers but this may not always be applicable to time series data so thresholds are used to determine outlier values instead [17]. In this approach, the data points are grouped into 4 clusters with each cluster having its own center point. For a point x_1 in a cluster, a distance between x_1 and the center point of that cluster is measured. If this distance exceeds a threshold value then x_1 is considered to be an outlier. The same process is then repeated for all points in each cluster. Bits of code used has been adapted from the python data science handbook [90]. Figure 4.21 shows a visualization of the outliers detected using KMC.

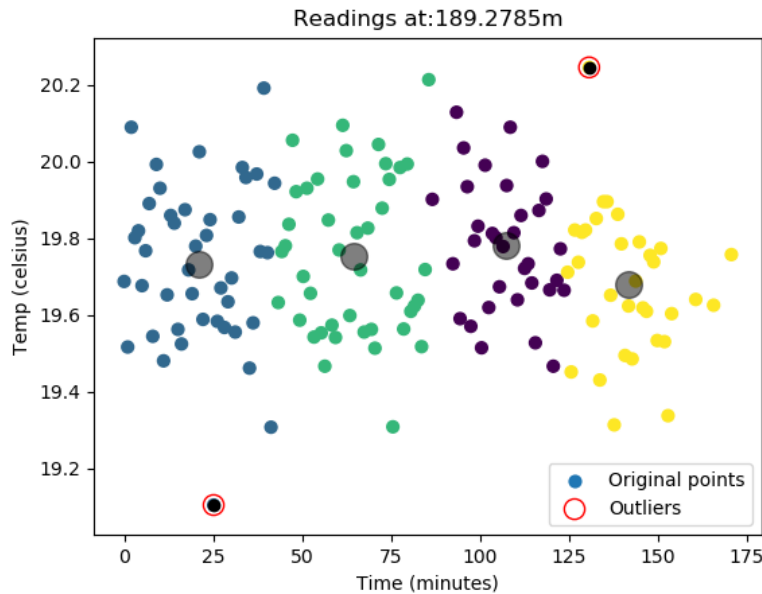


Figure 4.21: K-means clustering for detecting and removing outliers.

4.2.3.2 Two-sided median method

This approach had been proposed by [7] in 2005. For each data point, this approach calculates the median of a neighbourhood of points. This median value is then used to determine whether the point in question is an outlier. A neighbourhood window of size 10 has been considered in this case. For the first set of points, 10 neighbors to the right are considered and for the last set of points, 10 neighbors to the left are considered. For all the other points, 5 neighbors to the left and 5 neighbors to the right are considered. For a point x_1 , a median value is calculated from the neighbours considered and this median value is then compared to x_1 . If the difference between the two values exceeds the threshold, then x_1 is considered an outlier. The same process is then repeated for all points in the dataset. Figure 4.22 shows how this works.

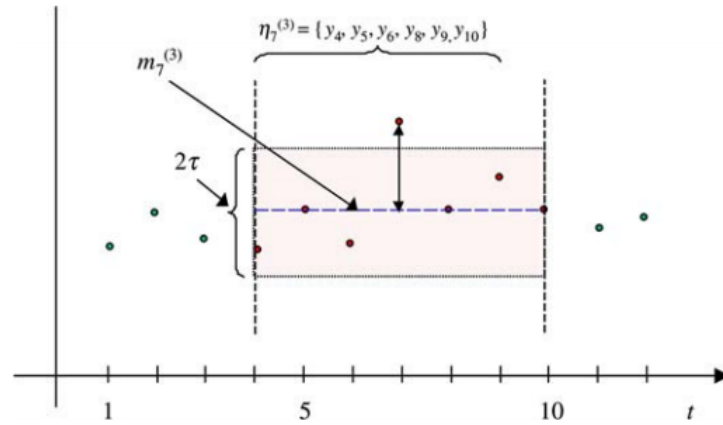


Figure 4.22: Two-sided median method for detecting and removing outliers.

[7]

4.2.3.3 Interquartile range

IQR is a statistical method commonly used to detect and remove outliers. It is the difference between the third (Q3) and the first (Q1) quartile calculated from a list of values. All values above $Q3 + 1.5 \cdot IQR$ (upper limit) and below $Q1 - 1.5 \cdot IQR$ (lower limit) are considered as outliers as this roughly corresponds to three S.D (3 sigma) from the mean value [65, 107, 73].

4.2.3.4 Histograms

A histogram shows you the shape of distribution of the recorded points and is commonly used for detecting and removing outliers [110, 111, 79, 65]. It divides the values of the dataset into bins (ranges) and then shows the frequency of values in each bin. Outliers are values that occur infrequently and are distant from majority of the recorded points. Looking at figure 4.23, the value in the far left bin can be considered as an outlier. The histogram function used returns a visualization as well as the number of elements in each bin. Bins that contain one element and lie far from the rest of the bins are chosen. The reason bins with one element are chosen is because outlier values are unique and do not repeat. The value in each chosen bin is then retrieved and classed as an outlier.

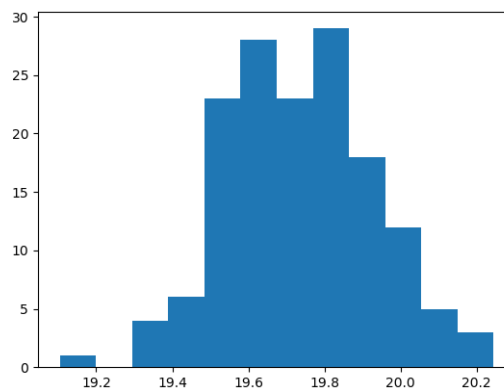


Figure 4.23: Using Histograms to spot and remove outliers.

4.2.3.5 Mean intervals method

This technique has been devised by this paper and is same to the systematic consistency checker (section 4.1.4.2) but uses slightly different threshold values. Potential outlier x_1 (at h_1) is considered an outlier only if a similar value does **not** exist at h_0 or h_2 at the same index. The idea behind this metric is to effectively collect outliers by making sure that the potential outlier values obtained are not systematic noise instead.

4.2.3.6 Moving precision checker

This technique has been discussed in section 4.1.3.3. All the imprecise points obtained from this technique are outputted as outliers.

4.3 Software testing

As mentioned in section 3.1, the purpose of this research is not focused on producing a substantial codebase but rather on the meaningful results drawn from the concept-proofing code implemented. Therefore, testing the software was not necessary.

Chapter 5

Results and evaluation

The following sections evaluate the results obtained from the IM for each quality issue. This evaluation aims to prove the hypothesis by meeting the main objective of this research.

5.1 Noise

Most of the following sub-sections evaluate random noise besides the last one which provides recommendations for SN.

5.1.1 Deciding between preserving the accuracy of the original points and maximum noise improvement

This is the main recommendation when considering IM for random noise. The context in this case is an organization's priority when making a choice between preserving the accuracy of the original points and maximum noise improvement. A statistical t-test for evaluating results is not applicable in this case because each IM is unique and the results obtained from each is independent from the other. So, if a t-test was used, the results drawn from it would be inapplicable to other IM and also because the sample size is too small to receive reliable results. Table in figure 5.1 shows the average uncertainty increase and S.D of the improved points obtained from each IM. The average uncertainty increase measures the accuracy of the original points. The lower the uncertainty score, the smaller the loss in accuracy of the original points (reason explained in section 3.3.3.1). For this research, the S.D is used to measure the improvement in noise. The lower the S.D score, the better the improvement in noise. This is because, temperature values at a given depth are supposed to be more or less the same and would hence have a low variation. So an ideal improvement in noise would be to reduce the S.D of the points as much as possible.

It is important to note that all these recommendations are subject to change depending on the

	Average uncertainty increase	Standard deviation
Linear regression	0.148	0.015
Cubic parabola	0.147	0.039
Moving average filter	0.144	0.061
Kalman filter	0.143	0.026
Fast Fourier Transform	0.141	0.041
Cubic splines	0.135	0.066
Savitzky-Golay filter	0.133	0.086
Binning	0.127	0.098

Figure 5.1: Results for noise improvement methods

dataset used and the settings used for each IM but the main thing to understand is the IM chosen based on priorities. Looking at the results in figure 5.1, there seems to be a trade-off between the average uncertainty increase and the S.D. The two seem to be roughly inversely proportional to one another. For example, when LR is applied to the original points, the average uncertainty increase is 0.148 and the S.D is 0.015. Binning on the other hand has a lower uncertainty increase (0.127) but a higher S.D (0.098) than LR. But in rare cases such as the MAF both uncertainty increase and S.D are higher than the rest of the methods considered.

