TITLE

by

Gideon Pieter Slabbert

Submitted in partial fulfillment of the requirements for the degree
<missing degree code option>

in the

Department of Chemical Engineering
Faculty of Engineering, Built Environment and Information Technology

UNIVERSITY OF PRETORIA

October 2020

SUMMARY

TITLE

by

Gideon Pieter Slabbert

XXXXXXXXXXX:	Mr C Sandrock
Department:	Chemical Engineering
University:	University of Pretoria
Degree:	<missing degree code option>
Keywords:	Keyword, keyword, keyword, keyword, keyword, keyword, keyword, keyword, keyword, keyword, keyword, keyword

Summary paragraph one summary paragraph one summary paragraph one

LIST OF ABBREVIATIONS

ABC	Write out in full
DEF	What the abbreviation represents
GHI	Note the use of case
JKL	Write out in full
MNO	What the abbreviation represents
PQR	Note the use of case
STU	Write out in full
VWX	What the abbreviation represents
YZ	Note the use of case

TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION

1.1 PROBLEM STATEMENT

1.1.1 Context of the problem

The modern industrial complex has a large number of automated processes. The automation of processes enables sustainable and predictable production. Control loops are one of the key building blocks that make automation possible.

Each sub section of a process consists of multiple control loops that together form a complex system. Continuous maintenance and fast response to disturbances is required to ensure that the overall goals of the automated process is sustained.

However, the complexities in analyzing the various number of control loops that form part of industrial systems causes a time delay between the onset of a fault and the detection of that fault. Furthermore, once a fault is detected it can be very difficult to discern the source of a disturbance without time consuming interventions that add risk to process stability.

Dynamic analytical methods for fault detection based on fundamental knowledge of the processes are well developed in academic research environments. These methods struggle to gain traction in industrial environments because of the time and implicit costs involved with incorporating these dynamic, first principle models within existing processes.

In industry, a data driven approach is mostly applied in solving problems and a lot of focus is given in developing methods and tools that support this approach. An approach that provides the user with an intuitive and interactive tool to analyze process behavior can add value by requiring less fundamental knowledge and quicker response time to address process instabilities.

A key element of any tool that assists in process fault finding is the visualization and interaction with the data. Every decision made in a industrial process involves reacting to process data. In modern industrial processes most short term decisions are made using the process data as visualized on the process control system. This same data is used to monitor and detect deviations in the process. When deviations are detected a fault finding process needs to be followed to determine what caused the deviation and how to correct it.

The fault finding process often requires a lot of data analysis using a certain frame of reference to determine what is normal behavior. With any process there needs to be some kind of visual interaction with the data in order for the human brain to process it. This visual interface can range from the simplest form of simply looking at how a point value on the control system changes over a period to constructing graphs of complex time series transformation of multiple data sets.

1.1.2 Research gap

The intent of this study is to explore different approaches to the fault detection and analysis process with specific focus on the human interaction with data to analyze and solve a problem. The core element of this process is the visual context in which the data is viewed and transformed in order to generate understanding and knowledge.

In the modern industrial era a lot of fault detection problems are solved using computer based tools that are very efficient at processing large amounts of data and performing complex data transformations in a relatively short period of time.

A time series fault detection and diagnosis problem is often solved by creating a specialized tool that would transform data into knowledge which would most likely have a visual representation as the end result. The problem and solution are then presented as the end result of the study. If the audience then want to re-produce the results or test the method on a different data set they have to reproduce the tool that was used to generate the knowledge as well. This is a time consuming process and a barrier to further development of the proposed method.

A researcher might also make the specialized tool available to the audience but if the tool was developed on a platform that uses proprietary code then it also presents a barrier to further development. There are a lot of tools available for visualization of data but very few of them are built on open source platforms. Recent development in the computer science industry has made available a large number of open source tools and platforms, but there are very few examples in literature of them being applied in the visual analytics realm.

New developments in the fault detection and diagnosis problem have presented many new views of how to identify that a fault has occurred or how to identify the root cause of a fault. These new methods display the results in static graphs that highlight the most important elements of the transformed data set. One key element of the user experience that has not been developed yet is the ability to interact with the data. Interaction allows for faster knowledge transfer and easier understanding of how the parameters used in the transformation effect the final result. If the audience could be able to interact with the presented data at every step of the transformation then a much faster and deeper understanding could be achieved of the communicated method.

1.2 RESEARCH OBJECTIVE AND QUESTIONS

The focus of this dissertation will be to explore the ways in which visual analytic systems can be used in the fault finding process. One particular fault finding technique that involves complex transformations of data to determine the root cause of disturbances will be discussed and demonstrated. A visual analytic system will be developed based on best practices to attempt improving the user experience when analyzing a complex dynamic problem.

- Is this a research question?
- Is this the second research question?
- Is this the third research question?

Research objective and questions research objective and questions

1.3 HYPOTHESIS AND APPROACH (FOR ILLUSTRATION THIS HEADING CONTINUES ONTO THE NEXT LINE)

Hypothesis and approach hypothesis and approach hypothesis and

1.4 RESEARCH GOALS

Research goals research goals research goals research goals research

1.5 RESEARCH CONTRIBUTION

Research contribution research contribution research contribution research contribution research contribution research contribution research contribution research contribution research contribution research contribution.

1.6 OVERVIEW OF STUDY

Overview of study overview of study overview of study overview of

CHAPTER 2 LITERATURE STUDY

2.1 CHAPTER OBJECTIVES

This chapter explores the literature in the field of disturbance detection and diagnosis in control of industrial processes. It then elaborates of visualization techniques and how they could be used to aid the in the fault detection and diagnosis process.

2.2 FAULT DETECTION AND ANALYSIS

Disturbances in single control loops are fairly straight forward to solve but disturbances that propagate through a complex system on a plant wide scale are much harder to pin point. One of the goals of process control is to shift process variability to places like buffer tanks and plant utilities. In modern industrial processes these options have become limited as inventory is usually kept to a minimum and the use of recycle stream for heat integration has introduced extra sources complex interaction [1].

When considering which type of disturbances to analyze, the time frame of the disturbance can be used to distinguish between different types of disturbances. [1] classified the types of disturbances in a plant wide setting into three main categories with regards to timescales:

1. slowly-developing, i.e. fouling of a heat exchanger
2. persistent and dynamic
3. abrupt, i.e. a compressor trip

The second item, persistent and dynamic disturbances that propagate through an entire system on a continuous basis is of interest for this study.

New data driven methods that apply concepts from other fields have started to show some promising results. One of these methods is the use of the PageRank algorithm [2] employed by google on the ranking of control loops based on the degree of interaction.[3] has developed a methodology called LoopRank with the aim of prioritizing control loop maintenance by highlighting loops that have a higher correlation with other loops. In this methodology a relative weight matrix is compiled using partial correlation between input and output variables. The importance scores of each variable is then determined by calculating the normalized eigenvector of the relative weight matrix.

2.2.1 PAGERANK DERIVATION

The ranking algorithm developed by [2] started with the idea of ranking web sites depending on how many times they are referenced by other websites.

Consider a group of 4 websites that contain references to each other in the manner indicated in Figure 2.1. References are indicated by arrows.

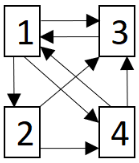

Figure 2.1. web pages with references

An initial idea would be that a page that has more references or back links is more important. Let x_k be the importance of page k. If this method is followed then $x_3 = 3$ would be the most important web page because it has the most references.

This initial ranking method indicates that x_3 is important, but it doesn't take into account the importance that x_3 passes on to x_1 because it is the only page referenced by x_3.

Another approach would be to let back links from important pages count more than back links from less important pages. If a page k has a back link from page j, then the score given to this back link can be calculated as $x_k = \frac{x_j}{n_j}$ where n_j is the total number of references made by page j. The total score for page x_k is then calculated by the sum of all back links to it adjusted for the number of references made.

Applying this new method the importance score for each node is calculated:

$x_1 = \frac{x_3}{1} + \frac{x_4}{2}$

$x_2 = \frac{x_1}{3}$

$x_3 = \frac{x_1}{3} + \frac{x_2}{2} + \frac{x_4}{2}$

$x_4 = \frac{x_1}{3} + \frac{x_2}{2}$

These equations can be written in a matrix form, which will be defined as the link matrix A:

$$A = \begin{bmatrix} 0 & 0 & 1 & \frac{1}{2} \\ \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{2} & 0 & 0 \end{bmatrix}$$

Each row in the matrix A represents a page or node while each column represents the importance score from another node. For example row one represents node 1 and it has a back link from node 3 with an importance score of 1 and a back link from node 4 with a importance score of $\frac{1}{2}$.

if $x = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix}^T$ then the problem can be defined as the eigenvector problem:

$Ax = x\lambda$

with x being an eigenvector for A with en eigenvalue of 1

if A is a $n \times n$ matrix, then a nonzero vector x is an eigenvector if $Ax = \lambda x$ for some scalar λ. The eigenvector x that solves this equation is then calculated to be:

$x \approx \begin{bmatrix} 0.387 & 0.129 & 0.290 & 0.194 \end{bmatrix}^T$ which shows that node one is in fact more important than node three.

2.2.2 LOOPRANK ADAPTION

[3] developed a method called LoopRank which is based on the PageRank algorithm to prioritize control loop maintenance. Modern industrial sites have a large number of control loops which becomes challenging to maintain effectively given limited technical resources.

The LoopRank method can be summarized in five steps:

1. Collect time series data of control loop parameters
2. Calculate the partial correlation between all the variables in the data set. This information is used to build a relative weight matrix A or link matrix as referred to in the PageRank algorithm

3. Compute the LoopRank based on the relative weight matrix A using the eigenvector centrality theorem
4. multiply the result for each loop with a pre-defined weight that is based on the criticality of the loop. This weight is assigned heuristically depending on loop type and profitability
5. Normalize the results to express the score for each loop as a percentage

[3] demonstrated the efficacy of LoopRank by testing it on an industrial case study in which it successfully flagged the most critical control loops.

2.2.3 LOOPRANK DEVELOPMENT

[4] compared LoopRank with a more traditional data driven method that utilizes the integral of absolute or squared error to determine loop interaction. Canonical correlation was used to calculate the relative weight matrix in the LoopRank methodology. It was found that both methods delivered similar results.

The LoopRank methodology was further expanded on by [5] who tested the LoopRank using transfer entropy in an attempt to get a better estimation of causality when calculating the relative weight matrix. [6] used this methodology for incipient fault detection by calculating importance scores over multiple overlapping time regions.

2.3 VISUALIZATION FOR FAULT DETECTION

The methodologies involved in fault detection and diagnosis of industrial processes rely heavily on the visual analysis of time series data. Data is continuously generated by industrial processes, this data in its simplest form can be floating point or boolean values on a control system interface screen. This data is used by plant personnel as feedback on the current status of automated processes.

This study will focus on methods and approaches taken to visualize temporal or time series data.

2.3.1 VISUALIZATION IN GENERAL

Some of the first books to discuss information visualization date back to 1967 with the publication of Jacques Bertin's Semiologie Graphique [7]. A survey by [7] highlighted the number of visualization books have been published each year, displayed in 2.2.