As mentioned in section 3.3.3.1, when moving from a noisy dataset to a less-noisy one with the use of IM, there is a loss in accuracy of the original points as their uncertainty increases. This is important to understand when decisions need to be made on choosing the best IM. The choice of IM depends on an organization's priority. If preserving the accuracy of the original points is the main priority then the best technique would be binning in this case since it has the lowest uncertainty change. If maximum noise improvement is the main priority then the best technique would be LR in this case as it has the lowest S.D. If the priority is to have a balance between preserving accuracy and noise improvement then the KF would be best because it has a medium (not too high, not too low) uncertainty increase in comparison to the other methods and has the second lowest S.D score. Thus, the choice of the best noise IM does indeed depend on the context such as priorities set by organisations,

5.1.2 Deciding between saving storage costs and effective noise removal

The context in this case is an organization's priority when making a choice between saving storage cost and effective noise removal. Equations (I.e: $y = mx+c$) obtained from techniques such as LR and cubic parabolas can be stored instead of having to store all the improved points. When needed, improved y points can be easily retrieved by imputing the original x values into the equations stored. Datasets are usually big (often in TB) so storing only the equations saves on storage space and costs

But on the other hand, some techniques do not generate equations but are better at removing noise effectively. Upon understanding the functionality of each noise IM, FFT has proven to be most effective at removing noise. This is because it lets you remove all the unwanted frequencies that contribute to noise. This leads to an effective removal of noise because only the noisy data points are removed and the non-noisy data points are untouched. LR improves noise but does not guarantee only removing noisy values. It could also remove some non-noisy ones because unlike FFT, it does not have the option to only remove noisy points. Hence, FFT is better at removing noise effectively but the downside is that it does not generate equations. This means that all the improved points would need to be stored which leads to increased storage costs and space. So, there is a trade-off between saving storage costs and effective noise removal. Thus, once again, choosing the best IM depends on the context such as priorities set by organisations.

5.1.3 Deciding between wanting a rough trend of the data points and observing data patterns.

The context in this case is whether the organisation's priority is to obtain a rough trend of the dataset or is interested to observe data patterns. Improved points obtained from techniques such as LR and cubic parabolas show a rough trend of the data which summarize the overall behaviour

of the dataset for example whether the temperature values are increasing, decreasing or relatively stationary (first picture in figure 5.2). This would make data interpretation quicker in the analysis phrase. But on the downside, interesting phenomenon such as a sudden increase or decrease in values occurring at a particular time can be missed out. Therefore, preserving noise in some cases can be useful because it can show you some interesting patterns in your data (second picture in figure 5.2). These patterns could be useful in the analysis phrase as they could unlock some meaningful phenomenon. Improved points obtained from most other techniques such as the convolution filters for example show these data patterns and how they change along the way but on the downside, they do not show you a rough trend of your data. So once again, the choice of the best IM depends on the context such as the priorities set by organisations.

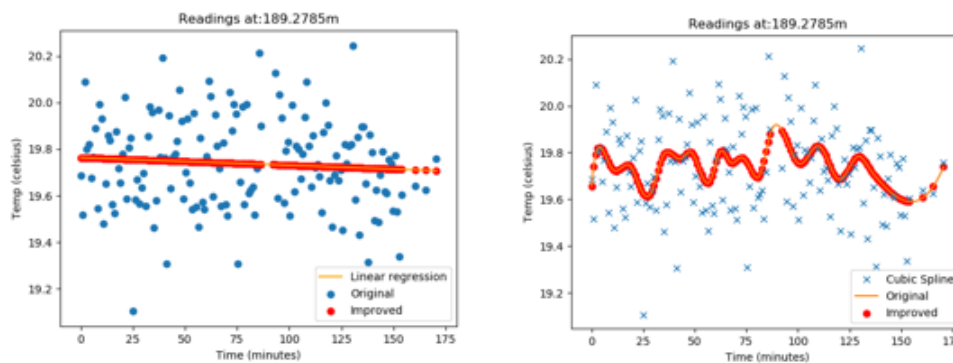


Figure 5.2: Technique showing a rough trend vs technique preserving some noise

5.1.4 Choosing a technique based on the natural behaviour of the data points

It is important to choose a technique based on the natural behaviour of the data so that the data points can be fitted well. For example, if the behaviour of your data points is increasing and then decreasing and you want to reduce noise while preserving this pattern then a cubic parabola should be used. But if your data points show some sort of correlation then LR should be used. If data points are subject to change at regular intervals of 5 for example, then mean binning can be used because it would preserve this data pattern of change. So, the choice of best IM depends on the context which in this case is the kind of data being dealt with. Techniques that better fit the shape of the data should be used because in this way, trends or patterns are preserved and only the noisy data points are removed which also results to a minimal loss of accuracy.

5.1.5 Choosing a technique based on noise considerations

It is important to understand what is considered as noise and what is not. Establishing what band of values are considered as acceptable/non-noisy can also help determine which method to choose. For example, in the first visualization in figure 5.3, the band (the two black lines) of non-noisy values is small. Everything else beyond the two black lines is considered as noise. So in cases where the band is small, techniques such as LR should be used. But in scenarios such as the second visualization in figure 5.3 where the band of non-noisy values chosen is big, techniques such as splines (with suitable smoothing) or convolution filters (with suitable settings) could be used as they would cover most of the points. So it is important to have a knowledge of what is considered

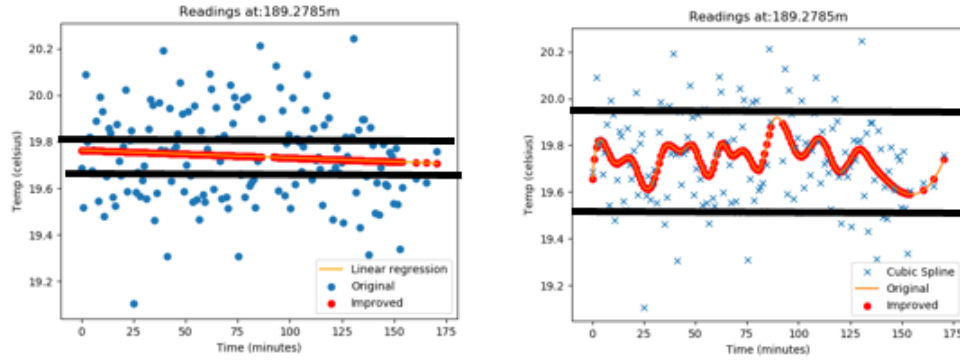


Figure 5.3: Bands showing what is considered as non-noisy.

as noise in a dataset. This would help choose the most suitable method that best covers most of the needed/non-noisy data points and filters the rest. The context in this case is what an organisation considers as noise and hence, the choice of best IM depends on this.

5.1.6 Recommendations on systematic noise

Decisions on SN need to be made before applying any of the noise IM. Relevant metrics tell you whether or not there is SN in your data. If there is SN then decisions need to be made on whether to keep or discard it. If the decision is to discard it then it should be discarded from **all** depths (at which it occurs) in order to maintain consistency. It can be discarded from each depth by applying noise IM. If the decision is to keep the SN then it should be kept in **all** depths (at which it occurs) to maintain consistency. So when noise IM are applied, the SN components should be left untouched for all depths. So whether the decision is to keep or discard SN, it is important that it is treated the same for all depths at which it occurs

5.2 Outliers

In order to detect and remove outliers, it is important to have a knowledge of the range of acceptable values outside which points would be considered as outliers. For example if a true value of 19.0 was given and if it was said that all points outside ± 0.5 of this value are considered outliers then it's easy to detect and remove them. But this information is never provided in real-world datasets which makes it difficult to remove and detect outliers because then you do not actually know whether outlier techniques have removed outliers or noisy values instead. So, unlike the case with noise and gap IM, there is no way of concluding which outlier technique is better than the other because no information is provided on what is considered as an outlier. If a list of potential outlier values was provided than this could have been compared with the results obtained but this is not provided either.

So, the only option was to pick a threshold that best suits the dataset. After carefully examining visualizations, a threshold value of 0.5 was chosen. This threshold value is considered high for this dataset and it has been chosen because the higher the threshold value, the better the outlier detection [59]. This value is used by all techniques that require a threshold to function so that results can be compared on similar grounds. Based on this, potential outliers obtained from each

	Outliers
K means clustering	19.104, 20.245
Two-sided median	19.104, 19.308, 20.213
Interquartile range	19.104
Histogram	19.104
Moving precision checker	19.104
Mean intervals method	19.104, 20.245

Figure 5.4: Table containing results obtained from different outlier techniques.

of the outlier IM is shown in the table in figure 5.4. The performance of the techniques is measured based on the occurrence of an outlier value throughout all techniques. The most reoccurring outlier value seems to be 19.104 and the second most reoccurring value seems to be 20.245. So this means that 19.104 is most likely to be an outlier and 20.245 seems to be second most likely. So based on this, the IQR, histograms and the MPC seem to be the best outlier techniques. The second best techniques seem to be KMC and the mean intervals method. And the least efficient technique seems to be the two-sided median because it has detected values which have not been detected by other techniques.