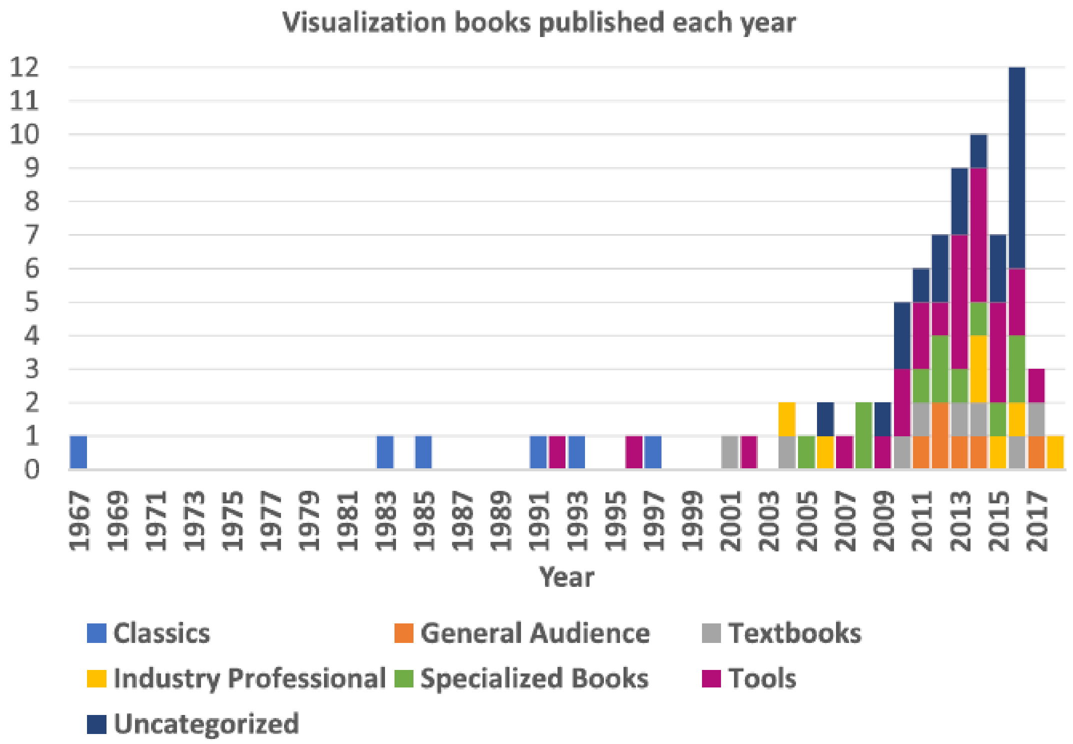

Figure 2.2. Visualization books published each year

The IEEE Visualization conference series aimed at sharing developments in this field started in 1990 and has contributed a large amount of data on how visualization assists other fields to better understand their data. [8] compiled a database of these publications to assist researchers in this field to understand the history of visualization in a field and make the researcher aware of new developments trends. The papers presented through these conferences serve as valuable source to describe previous work related to visualization for the use of fault detection and diagnosis.

2.3.2 VISUAL ANALYTICS

Before designing it is important to understand the mental processes behind data analytics that are used by people to understand and gain insight into data. The work discussed in this section first attempts to describe the human aspect of of the process and then attempts to integrate the role of the machine or computer into the process.

One of the earliest models for that describe the process of 'sensemaking' was developed by [9]. In this model intelligence analysis or sensemaking consists of:

1. information gathering
2. representation of the information in the form of an outline or model that aids the analysis
3. development of insight through manipulation of the outline or model
4. Creation of a knowledge product or direct action based on the analysis

The process is visually described by Figure2.3.

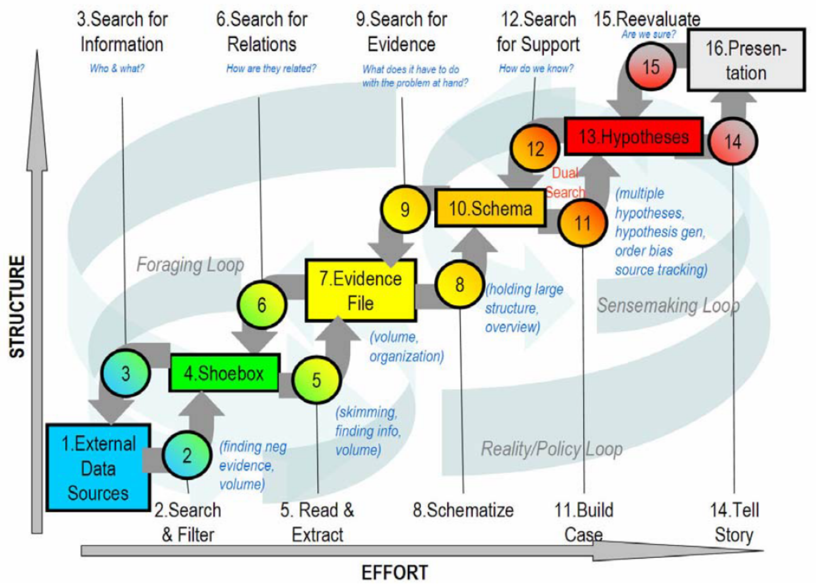

Figure 2.3. [9] sensemaking model

The rectangles represent the flow of data and the circles represent the process flow. What is very clear from this model is that there are a lot of back loops and sets of activities that cycle around finding information and making sense of the information. The end result of the process is the generation of novel information by the analyst.

The External data sources are filtered into a smaller data set called a shoebox. Snippets are then extracted from the shoe box based on observed relationships and placed in the Evidence File. A Schema or model is then developed and verified using the Evidence file. A Hypotheses is then form based on this model with a feedback loop that search for additional support of this hypotheses. Finally there is a presentation that is first reevaluated before becoming the product of this work. This model can be used in a top-down (theory to data) or bottom-up (data to theory) approach.

Another approach that represents a more non-linear model that might be a closer representation to how humans actually process information. [10] present a model that starts of with the assumption that people always start making sense of something with some perspective, viewpoint or framework. The frame basically functions as hypothesis of how the data fits together.

These frameworks or frames actually define what counts as data and shape the data. Frames also change as we acquire data making it a two way path between a frame and data.

The first cycle that starts from the frame/ data idea is the elaboration cycle. In this cycle the frame is expanded and questioned - there could be doubt about the explanation it provides. If the doubt, possibly caused by troublesome data is explained away then the frame is preserved.

The second cycle involves a re framing process. If questioning the frame leads to its rejection, it might lead to its replacement with a better one. We could compare frames to see which one fits the data best.

This model is displayed in Figure2.4.

This is a continuous process in which a frame is strengthened or improvement at each iteration.

Any method for turning data into knowledge requires the manipulation of the data. The analysis would start with an analysis of the data. This analysis has traditionally been done manually but with the exponential growth in the amount of data that can form part of an analysis other methods are being developed to make this step more efficient. [11] analyzed this challenge in the emerging field of knowledge discovery in databases (KDD). In simple terms this field is concerned with development of methods and techniques for making sense of data.

One of the most noteworthy fields where these methods are applied is in the areas of astronomy. These fields contain terabytes of data contained in images. Although industrial application of data analysis contain a lot less data than this it could still be worth while studying these techniques to see if they can assist with effectively utilizing large amounts of time series data.