It is important to understand that the results would change depending on the threshold values chosen. A high threshold values is more likely to detect real outlier values. But this would not always be chosen by organisations as the choice of threshold value depends on what they consider as outliers based on their data requirements. If a low threshold was chosen for this experiment then many different potential outlier values would have been returned by each IM because each method operates differently. But this was not chosen because there is no way of telling whether the values obtained are outliers or random noise instead. So it would have been difficult to tell which techniques are better than the others if a low threshold was chosen.

Under the settings used for this experiment, some techniques have been proven to be better than the others but this would not always be the case. The choice of the best outlier technique depends on the thresholds set by organisations which determine which values are considered as outliers. For example the IQR is a well known technique used for outlier detection. It classes values that are 3 S.D away from the mean as outliers. But what if a company classes their outliers to be values that are 2 S.D away from the mean? Then in this case, the IQR would not be the best technique for that organisation as it would not capture all the outliers. So although IQR would be the best technique for some organisations, it may not be the best technique for other organisations. So, not many recommendations can be provided indicating which method to use under what conditions because the choice of best outlier technique depends on the context which is the data requirements (threshold value) set by organisations.

5.3 Gaps

From figure 5.5, it can be concluded that cubic splines are parabolas are the best IM when it comes to filling gaps because they have the lowest average absolute score. But it is important to understand that this will not always be the case. The choice of the best gap technique depends on different factors which will be discussed in the following sections.

	Average absolute distance
Interpolation	0.389
K nearest neighbour distance	0.229
K nearest neighbour uniform	0.194
Mean imputation	0.166
Autoregressive moving average	0.164
Linear regression	0.157
Cubic parabola	0.146
Cubic splines	0.146

Figure 5.5: Table containing results obtained from different gaps techniques.

5.3.1 Choosing a technique based on the visual appearance of data

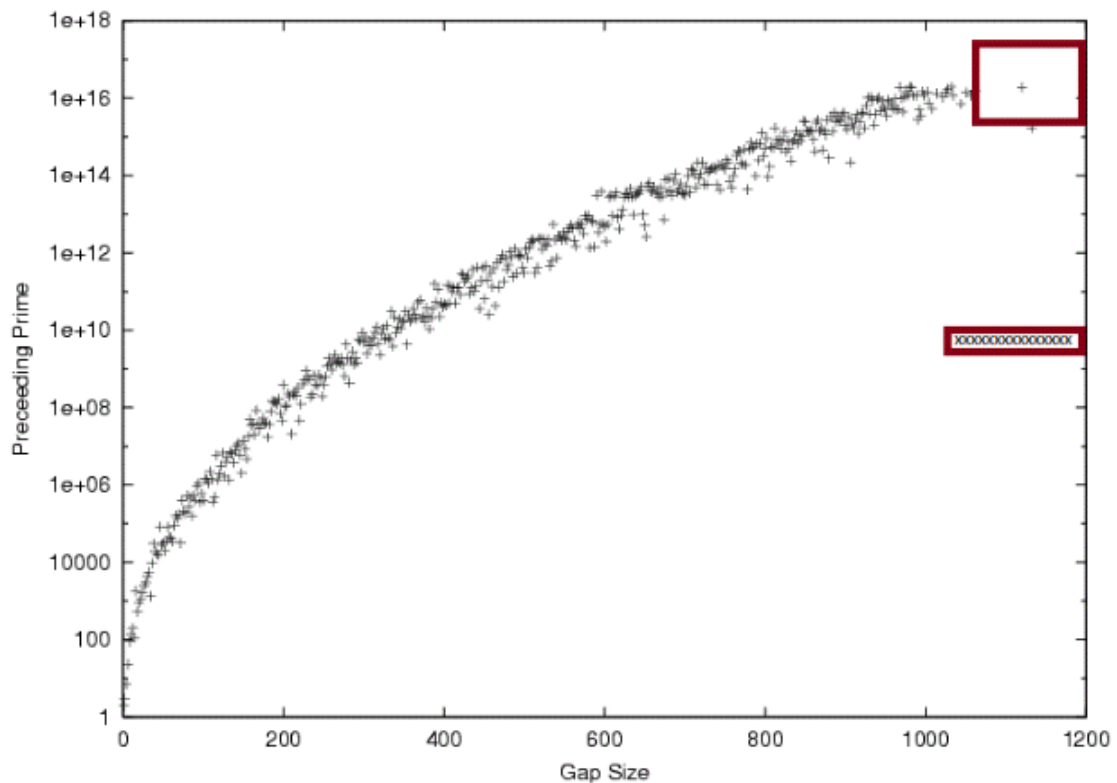


Figure 5.6: Choosing a technique based on the visual appearance of data

[36]

It is important to choose a technique based on the visual appearance of your data so that the predicted values imputed can be inline with rest of the data points. For example, consider a set of data points with an increasing trend as shown in figure 5.6. To efficiently fill the gap in the big red box, you could use ARMA, KNN, cubic parabola, spline, or the interpolation method. LR could also be used but it is not recommended in this case because figure 5.6 is curve and not a straight line. But what definitely can not be used in this case is MI. MI would impute values based on the mean which seems to be roughly around $1e+10$. The predicted values imputed using this technique are shown in the little red box. The reason MI is not recommended in this case is because it breaks the overall trend/shape of the data points as you can see in figure 5.6. This is why it is important to acknowledge the visual appearance of the dataset so that a method can be chosen based on how

well it fits and preserves the data trend if there is any. The dataset chosen for this project does not have a trend of any sort. The data points are rather dispersed/random which makes predicting missing points difficult and results unreliable. With random data you can use any method because there is no way of predicting what the real values could have been. The results in figure 5.5 are specific to this dataset and are subject to change if a different dataset is used. But the key thing to understand is that the choice of best gap IM depends on the context which is the visual appearance of the data in this case. If the data is random, any IM can be used but if the data is trendy then IM that best fit the shape of the data should be used.

5.3.2 Deciding between fast and effective techniques

In most cases, there is a trade-off between how fast vs how effective a technique is. The choice depends on company requirements. For example, MI is fast because it computes the mean and simply imputes that for all the missing values. Multiple imputation on the other hand replaces each missing value with a set of plausible ones by filling the missing data N times to generate N complete lists which are analyzed using S.D. The results from the M complete data sets are then combined to draw an inference on the plausible missing values [26]. So, multiple imputation is more effective than MI because it allows appropriate assessment of the imputation uncertainty whereas MI does not [26, 43, 27]. Hence, Little and Rubin (2002) concluded that MI is inferior when compared to multiple imputation [26]. But on the downside, multiple imputation can be labor intensive since there are not many stats packages available to do it which makes it time consuming [28]. So specific to this example, If a company wishes to quickly impute missing values then the best method would be MI but if they wish to impute values effectively, then the best method would be multiple imputation. Therefore the choice of best IM depends on the context which is an organisation's priority in this case.

5.3.3 Choosing a method based on the behaviour of the data points

It is important to take the behaviour of the data points into account when choosing the best gap technique. For example, as suggested in the result section for noise, if for whatever reason the dataset used is subject to change at regular intervals (not the case for this dataset), then it is important to choose a technique that preserves this pattern of change. For example consider the dataset shown in figure 5.7. In this, the temperature recorded seems to change every 15 minutes. For the first interval, the values are roughly between 19.1 and 19.2 degrees. And for the fourth interval they are roughly between 19.9 and 20 degrees. Both these intervals have a gap as highlighted in the red boxes. To effectively fill these gaps, techniques such as KNN and the interpolation method should be used. KNN would predict a missing value by considering 5 ($K = 5$) values surrounding it. The interpolation method would only consider the value before and after the gap in order to predict a value between the two. The predicted values obtained from both the methods would be closely related to other values in the same intervals. This is because when making a prediction, these techniques only consider values close to the missing value instead of considering the entire dataset. By using these techniques, patterns in each interval are not disrupted or affected by patterns in other intervals.