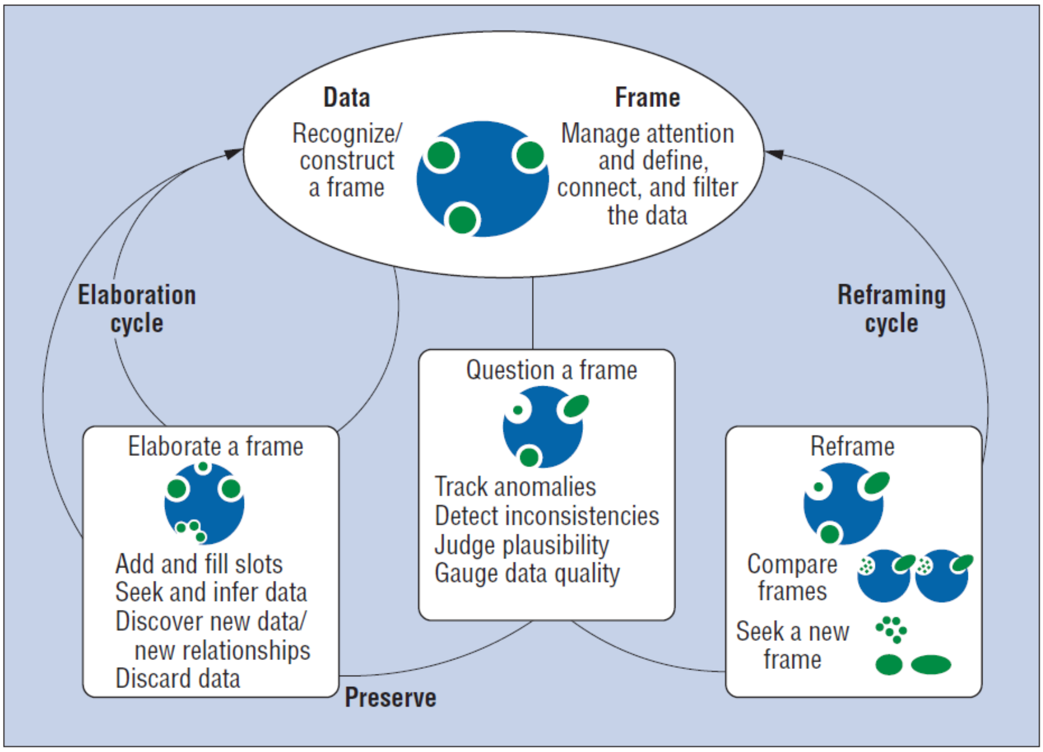

Figure 2.4. [10] frame-data sensemaking model

The goal of KDD is extracting high-level knowledge from low-level data in the context of large data sets [11]. The KDD process is visually described by Figure 2.5. The process is described by the

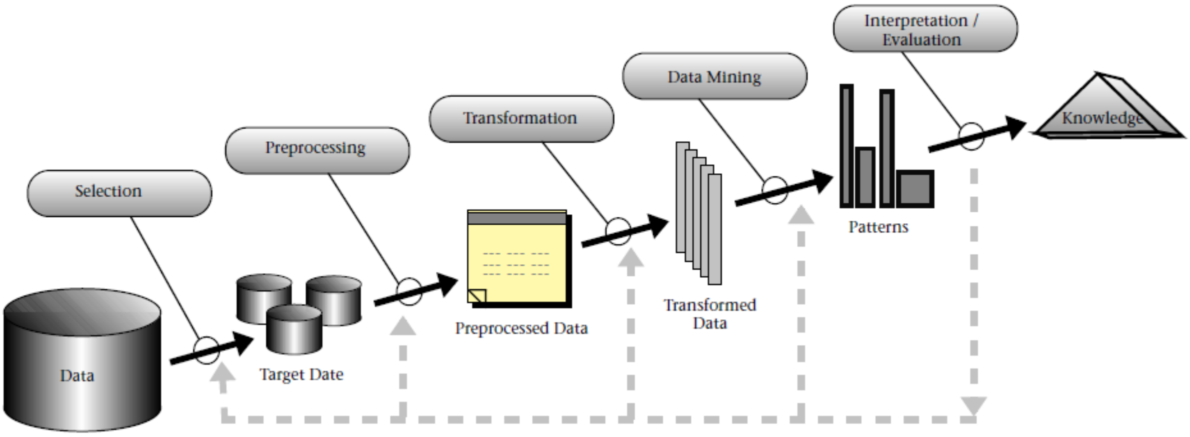

Figure 2.5. [11] steps in the knowledge discovery in databases model

following steps:

1. Develop understanding of application domain and use prior knowledge to determine the goal of the KDD
2. Create a data set or subset of the available data that will be used to discover useful information
3. Data cleaning, in this step invalid data like noise of questionable data is removed in order to prevent false deductions
4. Data reduction and projection - finding useful features to represent the data, in this step dimensional reduction can be done to simplify the analysis
5. Match the goals of KDD to a specific data mining method
6. Exploratory analysis of the model and hypothesis selection, in this step the data mining method is chosen as well as the methods that will be used to search for patterns
7. Data mining - searching for patterns in the data that would confirm the hypothesis
8. Interpreting the mined patters
9. Acting on the discovered knowledge, using it to solve the problem

These steps describe a linear process but there can be movement back and forth between the steps to refine the process.

[12] goes a step further to describe the process in which knowledge is generated from data by developing a model that defines how the human and computer interact. This model is split into two parts:

1. computer system which involves the data, visualization and analytic models
2. human component that describes the cognitive process associated with an analytical session

What makes visual analytics so effective is the integration of these two parts; the human side provides the perceptive skills, cognitive reasoning and domain knowledge while the computer side provides the computing data storage capability [12]. This visual analytic model also attempts to incorporate some of the other models described earlier like the Sensemaking model [9] and the Knowledge Discovery Process in Databases (KDD) [11]. The complete knowledge generation model with the reference to other models is displayed in Figure 2.6.