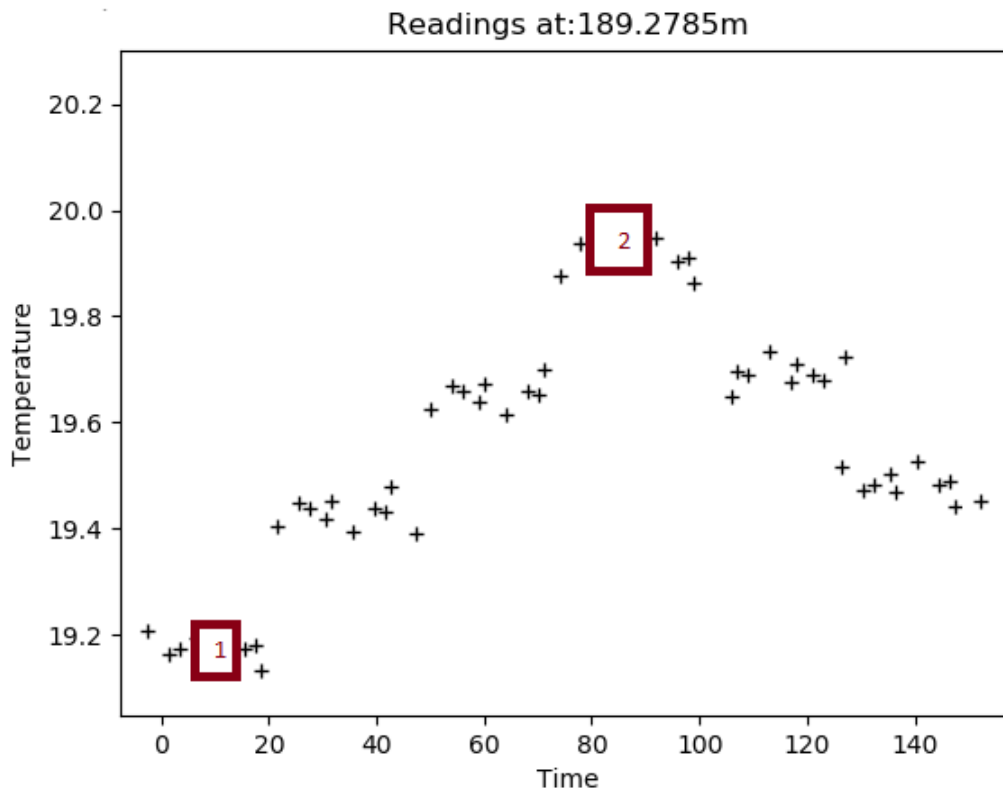


Figure 5.7: Choosing a method based on the behaviour of the data points

Using MI, LR, cubic splines or parabolas in this scenario would have disrupted the pattern observed. ARMA would have correctly predicted the missing values in the first interval but not the second since it considers all the past values when making a prediction. Therefore, it is important to carefully examine the behaviour of the data points to look for any patterns they may have and to then choose an IM that would not disrupt these patterns. So the context, which is the the behavior of the data points in this case, needs to be considered in order to choose the best gap IM.

5.3.4 Choosing a method based on data expectations

The method chosen also depends on what is expected out of the data. In cases such as the dataset acquired, most values are random and spread all over the place. But as previously mentioned, temperature values at a given depth are supposed to be more or less the same. So in this case, to avoid imputing a set of random values, methods like LR, cubic parabola, or MI can be used in order to introduce stability in the missing values imputed. But if techniques such as KNN or ARMA are used then the missing values imputed will be also random as these methods consider surrounding values which are random. The context in this case is data expectations and the choice of best IM depends on this.

5.4 Recommended order in which these quality issues should be treated

In order to improve the quality of data effectively, this research recommends to treat the outliers first. Treating outliers, by either removing or improving them, before applying noise or gap techniques would yield into better quality of data. If for example noise improvement is applied first then the outlier points will affect the improved values generated. Consider LR for example. With the presence of outliers, the equation of the line generated would be different to the one generated without outliers because it would also consider outlier points when computing the equation. This would affect improved values obtained using the equation. Also, if noise improvement is applied first then the outlier points would move to a better position but they would still be a little further away from the rest of the improved points which again affects data quality. Similarly, If gap techniques are applied first then the estimates of the missing values would be affected by the presence of outliers. Therefore, to effectively improve the quality of data, the first step should be to treat outliers and then apply noise or gap IM. It can be outliers -> noise -> gap or outliers -> gaps -> noise. The order with noise and gaps IM depends on company requirements.

Chapter 6

Discussion and Future work

6.1 Discussion

Which is the best IM for addressing data quality issues: noise, gaps and outliers? The hypothesis claimed that this is context dependent which means that the choice of best IM depends on an organisation's priorities, requirements and the dataset they're working with. Chapter 5 evaluated the results obtained from each quality issue along with providing a further set of recommendations that could help an organisation choose the best IM given a context. As seen with noise, there is a trade off between preserving the accuracy of the original points and maximising noise improvement. In order to choose the best IM in this case, organisations would have to prioritize one of these two. If preserving accuracy of the original points is a priority, then the best IM would be the one with the lowest average uncertainty increase which is binning for the dataset examined. If maximizing improvement of noise is a priority then the best IM would be the one with the lowest standard deviation which is LR for the dataset examined. So in this case, the choice of best noise IM depends on an organisation's priorities. For outliers, the better techniques seem to be IQR, histograms and the MPC but this would not always be the case as it depends on the threshold value set by an organisation based on their requirements. Hence, the choice of best outlier IM depends on an organisation's requirements. And for gaps, the best IM seems to be cubic splines and parabolas but this would not always be the case as the results would be different for data with trends, patterns or expectations. Hence, the choice of best gap IM depends on the dataset acquired and the organisation's requirements. The analysis confirms that the choice of best IM does indeed depend on the context in which the data is being examined and therefore, I accept my hypothesis.

The reason this research is important is because as seen in section 2.3, data analysis is beneficial to O&G companies in numerous ways. In order to get the best possible insights from the analytical process, it is important that the data being examined is of best quality and this is only possible if the best IM are applied to improve it. The choice of best IM depends on the context and if organisations are not provided with an in-depth understanding of this then they may choose a technique which may not necessarily be the best choice. Therefore, a set of recommendations have been provided to help those in the field of O&G pick the best IM given the context. No research has been conducted in the past to provide a thorough understanding of the different factors (context) that contribute towards the decision-making process when choosing the best IM. Therefore, these recommendations/results contribute to a clear understanding of that.

Most of the limitations are with the dataset acquired than the execution of the research itself. To begin with, the dataset acquired is not very noisy. This is why there are not any major differences between the average uncertainty increase values obtained by each IM. They are all relatively similar and this makes it difficult to strengthen the argument made on the existence of a accuracy vs improvement trade-off. The true temperature values at each depth were not provided which made it difficult to understand what was considered as noise and how much of it was considered acceptable. Because knowing how much noise is considered acceptable informs how much cleaning needs to be done and it also tells you which improvement method performed better (if S.D was not used to measure this). The time recorded was not in regular intervals and this is a requirement for some IM like FFT as explained in section 4.2.1.7. Not having regular intervals can make results obtained from the FFT a bit unreliable. Results obtained from gap IM are limited to the dataset used because the data points are rather random. It was also difficult to confidently tell which outlier techniques were better than the others because knowledge of outliers was not provided. If a list of outliers at a given depth was provided then results could have been compared but this was not the case.

So ideally, all the results obtained are limited to the dataset used. But what needs to be understood is that this dataset has been acquired from a real O&G company so it represents the type of data dealt with in the real world. Therefore in reality, no knowledge of true values or outliers will be provided for each depth in a dataset. And in most cases, data values will be random and will not have any trends. So results obtained from noise, gaps and outlier IM will always be different no matter what. They would be different because of the dataset used, company requirements set, and the type of settings used for some IM.

6.2 Future work

This research can be extended to provide recommendations for the same quality issues using acoustic datasets. Acoustic data is also very widely used in O&G industries but because of time constraints, this research was not able to cover it. Methods like FFT are designed for acoustic signals and hence, FFT results would be more reliable. The research can also be extended to explore data with different trends and patterns. Although the research provides recommendations on how to deal with this kind of data, real experiments on such data can validate these recommendations and make them more reliable. The research was not able to implement the 'should have' specified in section 3.1 because of time constraints. It would be helpful to have these functionalities so that the improved points generated can be of use to the user.

Chapter 7

Conclusion

From the past two decades, O&G companies have spent a lot of money on data because they have come to realize its importance and the potential it possesses in improving their operational efficiency [64]. But in order to get meaningful inferences from data analysis, the data needs to be of good quality. O&G companies use numerical time series data (Log ASCII standard) for well logging. The quality issues faced by this kind of data are noise (random and systematic), gaps and outliers. It is important to transform the quality of data by improving the existence of these key issues. In order to achieve the best data quality, it is important to choose the best IM. But which is the best IM for each of the quality issues mentioned? The hypothesis claimed that this is context dependent which means that the choice of best IM depends on an organisation's priorities, requirements and the dataset they're working with. The main objective of this research was to critically evaluate the IM for each data quality issue in order to understand which ones are best depending on the context in which they are being examined. The purpose of doing this was to provide a set of meaningful recommendations that could help those in the field of O&G pick the best IM given the context. The other objectives were to walk the reader through the data quality process (DQF) and to present a set of metrics used to assess data quality.

The process and components of the DQF had been thoroughly touched upon. A set of dimensions relevant to the research had been defined and a set of corresponding metrics to measure each dimension had been devised by this paper. These were presented along with the exploration of other relevant metrics currently used by industries. This would be of interest to those in charge of assessing data quality because it would provide them with a variety of metrics to choose from depending on how they want to measure their data and what they want their outcomes to be.

After thorough research, a set of relevant IM for each quality issue had been implemented. The results obtained from these were then thoroughly evaluated to prove the hypothesis. In doing so, the main objective of the research had been met. The evaluation proved that the choice of best IM does indeed depend on the context in which data quality is being examined and therefore, the hypothesis was accepted. No research has been conducted in the past to provide a thorough understanding of the different factors (context) that contribute towards the decision making process when choosing the best IM. Therefore, the results obtained contribute to a clear understanding of that by providing a set of meaningful recommendations to help O&G companies choose the best technique based on their priorities, requirements, and the type data used. The research did have a few limitations and these have been provided in chapter 6 along with ideas that can be implemented to extend the research in the near future.

Bibliography

- [1] Shruti Aggarwal and Janpreet Singh. Analyzing outlier detection techniques with hybrid method.
- [2] Tate M. A. Alexander, J. E. *Web wisdom: How to evaluate and create information on the web*. Mahwah, N.J: Lawrence Erlbaum Associates, 1999.
- [3] Peter Anderson. *Accuracy and Precision*. arXiv preprint arXiv:1810.09399.
- [4] S Auer, J Lehmann, R Pietrobon, A Maurino, A Rula, and A Zaveri. Quality assessment for linked data: a survey, a systematic literature review and a conceptual framework. *Semantic Web–Interoperability, Usability, Applicability*, 7(1):63–93, 2012.
- [5] Hamed Azami, Karim Mohammadi, and Behzad Bozorgtabar. An improved signal segmentation using moving average and savitzky-golay filter. *Journal of Signal and Information Processing*, 3(01):39, 2012.
- [6] Donald P Ballou and Harold L Pazer. Modeling data and process quality in multi-input, multi-output information systems. *Management science*, 31(2):150–162, 1985.
- [7] Sabyasachi Basu and Martin Meckesheimer. Automatic outlier detection for time series: an application to sensor data. *Knowledge and Information Systems*, 11(2):137–154, 2007.
- [8] Gustavo EAPA Batista, Maria Carolina Monard, et al. A study of k-nearest neighbour as an imputation method. *HIS*, 87(251-260):48, 2002.
- [9] Behshid Behkamal, Mohsen Kahani, Ebrahim Bagheri, and Zoran Jeremic. A metrics-driven approach for quality assessment of linked open data. *Journal of theoretical and applied electronic commerce research*, 9(2):64–79, 2014.
- [10] Lorenzo Beretta and Alessandro Santaniello. Nearest neighbor imputation algorithms: a critical evaluation. *BMC medical informatics and decision making*, 16(3):74, 2016.
- [11] Laure Berti-Equille. Measuring and modelling data quality for quality-awareness in data mining. In *Quality measures in data mining*, pages 101–126. Springer, 2007.
- [12] Li Cai and Yangyong Zhu. The challenges of data quality and data quality assessment in the big data era. *Data Science Journal*, 14, 2015.
- [13] Jin Chen, Per. JÃÃnsson, Masayuki Tamura, Zhihui Gu, Bunkei Matsushita, and Lars Eklundh. A simple method for reconstructing a high-quality ndvi time-series data set based on the savitzkyÃÃnÃlgolay filter. *Remote Sensing of Environment*, 91(3):332 – 344, 2004.
- [14] SciPy Cookbook. *Kalman filtering*.
- [15] Peter Craven and Grace Wahba. Smoothing noisy data with spline functions. *Numerische mathematik*, 31(4):377–403, 1978.
- [16] Andrea De Mauro, Marco Greco, and Michele Grimaldi. What is big data? a consensual

- definition and a review of key research topics. *AIP Conference Proceedings*, 1644(1):97–104, 2015.
- [17] Anwesha Barai Deb and Lopamudra Dey. Outlier detection and removal algorithm in k-means and hierarchical clustering. *World Journal of Computer Application and Technology*, 5(2):24–29, 2017.
 - [18] Ahmet M Eskicioglu, Paul S Fisher, and Si-Yuan Chen. Image quality measures and their performance. 1994.
 - [19] Randall L Eubank. *Nonparametric regression and spline smoothing*. CRC press, 1999.
 - [20] Andrea Facchinetti, Giovanni Sparacino, and Claudio Cobelli. An online self-tunable method to denoise cgm sensor data. *IEEE Transactions on Biomedical Engineering*, 57(3):634–641, 2010.
 - [21] A Famili, Wei-Min Shen, Richard Weber, and Evangelos Simoudis. Data preprocessing and intelligent data analysis. *Intelligent data analysis*, 1(1):3–23, 1997.
 - [22] Jill Feblowitz et al. Analytics in oil and gas: The big deal about big data. In *SPE Digital Energy Conference*. Society of Petroleum Engineers, 2013.
 - [23] Donatella Firmani, Massimo Mecella, Monica Scannapieco, and Carlo Batini. On the meaningfulness of “big data quality”. *Data Science and Engineering*, 1(1):6–20, 2016.
 - [24] Christopher Fox, Anany Levitin, and Thomas Redman. The notion of data and its quality dimensions. *Information processing & management*, 30(1):9–19, 1994.
 - [25] David S Fung. Methods for the estimation of missing values in time series. 2006.
 - [26] Pedro J García-Laencina, José-Luis Sancho-Gómez, and Aníbal R Figueiras-Vidal. Pattern classification with missing data: a review. *Neural Computing and Applications*, 19(2):263–282, 2010.
 - [27] Salvador García-Ba, Julián Luengo, and Francisco Herrera. Tutorial on practical tips of the most influential data preprocessing algorithms in data mining. *Knowledge-Based Systems*, 98:1 – 29, 2016.
 - [28] Karen Grace-Martin. *EM Imputation and Missing Data: Is Mean Imputation Really so Terrible?* The analysis factor. Website.
 - [29] Karen Grace-Martin. *Seven Ways to Make up Data: Common Methods to Imputing Missing Data*. The analysis factor.
 - [30] Donna Green. What is big data and why does it matter? <https://www.youtube.com/watch?v=qXyzDd2heK8>, 2015.
 - [31] José Luis Guinón, Emma Ortega, José García-Antón, and Valentín Pérez-Herranz. Moving average and savitzki-golay smoothing filters using mathcad. *Papers ICEE*, 2007, 2007.
 - [32] Jiawei Han, Jian Pei, and Micheline Kamber. *Data mining: concepts and techniques*. Elsevier, 2011.
 - [33] Andrew C Harvey and Richard G Pierse. Estimating missing observations in economic time series. *Journal of the American statistical Association*, 79(385):125–131, 1984.
 - [34] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. The elements of statistical learning: data mining, inference, and prediction, springer series in statistics, 2009.
 - [35] Benjamin T. Hazen, Christopher A. Boone, Jeremy D. Ezell, and L. Allison Jones-Farmer.

- Data quality for data science, predictive analytics, and big data in supply chain management: An introduction to the problem and suggestions for research and applications. *International Journal of Production Economics*, 154:72 – 80, 2014.
- [36] Siegfried Herzog. *First Occurrences of Prime Gaps*. zig Herzog, 2004.
 - [37] GM Hieftje, BE Holder, AS Maddux, and Robert Lim. Digital smoothing of electroanalytical data based on the fourier transformation. *Analytical Chemistry*, 45(2):277–284, 1973.
 - [38] Jennifer N. Hird and Gregory J. McDermid. Noise reduction of ndvi time series: An empirical comparison of selected techniques. *Remote Sensing of Environment*, 113(1):248 – 258, 2009.
 - [39] Eduardo R Hruschka, Estevam R Hruschka, and Nelson FF Ebecken. Towards efficient imputation by nearest-neighbors: a clustering-based approach. In *Australasian Joint Conference on Artificial Intelligence*, pages 513–525. Springer, 2004.
 - [40] J Huang, YW Lee, and RY Wang. Quality information and knowledge. prentice hall. 1999.
 - [41] Talal Hussein. *Big data in oil and gas operations and other tech advancements: seven expert opinions*. Offshore technology website, 2018. Article.
 - [42] Iris Eekhout. *Single imputation methods*. Website.
 - [43] JosÃ‰ M. Jerez, Ignacio Molina, Pedro J. GarcÃ­a-Laencina, Emilio Alba, Nuria Ribelles, Miguel MartÃ­n, and Leonardo Franco. Missing data imputation using statistical and machine learning methods in a real breast cancer problem. *Artificial Intelligence in Medicine*, 50(2):105 – 115, 2010.
 - [44] Jacob Joseph. *How to detect outliers using parametric and non-parametric methods : Part II*. Clever Tap.
 - [45] S. Juddoo. Overview of data quality challenges in the context of big data. In *2015 International Conference on Computing, Communication and Security (ICCCS)*, pages 1–9, Dec 2015.
 - [46] EL Kosarev and E Pantos. Optimal smoothing of ‘noisy’ data by fast fourier transform. *Journal of Physics E: Scientific Instruments*, 16(6):537, 1983.
 - [47] Leo L. Pipino, Yang Lee, and Richard Y. Wang. Data quality assessment. *Communications of the ACM*, 45, 07 2003.
 - [48] Tony Lacey. Tutorial: The kalman filter. *Georgia Institute of Technology*, 1998.
 - [49] Demetrios G. Lainiotis. Partitioned estimation algorithms, ii: Linear estimation. *Information Sciences*, 7:317 – 340, 1974.
 - [50] In Lee. Big data: Dimensions, evolution, impacts, and challenges. *Business Horizons*, 60(3):293 – 303, 2017.
 - [51] Yang W. Lee, Diane M. Strong, Beverly K. Kahn, and Richard Y. Wang. Aimq: a methodology for information quality assessment. *Information Management*, 40(2):133 – 146, 2002.
 - [52] Mathieu Lepot, Jean-Baptiste Aubin, and FranÃ§ois Clemens. Interpolation in time series: An introductive overview of existing methods, their performance criteria and uncertainty assessment. *Water*, 9(10):796, 2017.
 - [53] Gernot Liebchen, Bheki Twala, Martin Shepperd, and Michelle Cartwright. Assessing the