The computer part of the model has three main elements:

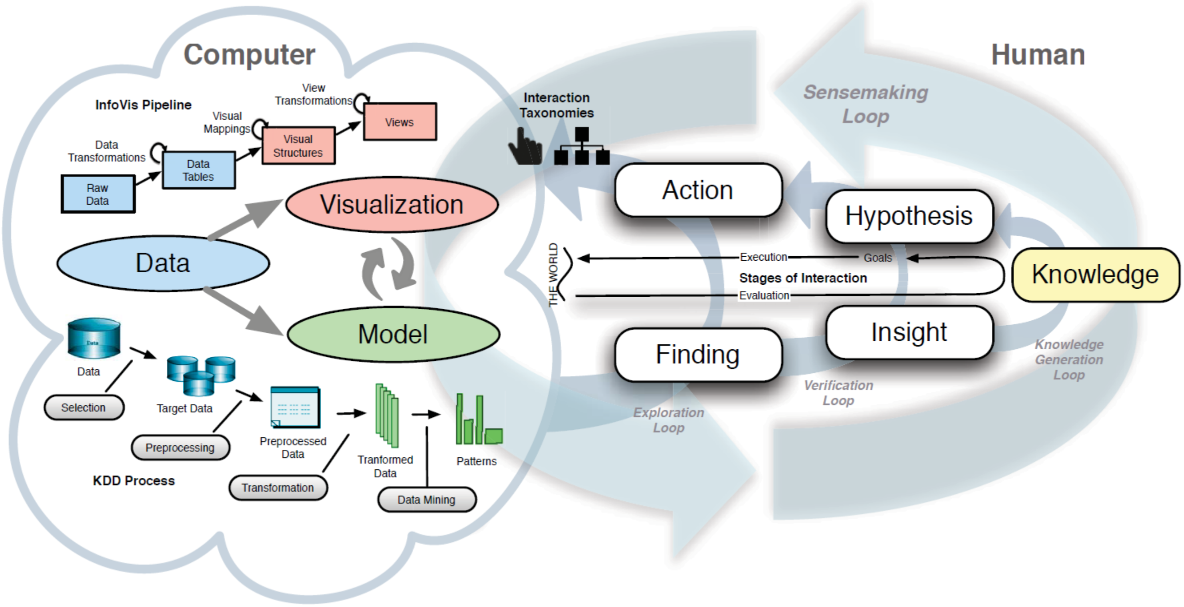

Figure 2.6. [12] computer-human interaction in knowledge generating process

1. Data - this includes all the data that describes the analytical problem and includes facts in any structured, semi-structured or unstructured manner. Another element to this is a term called meta data, which could be data that describes the structure of other data or data that summarizes other data.
2. Model - This can be any descriptive measure of the data ranging from statistical properties of the data to complex data mining algorithms. This process can deliver a single number to describe the strength of an hypothesis or a complex pattern.
3. Visualization - In this process the data is transformed into some kind of visual artifact that enables the analyst to detect relationships.

, The human element is also described in three main loops: The computer part of the model has three main elements:

1. Exploration loop - Findings are made based on observations of the visual analytics system, Actions or individual tasks are then performed that result in tangible and unique responses from the visual analytics system.
2. Verification loop - Insight is gained from interpreting Findings, often with previous domain knowledge to generate units of information. A Hypothesis is then formed based on these Insights

and tested in the visual analytics system through Actions.

3. Knowledge - The final part is to build up confidence in the Insights from the data and Hypotheses tested on the data. If patterns or stories told by the data makes sense and can be used to answer the questions from the visual analytic process then Knowledge is generated. This Knowledge is added to the domain knowledge of the analyst to ask more questions and generate more Knowledge.

This model proposed by [12] again highlights the iterative process of visual analytics and how feedback loops are used in every element of the model.

2.3.3 VALUE OF RECORD KEEPING IN VISUALIZATION

[13] conducted a experiment to study the visual analytics process and the role of record keeping in this process. This study also elaborated on key elements for the design of a collaborative visual analysis tool. The study required groups to complete tasks given a certain data set using computer aided visual analytics. Actions taken during visual analytics were grouped into four phases:

1. Problem definition - understanding what needs to be solved
2. Visualization - generating visual artifacts
3. Analysis - most complex phase that requires examining of charts and making sense of them combined with other data
4. Dissemination - compiling findings in a report

The findings from the study conducted by [13] confirmed observations by other researchers about the non-linear temporal order of activities. While completing the tasks users would move through the phases in a variety of orders. The visualization and analysis phases were strongly interrelated, the users moved back and forth between these two phases much more than the other phases.

[13] generated a very visual model of these four phases with key elements noted in each phase:

Record keeping involves the saving of visual artifacts and notes. In most cases a visual artifact can convey a complex idea more effectively than notes. Visual artifacts were saved for two main reasons; using them in reporting and using them to enhance the analysis process.[13] recommends that visual analytic systems provide functionality to save visual artifacts like charts for later use and comparison.

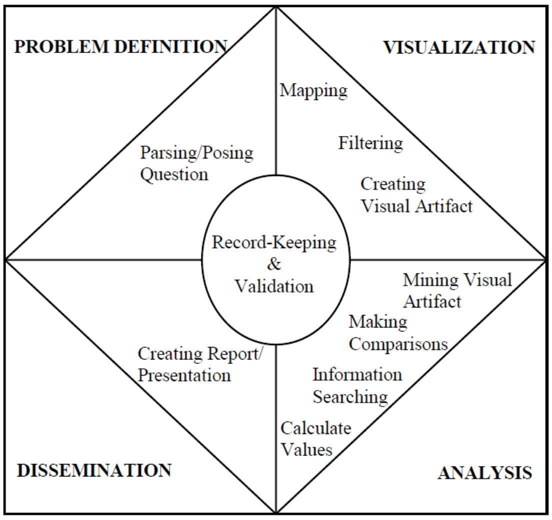

Figure 2.7. Visualization books published each year

Note taking also plays a critical role in all phases of the process. The notes high level content could be divided into two categories; findings and cues. Findings are a results of mathematical calculations, statistical operations or decisions and outcomes of the analysis process, they could also take to form of saved charts. Cues can be anything that is not directly extracted from the visualizations that could assist the user to framing or understanding the task at hand. A larger proportion of findings were recorded in the analysis phase while the cues were largely taken during the visualization phase. The study concluded that record keeping is used intensively by data analysts.

[14] completed a longitudinal study of a bioinformatics data set analysis to gain understanding of the entire analysis process from raw data to insights. Visual analytics can be described as a process wherein user gain insight into complex data. The study highlighted how analysts gather and build insights over long periods of time and connecting the data to extensive domain knowledge and expertise. It is emphasized that analyst spend a significant amount of time on manual manipulation of data and this creates a very important need for the visualization tool to be interactive. Inferior visualizations with interaction were preferred over superior static visualizations.