- quality and cleaning of a software project data set: An experience report. In *10th International Conference on Evaluation and Assessment in Software Engineering (EASE)*, pages 1–7, 2006.
- [54] Jianzheng Liu, Jie Li, Weifeng Li, and Jiansheng Wu. Rethinking big data: A review on the data quality and usage issues. *ISPRS Journal of Photogrammetry and Remote Sensing*, 115:134 – 142, 2016. Theme issue 'State-of-the-art in photogrammetry, remote sensing and spatial information science'.
- [55] Iain Lovatt. *What is the cost of poor quality data?* Blue sheep, 2017. Blog.
- [56] Diane M. Strong, Yang Lee, and Richard Y. Wang. Data quality in context. *Communications of the ACM*, 40, 08 2002.
- [57] S Madnick and RY Wang. Introduction to total data quality management (tdqm) research program. *Total Data Quality Management Program, MIT Sloan School of Management*, 1:92, 1992.
- [58] Stuart E Madnick, Richard Y Wang, Yang W Lee, and Hongwei Zhu. Overview and framework for data and information quality research. *Journal of Data and Information Quality (JDIQ)*, 1(1):2, 2009.
- [59] Jonathan I Maletic and Andrian Marcus. Data cleansing: Beyond integrity analysis. In *Iq*, pages 200–209. Citeseer, 2000.
- [60] Vasconcelos W Marev MS, Compatangelo E. *Towards a context-dependent numerical data quality evaluation framework*, 2018. arXiv preprint arXiv:1810.09399.
- [61] Luis MartÃŁn, Nayat Sanchez-Pi, JosÃŁ Manuel Molina, and Ana Cristina Bicharra Garcia. Anomaly detection based on sensor data in petroleum industry applications. *Sensors*, 15(2):2774–2797, 2015.
- [62] Mathworks. *Moving Average Model*, 2019. Documentation.
- [63] D. McGilvray. *Executing Data Quality Projects: Ten Steps to Quality Data and Trusted Information (TM)*. Elsevier Science, 2008.
- [64] Shahab D Mohaghegh et al. Essential components of an integrated data mining tool for the oil & gas industry, with an example application in the dj basin. In *SPE Annual Technical Conference and Exhibition*. Society of Petroleum Engineers, 2003.
- [65] Douglas C Montgomery. *Introduction to statistical quality control*. John Wiley & Sons, 2007.
- [66] Felix Naumann. *Quality-driven query answering for integrated information systems*, volume 2261. Springer, 2003.
- [67] NetVersity. What is big data? <https://www.youtube.com/watch?v=tkOw1XUaGMM>, 2014.
- [68] Yohan Obadia. *The use of KNN for missing values*. Towards data science, 2017. Website blog.
- [69] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [70] The Physicist. *Q: What is a Fourier transform? What is it used for?* Ask a mathematician, 2012.

- [71] Leo L Pipino, Yang W Lee, and Richard Y Wang. Data quality assessment. *Communications of the ACM*, 45(4):211–218, 2002.
- [72] William H Press, Saul A Teukolsky, William T Vetterling, and Brian P Flannery. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press, 2007.
- [73] Valentina Presutti, Claudia d’Amato, Fabien Gandon, Mathieu d’Acquin, Steffen Staab, and Anna Tordai. *The Semantic Web: Trends and Challenges: 11th International Conference, ESWC 2014, Anissaras, Crete, Greece, May 25-29, 2014, Proceedings*, volume 8465. Springer, 2014.
- [74] Thomas C Redman and A Blanton. *Data quality for the information age*. Artech House, Inc., 1997.
- [75] Steven Rossiter. *Big Data: Making Sense of Information and Analytics in Oil Gas*. AgileTek, 2019. Seminar.
- [76] Abraham Savitzky and Marcel JE Golay. Smoothing and differentiation of data by simplified least squares procedures. *Analytical chemistry*, 36(8):1627–1639, 1964.
- [77] G. Shankaranarayanan and Roger Blake. From content to context: The evolution and growth of data quality research. *J. Data and Information Quality*, 8(2):9:1–9:28, January 2017.
- [78] Graeme Shanks and Brian Corbitt. Understanding data quality: Social and cultural aspects. In *Proceedings of the 10th Australasian Conference on Information Systems*, volume 785. Victoria University of Wellington, New Zealand, 1999.
- [79] Bo Sheng, Qun Li, Weizhen Mao, and Wen Jin. Outlier detection in sensor networks. In *Proceedings of the 8th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc ’07, pages 219–228, New York, NY, USA, 2007. ACM.
- [80] Martin Shepperd. Data quality: Cinderella at the software metrics ball? In *Proceedings of the 2Nd International Workshop on Emerging Trends in Software Metrics*, WETSOM ’11, pages 1–4, New York, NY, USA, 2011. ACM.
- [81] Fatimah Sidi, Payam Hassany Shariat Panahy, Lilly Affendey, Marzanah A. Jabar, Hamidah Ibrahim, and Aida Mustapha. Data quality: A survey of data quality dimensions. 08 2013.
- [82] I. Taleb, R. Dssouli, and M. A. Serhani. Big data pre-processing: A quality framework. In *2015 IEEE International Congress on Big Data*, pages 191–198, June 2015.
- [83] Ikbale Taleb, Hadeel El Kassabi, Mohamed Serhani, Rachida Dssouli, and Chafik Bouhaddiou. Big data quality: A quality dimensions evaluation. 07 2016.
- [84] Ikbale Taleb, Hadeel T El Kassabi, Mohamed Adel Serhani, Rachida Dssouli, and Chafik Bouhaddiou. Big data quality: A quality dimensions evaluation. In *2016 Intl IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCoM/IoP/SmartWorld)*, pages 759–765. IEEE, 2016.
- [85] Yee Lin Tan, Vivek Sehgal, and Hamid Haidarian Shahri. Sensoclean: Handling noisy and incomplete data in sensor networks using modeling. *Main*, pages 1–18, 2005.
- [86] Giri Kumar Tayi and Donald P. Ballou. Examining data quality. *Commun. ACM*, 41(2):54–57, February 1998.
- [87] Choh-Man Teng. Correcting noisy data. In *ICML*, pages 239–248. Citeseer, 1999.

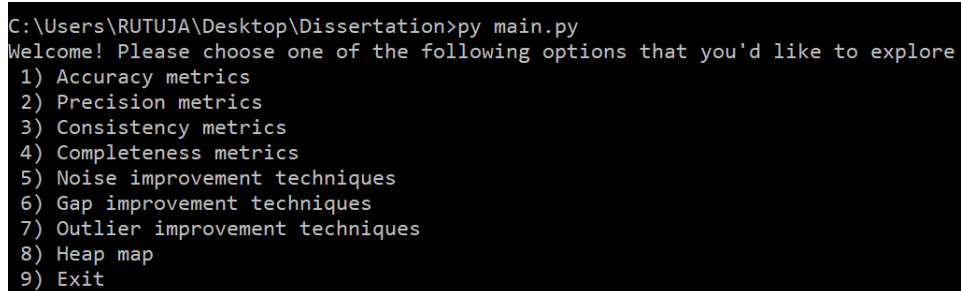
- [88] The university of North Carolina at Chapel hill. *Measurements and Error Analysis*.
- [89] Michel van Biezen. *The Kalman Filter (1 of 55) What is a Kalman Filter?* Youtube, 2015.
- [90] Jake VanderPlas. *In Depth: k-Means Clustering*. Python data science handbook.
- [91] Antonio VetrÃš, Lorenzo Canova, Marco Torchiano, Camilo Orozco Minotas, Raimondo Iemma, and Federico Morando. Open data quality measurement framework: Definition and application to open government data. *Government Information Quarterly*, 33(2):325 – 337, 2016.
- [92] Richard L Villars, Carl W Olofson, and Matthew Eastwood. Big data: What it is and why you should care. 2011.
- [93] Bruno A Walther and Joslin L Moore. The concepts of bias, precision and accuracy, and their use in testing the performance of species richness estimators, with a literature review of estimator performance. *Ecography*, 28(6):815–829, 2005.
- [94] Yair Wand and Richard Y Wang. Anchoring data quality dimensions in ontological foundations. *Communications of the ACM*, 39(11):86–95, 1996.
- [95] KQ Wang, SR Tong, Lionel Roucoules, and Benoit Eynard. Analysis of data quality and information quality problems in digital manufacturing. In *2008 4th IEEE International Conference on Management of Innovation and Technology*, pages 439–443. IEEE, 2008.
- [96] Richard Y Wang and Diane M Strong. Beyond accuracy: What data quality means to data consumers. *Journal of management information systems*, 12(4):5–33, 1996.
- [97] Wikipedia. AutoregressiveâŠšmoving-average model — Wikipedia, the free encyclopedia. <http://en.wikipedia.org/w/index.php?title=Autoregressive%E2%80%93moving-average%20model&oldid=891722434>, 2019. [Online; accessed 20-April-2019].
- [98] Wikipedia. Convolution — Wikipedia, the free encyclopedia. <http://en.wikipedia.org/w/index.php?title=Convolution&oldid=890687412>, 2019. [Online; accessed 19-April-2019].
- [99] Wikipedia. Kalman filter — Wikipedia, the free encyclopedia. <http://en.wikipedia.org/w/index.php?title=Kalman%20filter&oldid=893002439>, 2019. [Online; accessed 20-April-2019].
- [100] Wikipedia. Linear regression — Wikipedia, the free encyclopedia. <http://en.wikipedia.org/w/index.php?title=Linear%20regression&oldid=891733768>, 2019. [Online; accessed 19-April-2019].
- [101] Wikipedia. Log ASCII Standard — Wikipedia, the free encyclopedia. <http://en.wikipedia.org/w/index.php?title=Log%20ASCII%20Standard&oldid=788495901>, 2019. [Online; accessed 13-April-2019].
- [102] Wikipedia. Noisy data — Wikipedia, the free encyclopedia. <http://en.wikipedia.org/w/index.php?title=Noisy%20data&oldid=886793657>, 2019. [Online; accessed 17-April-2019].
- [103] Wikipedia. Polynomial regression — Wikipedia, the free encyclopedia. <http://en.wikipedia.org/w/index.php?title=Polynomial%20regression&oldid=887930216>, 2019. [Online; accessed 19-April-2019].
- [104] Wikipedia. SavitzkyâŠšGolay filter — Wikipedia, the free encyclopedia. <http://en.wikipedia.org/w/index.php?title=Savitzky%E2%80%93Golay%20filter&oldid=890486841>,

2019. [Online; accessed 19-April-2019].
- [105] Wolfram MathWorld. *Cubic Spline*. Website.
- [106] C.L. Wu, K.W. Chau, and C. Fan. Prediction of rainfall time series using modular artificial neural networks coupled with data-preprocessing techniques. *Journal of Hydrology*, 389(1):146 – 167, 2010.
- [107] Fu Xiao and Cheng Fan. Data mining in building automation system for improving building operational performance. *Energy and Buildings*, 75:109 – 118, 2014.
- [108] Hui Yang. Data preprocessing. 2012.
- [109] Amrapali Zaveri, Anisa Rula, Andrea Maurino, Ricardo Pietrobon, Jens Lehmann, and Sören Auer. Quality assessment for linked data: A survey. *Semantic Web*, 7(1):63–93, 2016.
- [110] Ji Zhang. Advancements of outlier detection: A survey. *ICST Transactions on Scalable Information Systems*, 13(1):1–26, 2013.
- [111] Yang Zhang, Nirvana Meratnia, and Paul JM Havinga. Outlier detection techniques for wireless sensor networks: A survey. *IEEE Communications Surveys and Tutorials*, 12(2):159–170, 2010.
- [112] Xiaolan Zhu and Susan Gauch. Incorporating quality metrics in centralized/distributed information retrieval on the world wide web. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 288–295. ACM, 2000.
- [113] Æl’loÁrse Gratton. *Big Data Analytics: Is Consent Required?*, 2016. Blog.

Appendix A

User Manual

To run this program it is important to have python version 3.6 installed. The first thing to do is unzip the SHAH_RUTUJA folder downloaded. Then the next step is to open the console/command prompt and cd (change directory) to the SHAH_RUTUJA folder. Once you are in the SHAH_RUTUJA folder, run the main.py file. This file contains function calls to all other files in the same folder so that you don't have to run each individual file separately. When run, it outputs a list containing different metrics, improvement methods and a heatmap of the dataset. You can then choose to view any of these as shown in figure A.1



```
C:\Users\RUTUJA\Desktop\Dissertation>py main.py
Welcome! Please choose one of the following options that you'd like to explore
1) Accuracy metrics
2) Precision metrics
3) Consistency metrics
4) Completeness metrics
5) Noise improvement techniques
6) Gap improvement techniques
7) Outlier improvement techniques
8) Heap map
9) Exit
```

Figure A.1: Snapshot of the output generated from the main.py file.

So suppose you want to explore metrics for the completeness dimension then choosing option 4 (by typing 4) will output the results obtained from all the different completeness metrics developed. After the results are outputted, you have an option to either go back to the menu main (list of options shown in figure A.1) or exit the program. To return back to the main menu, you must press the 'm' key as instructed. To exit, you may press any key you like.

Upon choosing improvement methods for a quality issue of your choice, you will be provided with a further set of options to choose from. For example, if you wish to view noise improvement methods (option 5) then once you type 5 onto the console, you will be provided with a list of improvement methods for noise to choose from. For every noise method you choose, the console will output the average uncertainty increase score, the standard deviation of the improved points, a visualization and a list containing all the improved points. This is shown in figure A.2. Then you can press 0 to explore a different noise improvement method or press m to go back to the main menu or press a random key to exit the program. All these instructions will be presented on the console itself.

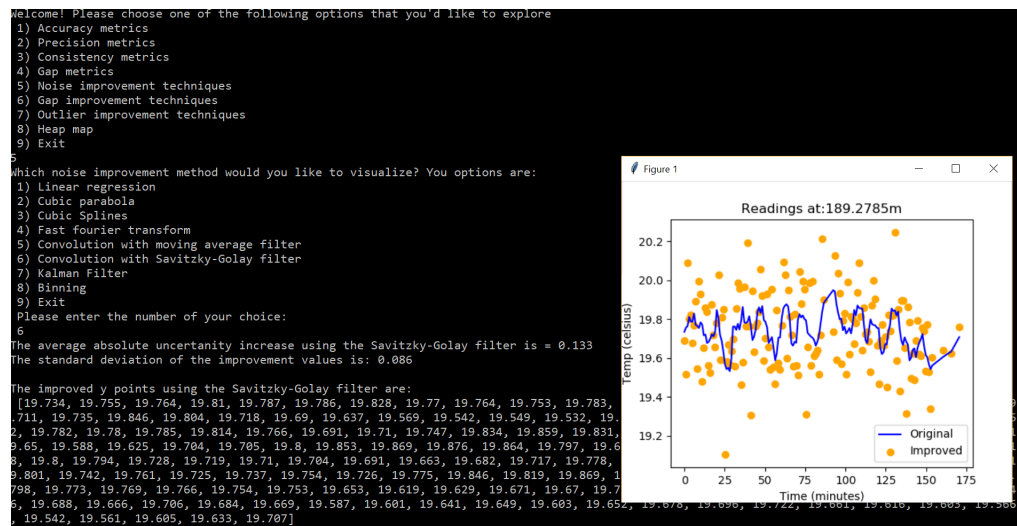


Figure A.2: Snapshot of the output generated from an improvement method.

Appendix B

Maintenance manual

B.1 Installations

As mentioned in appendix A, it is important to have **python 3.6** installed in order to run this program. A few libraries and packages need to be installed. In order to install these, please make sure you have **pip** installed first. The libraries and packages that need to be installed are:

- matplotlib 2.2.0
- scipy 1.2.0
- numpy 1.14.1
- statsmodels 0.9.0 (package)
- scikit-learn 0.20.3 (in order to download this library, you must have numpy and scipy installed first)

From the Scipy library, interpolate, fftpack and signal have been imported. The interpolate module has been used for cubic splines. The fftpack package has been used for FFT and signal has been used for the Savitzky golay filter. From the matplotlib library, pyplot and colors have been imported. The pyplot module is used for plotting all the visualizations and the colors module is used for the heatmap. From the statsmodels package, the ARMA module has been imported for ARMA. And lastly, from scikit-learn, neighbors and kmeans have been imported. Neighbors has been for KNN and kmeans has been used for KMC.