2.3.4 VISUALIZATION TOOLS

[15] performed a comparison of visualization tools in the development of a tool that can be used for data analysis at CERN. The goal of this tool was to enable scientist to observe a data set as a whole in order to make better decisions of what variable pairs and time frames to analyze during test campaigns.

JavaScript and Python Libraries were investigated and evaluated. D3 and Dygraphs were both discussed as JavaScript libraries that can be used for plotting. Matplotlib was discussed as a Python library that is based on the plotting functionalities of MATLAB. Two other Python libraries, Bokeh and Mpld3 were discussed with specific attention to their capabilities to generate HTML which enables them to easily present graphs on a web browser. Furthermore Mpld3 can integrate Matplotlob plots using D3 to render them to a web page.

[15] did a detailed comparison of Bokeh and Mpld3 which identified Bokeh as their preferred choice. Comparing key aspects like loading time, rendering time and features Bokeh was faster and had more actively developed libraries.

[16] did a comparison of data visualization systems with the specific focus on big data applications. Tools were categorized into coding and zero coding tools with graphical user interfaces.

Within the coding category two languages were discussed; Python and R. Within Python multiple libraries were analyzed for visualization, including Bokeh, Altair, Seaborn, ggplot and Pygal. The zero coding category included Tableau, Infogram, Charblocks, Datawrapper, Plotly, RAW and Visual. It's important to note that of these zero coding tools Plotly is also a Python library used by the data science community.

[17] also completed an overview of tools used for big data visualization and the challenges faced with attempting to visualization large amounts of data. The tools were compared using the following criteria:

1. Open Source
2. Integration with popular data sources like Hadoop Hive or Google Analytics
3. Interactive visualization
4. MOOCS; tutorials available for online learning of the tool
5. API; can the services of the tool be embedded

Five tools were analyzed using this criteria:

1. Tableau
2. Power BI
3. Plotly
4. Gephi
5. Excel 2016

A summary of the results is displayed in Table 2.1 shows an example of a table.

Table 2.1. This is a table caption

	Open source	**Integration**	**Interactive**	**MOOCS**	**API**
Tableau	N	Y	Y	Y	Y
Power BI	N	Y	Y	Y	Y
Plotly	Y	N	Y	Y	Y
Gephi	Y	N	Y	Y	Y
Tableau	N	Y	Y	Y	Y

One of the biggest challenges with modern visualization tools is the latency created by processing large data sets [17]. This can be addressed by using pre-computed data, paralizing data processing and rendering and using predictive middelware.

[18] completed a comprehensive survey of tools used in Big Data applications. The concept of Big Data has different dimensions, one model that attempts to described these dimensions is widely known as 3Vs. The 3Vs stand for Volume, Velocity and Variety. Each of these dimensions pose its own unique challenges in the data management process.

The analysis and visualization of data was represented in a diagram that distinguishes between four different types of software solutions [18]. The diagram displayed in Figure 2.8 depicts the different types according to the Volume and Variety challenges in the 3Vs model.

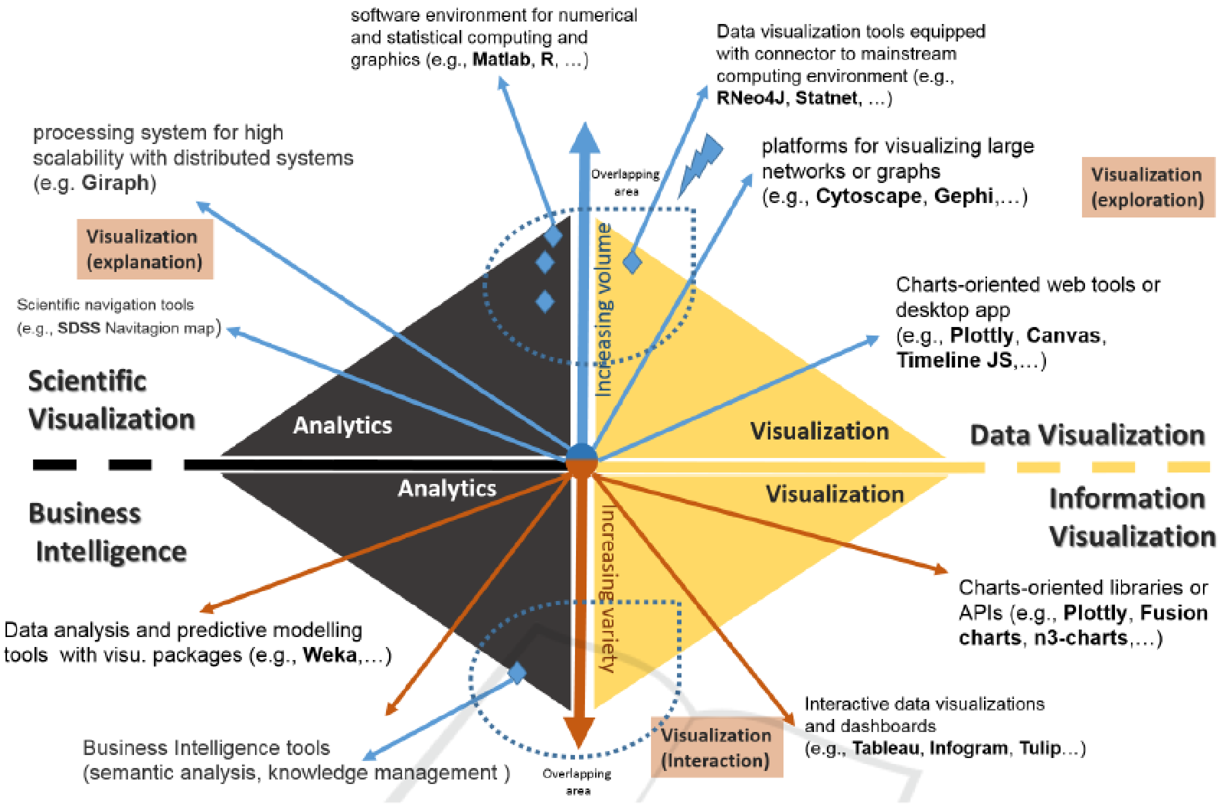
software environment for numerical and statistical computing and graphics (e.g., **Matlab**, **R**, ...)
Data visualization tools equipped with connector to mainstream computing environment (e.g., **RNeo4J**, **Statnet**, ...)
processing system for high scalability with distributed systems (e.g. **Giraph**)
platforms for visualizing large networks or graphs (e.g., **Cytoscape**, **Gephi**,...)
Visualization (exploration)
Overlapping area
Visualization (explanation)
Scientific navigation tools (e.g., **SDSS** Navitagion map)
Charts-oriented web tools or desktop app (e.g., **Plottly**, **Canvas**, **Timeline JS**,...)
Increasing volume
Scientific Visualization
Analytics
Visualization
Data Visualization
Business Intelligence
Analytics
Visualization
Information Visualization
Increasing variety
Charts-oriented libraries or APIs (e.g., **Plottly**, **Fusion charts**, **n3-charts**,...)
Data analysis and predictive modelling tools with visu. packages (e.g., **Weka**,...)
Interactive data visualizations and dashboards (e.g., **Tableau**, **Infogram**, **Tulip**...)
Visualization (Interaction)
Business Intelligence tools (semantic analysis, knowledge management)
Overlapping area

Figure 2.8. Big data visualization solutions

Differentiation is also done according to four main categories that describe the goals of visualization. The first goal would be the clear and effective communication of information to the user with the creation and study of the visual representation of data. The Information Visualization category is created to describe this goal. The second goal diverges from the first one in terms of spatial representation. If spatial representation is given to the visualization because of the intrinsic spatial layout of data like a flow simulation in a 3D space, then this can be seen as Scientific Visualization. These two categories can be further divided according to the dimension of volume and variety of data. In the end these four categories chosen by [18] to divide software are:

1. business intelligent tools for data mining of heterogeneous data
2. scientific visualization tools focused that focus on specific spatial representations using larger data sets
3. data visualization tools used for visual exploration
4. information visualization tools used for interactive visual exploration of data

[18] analyses 35 different software solutions in these four categories. What is interesting to note is that Plotly is well described in both the data and information visualization categories which implies that does well at handling big data challenges with variety and volume.

Visualization plays a critical role in these data driven problem solving techniques. [19] has shown that traditional methods like directed-acyclic graphs and Hasse diagrams can display casual links in an intuitive way but becomes less suitable when the number of nodes and interactions in a system grow. The use of two novel visualization techniques called Growing Squares and Growing Polygons demonstrate the use of animations, colors and patterns to better visualize casual relations.

Another approach to improve visualization of casual relations would be to adapt novel techniques used in other fields, similar to how PageRank was adapted to be used as LoopRank in industrial processes. Process monitoring and fault detection of batch processes has received attention in recent years. The extensive use of batch processing in the production of high value products makes it a important to ensure that batch runs are completed with a high level of success. [20] developed a novel approach to process monitoring and fault detection in batch processes by utilizing a technique they termed time-explicit Kiviat diagrams. The core idea behind this technique is based on Multi-way Principle Component Analysis that facilitates batch-wise unfolding of the three dimensional (Time*Variables*Batches) batch data matrix into a two dimensional matrix that captures the variation across batches. The type of data required to warrant the use of this technique is very similar to the data generated by multiple time region analysis in the LoopRank analysis done by [6]. Each batch can be seen as a time region in the LoopRank analysis and the analysis can be seen as the completion of multiple batch runs.

CHAPTER 3 METHODS (FOR ILLUSTRATION THIS HEADING CONTINUES ONTO THE NEXT LINE)

3.1 CHAPTER OVERVIEW

Chapter overview chapter overview chapter overview chapter overview chapter overview chapter overview chapter overview chapter overview chapter overview.

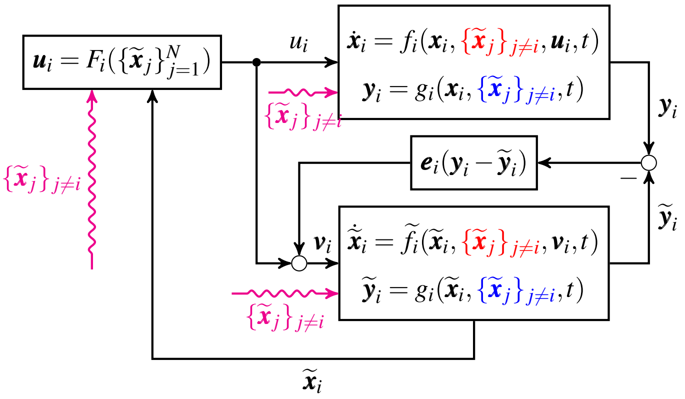
$\boldsymbol{u}_i = F_i(\{\widetilde{\boldsymbol{x}}_j\}_{j=1}^N)$
u_i
$\dot{\boldsymbol{x}}_i = f_i(\boldsymbol{x}_i, \{\widetilde{\boldsymbol{x}}_j\}_{j\neq i}, \boldsymbol{u}_i, t)$
$\boldsymbol{y}_i = g_i(\boldsymbol{x}_i, \{\widetilde{\boldsymbol{x}}_j\}_{j\neq i}, t)$
$\{\widetilde{\boldsymbol{x}}_j\}_{j\neq i}$
$\boldsymbol{y}_i$
$\{\widetilde{\boldsymbol{x}}_j\}_{j\neq i}$
$\boldsymbol{e}_i(\boldsymbol{y}_i - \widetilde{\boldsymbol{y}}_i)$
$-$
$\widetilde{\boldsymbol{y}}_i$
$\boldsymbol{v}_i$
$\dot{\widetilde{\boldsymbol{x}}}_i = \widetilde{f}_i(\widetilde{\boldsymbol{x}}_i, \{\widetilde{\boldsymbol{x}}_j\}_{j\neq i}, \boldsymbol{v}_i, t)$
$\widetilde{\boldsymbol{y}}_i = g_i(\widetilde{\boldsymbol{x}}_i, \{\widetilde{\boldsymbol{x}}_j\}_{j\neq i}, t)$
$\{\widetilde{\boldsymbol{x}}_j\}_{j\neq i}$
$\widetilde{\boldsymbol{x}}_i$

Figure 3.1. Another example of a figure

CHAPTER 4 RESULTS

4.1 CHAPTER OVERVIEW

Chapter overview chapter overview chapter overview chapter overview chapter overview chapter overview chapter overview chapter overview chapter overview chapter overview.