Below are some other modules that are required to run the program. In most cases these do not need to be installed. They are in-built modules provided with python. If however the program does not run without them and asks you to install them, then please do so. These modules are:

- glob
- time
- itertools
- statistics
- functools

B.2 How to run the program

It is very simple. Once you have unzipped the SHAH_RUTUJA folder and you have downloaded all the required libraries, open the command line. Then CD (change directory) to the SHAH_RUTUJA folder from your current location and type **py main.py** to run the file as shown in figure B.1 . The dataset path has been set inside the code files as shown in figure B.2 so you need not worry about this. More details from this point onwards have been provided in appendix A.

```

Microsoft Windows [Version 10.0.17134.706]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\RUTUJA>cd Desktop

C:\Users\RUTUJA\Desktop>cd SHAH_RUTUJA

C:\Users\RUTUJA\Desktop\SHAH_RUTUJA>py main.py
Welcome! Please choose one of the following options that you'd like to explore
1) Accuracy metrics
2) Precision metrics
3) Consistency metrics
4) Completeness metrics
5) Noise improvement techniques
6) Gap improvement techniques
7) Outlier improvement techniques
8) Heap map
9) Exit
5
Which noise improvement method would you like to visualize? You options are:
1) Linear regression
2) Cubic parabola
3) Cubic Splines
4) Fast fourier transform
5) Convolution with moving average filter
6) Convolution with Savitzky-Golay filter
7) Kalman Filter
8) Binning
9) Exit
Please enter the number of your choice:

```

Figure B.1: Snapshot of how to run the program

B.3 Organisation of files

This SHAH_RUTUJA folder contains the following files and folders (each of the IM and metrics in these files have been thoroughly explained in chapter 4):

- main.py (this is the main file which imports all the code files mentioned below)
- metric_accuracy.py (contains all the metrics developed for the accuracy dimension)
- metric_completeness.py (contains all the metrics developed for the completeness dimension)
- metric_precision.py (contains all the metrics developed for the precision dimension)
- metric_consistency.py (contains all the metrics developed for the consistency dimension)
- improv_noise.py (contains all the IM developed for the noise)
- improv_gaps.py (contains all the IM developed for the gaps)
- improv_outliers.py (contains all the IM developed for the outliers)
- heatmap.py (contains code for visualising the heatmap)
- GG_DTS_ASCLL (this folder contains the dataset used)

B.4 Important files and functions

Each code file (besides main.py) has an **inputuser** function which calls all the method/functions (used interchangeably) in that file. This function, in most cases, creates a user interactivity which enables the user to choose and view all the methods in that file. So for example, improv_noise.py file's inputuser function calls the respective noise IM (discussed in chapter 4) based on user input on the console. These individual files can be run to obtain results but would these only provided limited/local user interactivity.

In order to view the methods in ALL the code files, the **main.py** file should be run. The main.py file imports the inputuser functions from all the other code files. Based on user interactivity, It calls the respective inputuser function of a file which then calls methods in that file. The purpose of this file is to give a global user interactivity by letting the user explore all IM, metrics and heatmap all at once instead of having to run each code file individually.

The values of the dataset are obtained from a function called **func** defined in the improv_noise.py file. This function takes height/depth and path as arguments. The purpose of this function is to obtain a set of time (x) and temperature (y) values for a given depth in a given dataset. The depth is stored in the 'height' variable and the dataset path is stored in the 'path' variable. These two variables are defined outside the function as shown in figure B.2. This function is then imported in the rest of the code files (besides main.py) to enable re-usability.

```
height = 189.2785 #choose the height/depth for which you want to retrieve all the temperature vs time values for
path = "GG_DTS_ASCII/Post-B_G01/DTS_G1_BLEED_1/channel 1 20140718 *.dts"
```

Figure B.2: Snapshot of the height and path variables available in each code file.

A default depth is provided for all files but If you wish to change it then you can do in any file you want (besides the main.py). Similarly, you can change the dataset too as long as the dataset you have chosen is of the same format/standard as the one currently being used. Both depth and dataset path variables are located at the top of every file (besides the main.py). If you wish to make a change, you have to make it in these variables as shown in the figure B.2. If you want to change the default depth to 113.5m for example, then this value (113.5m) must exist in the dataset you are using otherwise you will get a key error.

B.5 The dataset

The two datasets used by this system are DTS_G1_BLEED_1 and DTS_G1_BLEED_1. Both of these are located in the subsubfolder of the GG_DTS_ASCLL folder. File paths are specified in code files as shown in figure B.2. The rest of the dataset has been thoroughly explained in chapter 3.

B.6 Space and memory requirements

Space requirements for SHAH_RUTUJA folder: Size: 15.1 MB, size on disk: 16.2 MB

Memory requirements: 138.1 MB

B.7 Extensibility and future use

Because IM and metric functions are independent of one another, it is rather easy to extend the program. To add a new IM or metric, simply create a new function and append it to the file of your choice. All you need to then do is to add a call to that function in the inputuser function of that file to enable interactivity. For future adaptations, an option to choose a depth or dataset could be provided on the console so that the user can change it there instead of having to changes in the code. Additionally, options to save the improved x and y points into a csv can be provided. A small script can then also be made to impute these improved points in place of the original ones in the dataset.

B.8 Further information on weighted gaps

So as you can see in figure B.3, a couple of variables have been defined. These can be changed by the user depending on their requirements. It is important to understand that all of these are in sync with one another so changing one would require changing all in most cases. The 'Start' variable defines the biggest gap size which is 10 in this case. So a gap size of 10 or above gets assigned the same weightage. The second variable is 'diff' which is the difference/step size between the first gap and the next. So first, the program looks for gaps of size 10 and above, after that it looks for gaps of size 9 and above so on etc...up until the 'limit' which is 1 has been reached. So the limit is where the loop stops. So for example, if the limit had been set to 2 then the loop will not assigned a weightage to gaps of size 2.

```
list = []
start = 10 #sta
diff = 1 #if st
start_w = 0.82
diff_w = 0.02 #
count = 1 #to d
limit = 1 # thi
```

Figure B.3: Variables set for the WGPM

A weightage is assigned to each gap size. So the start gap gets a start weightage assigned to it which is 0.82 stored in variable 'start_w'. So when scanning through the dataset, if a gap of size 10 is encountered, then it gets weighting 0.82. This 0.82 is typically 82%. So if % completeness calculated by the simple gap percentage metric is 58% and if it has a gap of size 10 then the score reduces to 47.56% (58×0.82) which is an 18% decrease ($100 - 82$).

The increase/step size between one weightage and the next is defined in variable 'diff_w' which currently holds the value of 0.02. So gap with size 9 gets assigned weightage 0.84 ($0.82 + 0.02$). So the weightage comes closer to 1 (100%) as the gap size decreases. So ideally, the start gap 10 gets weightage 0.82 and by the time the loop gets the gap 2, it gets weighted 0.98 which is the smallest weightage possible (because this means only 2% decrease). This is the smallest because weightage 1 (0% decrease) would be assigned when there are no gaps at all (although this would not be the case because the loop never gets to the limit value so this is only explained for

understanding purposes). So this is the reason all the variables have to be in sync. The values currently held in these variables have been carefully designed and if you want to make a change, you need to make appropriate changes to the rest.

So the algorithm loop shown in figure B.4 works as the follows. The loop starts with checking the difference between point 1 and point 2 and first checks if the difference between the two is of 10 or above. If yes, then 0.82 is appended into the list and the loop breaks and continues scanning the next two elements. If no, then the loop checks if the difference is 9 or above (because the difference could have also been 9.2 for example so the difference won't exactly be 9). If yes, then 0.84 is appended into the list and the loop breaks and continues scanning the next two elements. If not, then gap of size 8 or above is looked for and so on until it reaches the limit. Once it reaches the limit, the loop goes through the next two points. This process is repeated until all the data points of the dataset have been scanned. Then in the end, if the list is not empty, then the % completeness is calculated using the simple gap percentage metric. This % completeness score is then multiplied with a product of all the weightages in the list, which then reduces this score calculated. If the list is empty, then the % completeness score is computed using the simple gap percentage metric.

```

for i, val in enumerate(x):
    if i != len(x)-1:
        #so in this case program loops only 9 times.
        for j in range(start-limit): #This is important so that it does not loop for the limit-th value.
            if j == 0: #first loop looks for gap of size 10, if it finds it then it appends the corresponding wei
                if x[i+1]-x[i] >= start:
                    list.append(start_w)
                    break
            else: #then looks for gaps of size 9, then 8 etc up until it reaches size 1.
                if x[i+1]-x[i] >= (start-(diff*count)):
                    list.append(round(start_w + (diff_w*count),2))
                    break
            else:
                count = count + 1
        count = 1 #resetting to loop through next index value.
if list == []: #if list is empty then it means that there are no gaps
    #if there are no gaps, then the percentage is calculated in the classic way and will get same results as gap
    gap_ratio_2 = len(x)/x[len(x)-1] * 100
    print('The completeness of this data using the weighted gap metric is: ', gap_ratio_2, '%')
else:
    list_new = reduce(lambda x, y: x*y, list) #multiple all the weights
    gap_ratio_multiple_2 = (len(x)/x[len(x)-1] * 100) * list_new #multiple weights with % metric
    print('The completeness of this data using the weighted gap metric is: ', round(gap_ratio_multiple_2,2), '%')

```

Figure B.4: The WGPM algorithm