Table 4.1 shows a more complex example of a table.

Table 4.1. A comparison of the total thermal conductance of the $\mathrm{VO_x}$ devices at atmospheric pressure.

Device	Theoretical$^{(a)}$	Theoretical$^{(b)}$	Simulated	Experimental
	[μW K^{-1}]	[μW K^{-1}]	[μW K^{-1}]	[μW K^{-1}]
VOX4	14.28	11.02	9.55	8.30
VOX6	22.55	18.03	15.71	11.91
VOX7	22.89	18.11	15.91	13.27
VOX8	23.45	18.54	17.69	11.77
VOX13	18.16	14.96	14.39	13.45
VOX15	22.55	18.03	15.65	12.74
VOX23	21.54	17.62	15.62	9.27
VOX24	22.21	17.95	15.44	10.88
VOX32	16.97	13.75	11.41	9.31
VOX33	16.97	13.72	11.67	12.26
VOX36	21.48	17.56	16.51	15.42
VOX39	12.39	10.20	9.55	10.49

$^{(a)}$ Theoretical calculation using the standard approach.

$^{(b)}$ Theoretical calculation using the compensated method by including the thermal constriction resistance.

CHAPTER 5 DISCUSSION

5.1 CHAPTER OVERVIEW

CHAPTER 6 CONCLUSION

6.1 EXAMPLE HEADING

REFERENCES

[1] N. F. Thornhill and A. Horch, “Advances and new directions in plant-wide disturbance detection and diagnosis,” *Control Engineering Practice*, vol. 15, no. 10, pp. 1196–1206, 2007.

[2] K. Bryan and T. Leise, “The $25,000,000,000 eigenvector: The linear algebra behind google,” *SIAM review*, vol. 48, no. 3, pp. 569–581, 2006.

[3] M. Farenzena and J. O. Trierweiler, “Looprank: A novel tool to evaluate loop connectivity,” *IFAC Proceedings Volumes*, vol. 42, no. 11, pp. 964–969, 2009.

[4] A. Rahman and M. S. Choudhury, “New methods for detection of control loop interaction to prioritize control loop maintenance,” in *International Conference on Electrical & Computer Engineering (ICECE 2010)*. IEEE, 2010, pp. 263–266.

[5] S. J. Streicher, C. Sandrock *et al.*, “Eigenvector analysis for the ranking of control loop importance,” in *Computer Aided Chemical Engineering*. Elsevier, 2014, vol. 33, pp. 835–840.

[6] S. Streicher and C. Sandrock, “Plant wide fault and disturbance screening using combined transfer entropy and eigenvector centrality analysis,” *arXiv preprint arXiv:1904.04035*, 2019.

[7] D. Rees and R. S. Laramee, “A survey of information visualization books,” in *Computer Graphics Forum*, vol. 38, no. 1. Wiley Online Library, 2019, pp. 610–646.

[8] P. Isenberg, F. Heimerl, S. Koch, T. Isenberg, P. Xu, C. D. Stolper, M. Sedlmair, J. Chen, T. Möller, and J. Stasko, “vispubdata. org: A metadata collection about ieee visualization (vis)

publications," *IEEE transactions on visualization and computer graphics*, vol. 23, no. 9, pp. 2199–2206, 2016.

[9] P. Pirolli and S. Card, "The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis," in *Proceedings of international conference on intelligence analysis*, vol. 5. McLean, VA, USA, 2005, pp. 2–4.

[10] G. Klein, B. Moon, and R. R. Hoffman, "Making sense of sensemaking 2: A macrocognitive model," *IEEE Intelligent systems*, vol. 21, no. 5, pp. 88–92, 2006.

[11] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery in databases," *AI magazine*, vol. 17, no. 3, pp. 37–37, 1996.

[12] D. Sacha, A. Stoffel, F. Stoffel, B. C. Kwon, G. Ellis, and D. A. Keim, "Knowledge generation model for visual analytics," *IEEE transactions on visualization and computer graphics*, vol. 20, no. 12, pp. 1604–1613, 2014.

[13] N. Mahyar, A. Sarvghad, and M. Tory, "A closer look at note taking in the co-located collaborative visual analytics process," in *2010 IEEE Symposium on Visual Analytics Science and Technology*. IEEE, 2010, pp. 171–178.

[14] P. Saraiya, C. North, V. Lam, and K. A. Duca, "An insight-based longitudinal study of visual analytics," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 6, pp. 1511–1522, 2006.

[15] L. Barnard and M. Mertik, "Usability of visualization libraries for web browsers for use in scientific analysis," *International Journal of Computer Applications*, vol. 121, no. 1, 2015.

[16] S. A. Fahad and A. E. Yahya, "Big data visualization: Allotting by r and python with gui tools," in *2018 International Conference on Smart Computing and Electronic Enterprise (ICSCEE)*. IEEE, 2018, pp. 1–8.

[17] S. M. Ali, N. Gupta, G. K. Nayak, and R. K. Lenka, “Big data visualization: Tools and challenges,” in *2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)*. IEEE, 2016, pp. 656–660.

[18] E. G. Caldarola and A. M. Rinaldi, “Big data visualization tools: A survey,” in *Proceedings of the 6th International Conference on Data Science, Technology and Applications*. SCITEPRESS-Science and Technology Publications, Lda, 2017, pp. 296–305.

[19] N. Elmqvist and P. Tsigas, “Animated visualization of causal relations through growing 2d geometry,” *Information Visualization*, vol. 3, no. 3, pp. 154–172, 2004.

[20] R. Wang, T. F. Edgar, M. Baldea, M. Nixon, W. Wojsznis, and R. Dunia, “A geometric method for batch data visualization, process monitoring and fault detection,” *Journal of Process Control*, vol. 67, pp. 197–205, 2018.

ADDENDUM A DERIVATION OF HIGHER ORDER MODELS

A.1 EXAMPLE HEADING

A.1.1 Subheading example

Filler text to test template. Filler text to test template. Filler text to